Text analytics applications in job analysis and career research

Kobayashi, V.B.

Publication date
2023
Document Version
Final published version

VLADIMER B. KOBAYASHI

TEXT ANALYTICS APPLICATIONS IN JOB ANALYSIS AND CAREER RESEARCH

TEXT ANALYTICS APPLICATIONS IN JOB ANALYSIS AND CAREER RESEARCH

Vladimer Kobayashi

TEXT ANALYTICS APPLICATIONS IN JOB ANALYSIS AND CAREER RESEARCH

ACADEMISCH PROEFSCHRIFT
ter verkrijging van de graad van doctor aan de
Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. dr. ir. P.P.C.C. Verbeek
ten overstaan van een door het College voor Promoties ingestelde
commissie, in het openbaar te verdedigen
op donderdag 9 februari 2022 te 10:00 uur

door

Vladimer Birondo Kobayashi

geboren te Davao City, Philippines

PROMOTIECOMMISSIE

Promotor:	
prof. dr. E. Kanoulas	Universiteit van Amsterdam
Co-promotors:	
dr. S.T. Mol	Universiteit van Amsterdam
dr. G. Kismihók	Leibniz Information Centre for Science and Technology
Overige leden:	
prof. dr. K.G. Tijdens	Universiteit van Amsterdam
prof. dr. P.J. van Baalen	Universiteit van Amsterdam
prof. dr. A.E.M. van Vianen	Universiteit van Amsterdam
prof. dr. S. Bhulai	Vrije Universiteit Amsterdam
dr. C.C.S. Liem	Technische Universiteit Delft
dr. J. Lasser	Graz University of Technology

Faculteit Economie en Bedrijfskunde

This work was supported by the European Commission
through the Marie-Curie ITN EDUWORKS
(grant number PITN-GA-2013-608311)

ISBN: xxx-xx-xxxx-xxx-x

Book cover: *Mai Ongkiko*

Acknowledgement

This work would not be possible without the help of various people and organizations. Although, there are many people to thank for who have supported me in numerous ways, I cannot possibly list them all down here. Hence, I only mention here the people and organizations who were instrumental in finishing this dissertation.

I would like to express my sincerest gratitude to the European Commission who financed my entire post-graduate education from masters to this doctorate degree. Someday, I will be able to repay the people of Europe for their generous support. Looking back when I was faced with a difficult choice between staying in Europe and going somewhere else, I am happy that I accepted this PhD position.

I would like to express my profound gratitude to Prof. Dr. Evangelos Kanoulas who served as my promotor. His advice and comments shaped the form and structure of this dissertation. He is kind and brilliant, a rare breed indeed.

Also, I would like to thank my two supervisors, Dr. Stefan Mol and Dr. Gábor Kismihók who are with me throughout this journey. Thank you for advising and giving me the opportunity to have this PhD. And thank you for not giving up on me! Your relentless follow-ups kept me inspired to continue this PhD journey.

To the members of my committee, thank you for accepting our invitation. I have had the pleasure to directly work with some of you and you helped me think critically and clearly. Having your names included in this dissertation makes this dissertation shines even more!

Thank you to the University of Amsterdam, for hosting me, for providing me the best facilities, networking opportunities, and learning challenges. Thank you to the organizations who provided me with valuable data and who also taught me some practical knowledge in the fields of job analysis and career research. And to the University of the Philippines Mindanao, who believed and supported my dream of studying abroad. To my colleagues in the department, special mention to Dr. Norberto Navarrete, Jr, I am delighted that our dreams for the department are coming true.

I would like to thank the many individuals who I have the pleasure to work and collaborate. Their ideas and the many lessons they imparted to me form part and parcel of this dissertation. To my EDUWORKS family, special mention to Sofija, Eloisa, Jovana, and Hannah, who helped me surmount several obstacles through their support and understanding. I missed our summer and winter schools together, which were not only source of knowledge but also of fun memories. To our trainers who themselves are luminaries in their fields, I always carry with me the learnings you have imparted.

Lastly, I would like to thank my family, my mother Winefreda and father Jesser, who love me beyond measure and who remain steadfast in their belief that I am the best son they ever have (full disclosure: I am their only son). To my parents in Amsterdam, Mommy Wilma and Tatay Rene, thank you for the love and care. I will always cherish the times we have spent together. Tatay has taught me so much about life and how to survive it and Mommy Wilma taught me how to remain calm and demonstrate finesse in whatever situation I am in. Know that I am always here for you despite my busy schedule. To my aunt, mama Delia, who has supported my education at an early and who did not fail to buy me school supplies during my grade school years, may God continue to bless you with good health and long life. To my grandmother, Leonila, whom I dearly love, it pains me to know that you are not here to witness this culmination. Your dream of having an education which was cruelly denied to you in a time and place when women are expected to marry, stay home, and do household chores, has engulfed me with a burning desire to study with utmost perseverance. This work I dedicate to your memory.

The happiest and saddest part of this PhD journey is it must end. But the friendship, camaraderie, collegial exchanges, and learnings I have gained through this PhD have enriched my soul and being.

Table of Contents

Introduction

Every day, organizations and researchers collect and generate text data. Some examples of text data include company reports, open-ended survey responses, interview transcripts, and stories and news published in websites or social media pages. Analyzing such text data may provide actionable insights about people, organizations, and processes. Examples of analyses include ascertaining personality from blogs (Yarkoni, 2010), business model concept extraction and business model evolution analysis from company annual reports (Lee & Hong, 2013), capturing public views of immigration from responses in open-ended surveys (Roberts et al., 2014), and identifying coaching criteria and deducing the length of coaching experience from evaluation reports (Theeboom et al., 2017). Since text data are abundant and likely to contain relevant knowledge, they can be analyzed to enhance processes that produce superior strategies and gain competitive advantages, as well help shed light on scientific facts.

Organizational researchers have traditionally analyzed text data by applying qualitative approaches. However, the sheer magnitude of text data that exist about issues of interest oftentimes makes qualitative analysis impractical and inadequate. Hence, these challenges remain: how do we popularize text analysis among organizational researchers to empower them to make better use of these data? What sort of problems can text analysis help address, and crucially, how can such data be analyzed expediently and effectively?

Unlike structured data which are arranged in tabular format (i.e., rows and columns), text data often appears in an unstructured or free form. The analysis of unstructured text data poses a challenge, since many analytical methods only accept data in tabular format. Hence, text data needs to be preprocessed and transformed before it can be subjected to further analysis using statistical or analytical methods. The preprocessing usually involves the deletion of irrelevant textual and non-textual content and applying suitable normalization (e.g., stemming, lemmatization) (Palmer, 2010; Uysal & Gunal, 2014). Transforming text data means extracting features from them and representing such data as a mathematical structure, a vector. Another challenge is how to choose an appropriate analytical method for a given problem. Advances in data analytics have led to the development of a dizzying array of methods that are tailored to the analysis of text data, and over the past years, new analytical procedures have been proposed including, more recently, deep learning approaches. Choosing which method to deploy always depends on the research problem. Creating a taxonomy of problems and applications, then mapping the methods that can suitably address them may help organizational

researchers situate the goals of their own investigations. These analyses may be one of the following: automatic extraction of topics, sentiment analysis, opinion mining, and other predictive analyses (Zhang et al., 2015).

Text analytics techniques can even be useful in studies that do not primarily deal with the semantics of text. For example, due to the sequential nature of text, some techniques developed for them have been used to study time series (continuous values) and categorical sequences (e.g., DNA sequences). Hence, the benefits of popularizing text analysis not only contribute to the analytical toolkit of organizational researchers, but also espouses collaboration across different fields.

Text analytics in HR

Human Resource Management (HRM) researchers have started to explore the use of analytics as part of their research strategy. Moreover, some HR functions such as job analysis, recruitment, and career management generate huge amounts of data in the form of job descriptions, resumes, interview transcripts, and survey responses. Arguably text data is the most abundant type of data. However, most of these data are largely underutilized and seldom exploited by researchers. This is possibly because some HR researchers are not familiar with text analytical techniques. Furthermore, the benefits of qualitative techniques are offset by the time and effort required to apply the techniques effectively, as well the constant threat of introducing researcher's biases into the analysis. In addition, modern text analytical methods enable the analysis of text at a larger scale and using a myriad of text sources, and such methods produce more efficient, effective analysis and tend to reduce bias from using only one source of data.

Applications of data analytical techniques to address HR problems coalesce into a new field called HR analytics (Marler & Boudreau, 2017). HR analytics have largely focused on the analysis of numerical data examples, which include optimizing performance (Sharma & Sharma, 2017), employee engagement (Gaur, 2020), and employee turnover (Avrahami et al., 2022). Often, there is an outcome variable and a set of independent variables that are used to build predictive models, and data collections are customarily conducted using questionnaires that generate numerical data from the onset (e.g., Likert-type). But there are also instances where researchers must deal with free form text (e.g., responses to unstructured job interviews, performance appraisals, etc.). A straightforward approach would be to use simple text summarization techniques such as counting the frequencies of words. However, this approach may be so reductionist that crucial information is lost. This inadequacy is especially pronounced in recruitment and selection, where text data are abundant but are seldom, if at all, used in more sophisticated analyses.

This partly explains why matching candidates to jobs, still takes a lot of time and is often fraught with error.

The primary goal of text data analysis is to extract information and produce deeper insights otherwise not attainable if text is excluded from the analysis. Text captures richer context and may reveal otherwise unseen nuances of individuals and organizations. The benefits are further magnified if text analysis is performed at scale harnessing the supremacy of big text data through more representative samples. The magnitude of text and textual sources and the possibility to analyze them simultaneously may eliminate bias, permit triangulation, and enhances the validity of the results. Hence, for this dissertation I have developed models and/or analyzed large text data from numerous sources that contribute insights about jobs and people. The findings here have implications not only for making practical HR decisions (such as in career planning, jobs to candidates, and candidate sourcing) but also for advancing knowledge in the HR field (e.g., understanding career paths and physical job mobility).

Job analysis (JA) refers to the study of work role requirements and the context in which these roles are enacted (Brannick, Levine, & Morgeson, 2007). Job data may either yield information about the job (i.e., work activities that include tasks, responsibilities, and job roles) or about the person fulfilling that job ((i.e., worker attributes such as the knowledge, skill, and abilities (KSA) of workers) (Brannick et al., 2007a). Job analysis may be conducted within the context of a specific organization or across organizations.

Methods of data collection in job analysis usually include observation, individual interviews, focus groups, and questionnaires (Sackett & Laczo, 2003a). Some issues encountered in job analysis include managing potential sources of bias, the time it takes to collect data, and the level of standardization (Morgeson & Campion, 2000). On one hand, data collected through observation does not suffer from reporting bias but may suffer from observation bias and require a lot of time to collect. Interviews or focus groups, on the other hand, may result in more data collected in less time but at the cost of incurring a high reporting bias such as disclosing only positive aspect of the job, or even recall bias.

The internet is a rich source of data for JA (Smith & Ali, 2014). Examples of online data that may be relevant to JA are job vacancies and resumes (Dusi et al., 2015). Hundreds of thousands of job vacancies are posted online across thousands of websites on a daily basis, and these vacancies offer an abundant source of job analysis data which in turn provides opportunity to augment existing ways to collect job information. Our assumption is that these vacancies often contain up-to-date information about jobs, and that they also reflect the role requirements

demanded by organizations (Harper, 2012; Sodhi & Son, 2010).

As mentioned above, resumes can also be used as a source of data for JA (e.g., characteristics of people (Sobotka et al., 1993)). Additionally, we find in resumes the educational qualifications and job experiences of individuals, which may both be useful for predicting physical job changes which in turn may result in identify specific career trajectories. Finally, analyzing resumes may also be expected to yield insights about jobs themselves, for example, through the study of job transitions.

The magnitude and velocity of vacancy and resume data sometimes preclude researchers from taking full advantage of them. As most of these sources are in text and manually extracting information from text is time consuming and laborious, there is a need to automate this process. In this way, we set out to augment traditional JA with text analysis techniques so that analysis can be scaled up and enriched. In this dissertation, the application of text analytical techniques to job analysis data is termed as Job Analytics.

The application of Job analytics to analyze text may be expected to generate knowledge about jobs and people (e.g., what people need to know to be able to perform their jobs) that are necessary to serve HR practices, such as recruitment, training, and development (Bennett Jr, Edens, & Bell, 2003; Schömann & O'Connell, 2003; Yamnill & McLean, 2001).

This Dissertation

Since organizational research is a broad area (including human resources management, job analysis and career research) that cuts across many disciplines, a discussion for the motivation of employing text mining methods and proposing a framework for text analysis (presented below) will set the tone for the succeeding analyses. Consequently, the framework will guide the researchers on a principled approach to applying text analytics to actual research. As such, this dissertation is organized into three themes: (i) proposing a framework and a taxonomy of text mining methods that is applicable to organizational research, (ii) applying text mining to job analysis through vacancy analysis (iii) analyzing timing and choices of job mobility to elucidate the determinants of physical job mobility by analyzing resumes.

This dissertation aims to address research questions relevant to organizational research using text analytics and to furnish new analytic strategies that can be useful for job analysis, career and HR research in general. The framework and methods proposed here would enable researchers to apply text mining (defined as a process to collect, manage, and analyze text big data) in

a systematic manner. The contribution of this dissertation is thus threefold: to create models that are of interest primarily to job analysis and career researchers, to derive knowledge from the models that can serve as input to theory building, and to demonstrate the practicability of the models to job recommendation and recruitment. Specifically, the focus is on the following five aspects:

First is to elucidate the text mining process to the general audience of organizational researchers; this has the purpose of popularizing text mining that will enable organizational researchers to analyze large corpora of text. The complexity of problems in organizational research needs not just one source or one type of data but a combination of different data, enabling the aggregation and triangulation of results. Additionally, this dissertation highlights opportunities for text analysis-based research, explains the text mining process in detail and discusses representative techniques for each problem or task.

Second is to provide a review of and tutorial on conducting text classification as it is one of the fundamental methods in many prediction tasks. Indeed, many problems, such as job performance classification and sentiment analysis in organizational research can be treated as classification problems. The tutorial also illustrates how text classification can be used in a job analysis of nursing. Specifically, we proposed an algorithm that can automatically extract tasks from nursing vacancies which complements observation and interview with subject matter experts.

Third is to build a classification model for automating the extraction of job information from job vacancies. Every day, hundreds of vacancies are posted to various job boards, which contains information about jobs including roles, responsibilities, and tasks. They also provide information about the qualifications of potential candidates such as the knowledge, skills, and attitudes required to effectively perform the job. Company information, compensation scheme, type of employment, and contact information are also present in these job vacancies. Although the purpose of job vacancies is primarily to advertise a job to attract candidates, the information contained therein could be used for job analysis. The challenge is to extract the relevant information. Through the classification model we can obtain job specific information applicable to understand skill requirement and tasks of a job. Moreover, with the output of this classification task, we demonstrate how to automatically build skill taxonomies and job clusters. Using vacancy data is advantageous because we can get job information across different organizations and geographical areas, with the information obtained being up to date because vacancy analysis can be done in near real time. Consequently, applying the classification models ensures that we can analyze vacancies at scale

and extract job information in a timely manner which stands in stark contrast to traditional job analysis, which is time consuming, painstakingly laborious and with results that will likely be obsolete since jobs can be dynamic (e.g., especially in the IT industry where technology and tools are continuously updated).

Fourth is to demonstrate the applicability of text analytics to careers research. Careers research studies often recommend the analysis of longitudinal and mostly numerical data. The introduction of text analysis would expand the available tools used by researchers and exemplify new avenues of research using text data. Moreover, a model was built that can predict salary from vacancies. The model can be examined to understand differences in salary structures among jobs and by investigating further we can determine the factors that drive such differences. By mapping job characteristics such as the industry, role, skill requirement and location we managed to shed light on pay differences and roles that tend to pay well. Some of the findings reveal that managerial and financial roles tend to pay well, while jobs that explicitly mention the communication skill in the vacancies pay relatively less well, as these skills appear to be the ones most often referenced in low paying jobs.

Fifth is to extract job histories resumes to create models that can predict the timing of job changes and predict the possible next job. This helps career researchers identify those factors that propel people to change jobs, and for recruiters to predict the timing of job changes so that they can approach that candidate who is more likely to accept a job offer. By investigating the models, we acquire an understanding of career progression, physical job mobility, and timing of turnover. In candidate sourcing, usually the goal is not only to find the best matching candidate but also to find candidates who are ready to switch jobs. Finally, by using the patterns in job changes across individuals, we can contribute to the literature on job analysis by operationalizing a measure of job similarity that is based entirely on job histories.

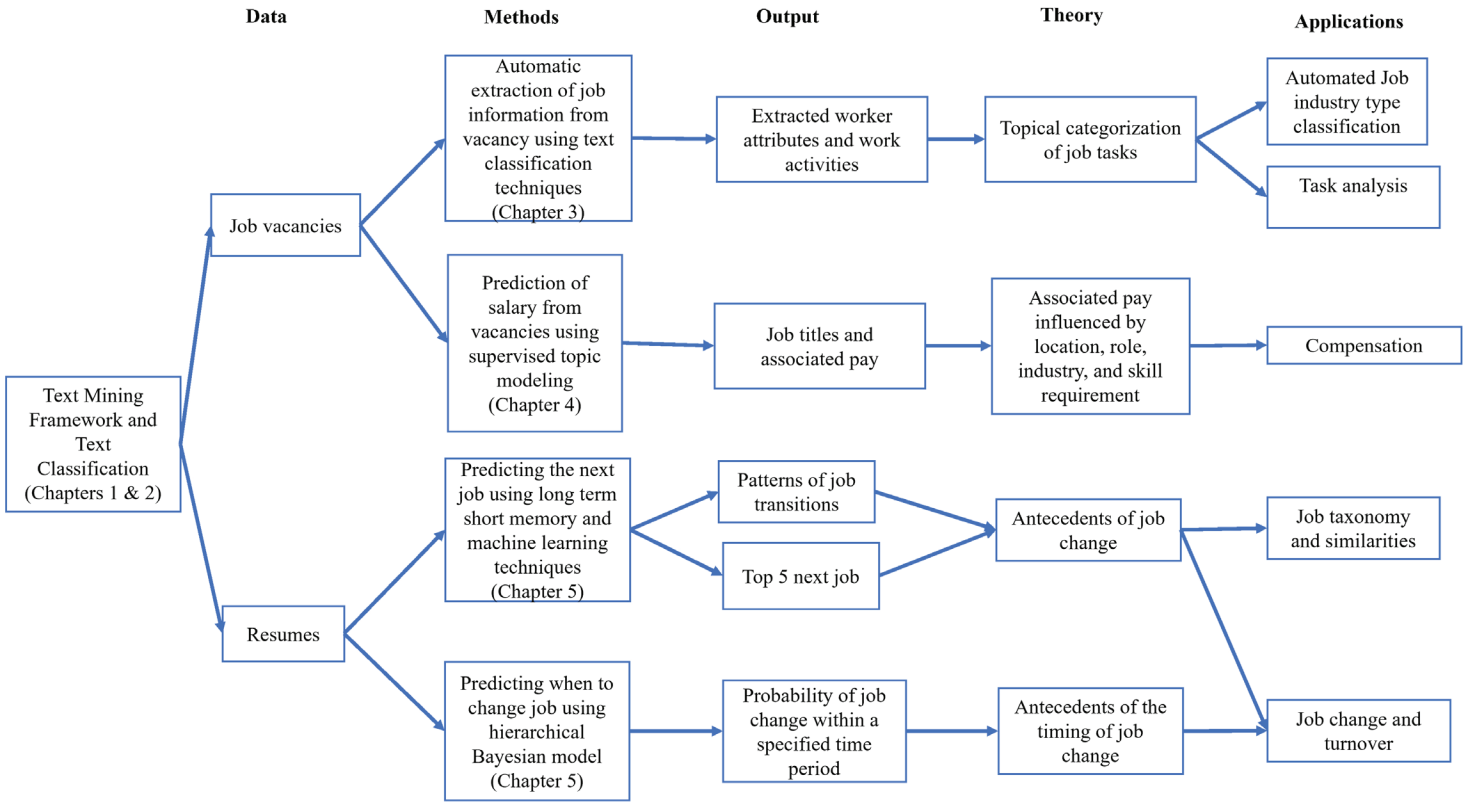


Figure 0.1. Diagrammatic representation of the data, methods, output, theory, and applications covered by the various chapters of the Dissertation.

Dissertation Overview

Figure 0.1 provides a diagrammatic framework of this dissertation. This dissertation consisted of 4 parts. The first part consisted of 2 chapters and provides a generic exposition of text mining as a viable method for organizational researchers. The second chapter is a tutorial in text classification which I used to develop an algorithm for the automatic extraction of work information from job vacancies. These two chapters contribute to the first theme, namely, proposing a framework and a taxonomy of text mining methods that is applicable to organizational research, along with providing a review, tutorial, and practical demonstration of text classification.

The second part elaborates on the applications of vacancy mining. Applications of vacancy mining include the application of text mining to automatic job information extraction, job clustering, job classification to industries, analysis of hybrid teachers, and task analysis for nurses.

The third part tackles resume mining in which I trained models that can predict the next job or career progression of a person using his/her education and job histories from resumes. I then used the output of the models to investigate patterns of mobility and to identify factors affecting the decision to change job. Also, I obtained job clusters from vacancies and compared them to the clusters obtained from job histories in resumes. These investigations contributed novel insights related to job mobility and employability.

The last part focuses on the prediction of salary from vacancies and exposes yet another means of how text mining can be used in career research.

Below we elaborate the main research questions of every chapter.

Text mining in organizational research

The abundance of text data opens new avenues for research, but it also presents research challenges. One challenge encountered is how to manage and extract meaning from massive amounts of text, since reading and manually coding text is a laborious exercise. To take full advantage of the benefits of doing research with 'big' text data, organizational researchers need to become familiar with techniques that enable efficient and reliable text analysis. Extant TM reviews and tutorials (Ghosh, Roy, & Bandyopadhyay, 2012; Gupta & Lehal, 2009; Solka, 2008) have mainly targeted readers with a strong affinity with programming and machine learning, and have focused on technical aspects of the TM process. Hence, for organizational researchers to gain confidence in using text mining in their research, we should demonstrate the applicability of TM to enhance organizational research, and illustrate TM and its methods and provide practical recommendations at each step in the TM process.

This chapter acquaints organizational researchers with the fundamental principles underpinning text mining, the analytical stages involved, and contemporary techniques that may be used to address different types of objectives. The following specific analytical techniques are reviewed: 1) dimensionality reduction, 2) distance and similarity computing, 3) clustering, 4) topic modelling, and 5) classification.

These were chosen because they encompass the bulk of text mining applications, and also because they follow the distinction of analytical techniques, which are supervised (classification) and unsupervised (clustering, topic modelling). Furthermore, we provide a description of how text mining may augment contemporary organizational research by allowing the testing of existing or new research questions with data that are likely to be rich, contextualized, and ecologically valid. After an exploration of how evidence for the validity of text mining output may be generated, an illustration of the text mining process in a job analysis setting using a dataset comprised of job vacancies is provided.

Text classification for organizational researchers: A tutorial

Organization research often addresses questions such as who will quit his or her job, whether to hire a candidate or not, and which organizations will survive in the long run. Each one of these questions can be reformulated as a classification problem. The same with using text, classifying texts or parts thereof into categories is likely to enable more effective use of information. Manual procedures for text classification work well up to a few hundred documents as is common in qualitative analysis. However, manually assigning large collections of text to categories is costly and may become inaccurate and unreliable due to cognitive overload on the part of those doing the classifying. Furthermore, idiosyncrasies among human coders may creep into the labeling process resulting in coding errors. Apart from actual categorization, we might also be interested in discovering the reasons for categorizing one object into one category than in the other, hence, insights that can be derived from the classification itself will likely be of interest to researchers who want to explain the underlying process or mechanism leading to the categorization.

One workaround is to code only part of the corpus (primary or secondary) as opposed to coding all documents. However, this comes at the expense of possibly omitting relevant information, which may lead to bias and a degradation of the internal and external validity of the findings. Another option is to hire multiple human coders, but this adds cost (e.g., cost of hiring and training coders) and effort pertaining to determining inter-rater reliability and consensus seeking (Sheng, Provost, & Ipeirotis, 2008). A final (and more affordable) option is to solicit the help of the public to label text, for instance through the Amazon Mechanical Turk platform (Buhrmester, Kwang, & Gosling, 2011). However, this may only

be effective in labeling objective information (e.g., names of people, events, etc.) since it is often difficult to establish consistency on more subjective labels (e.g., sentiments) (Wiebe, Wilson, Bruce, Bell, & Martin, 2004). Hence, automatic text analysis procedures that reliably, efficiently, and effectively assign text elements to classes are both necessary and advantageous, especially in seeking to process a massive corpus of text.

Text classification techniques facilitate the automatic assignment of text strings to categories, making classification expedient, fast, and reliable, which creates potential for its application to organizational research. To popularize and illustrate text classification, the following goals were formulated: **(i)** increase the uptake of text classification among organizational researchers; (ii) Elucidate the text classification process as applied to organizational research; and (iii) establish validity for results derived from text classification.

We devised a tutorial about text classification. The tutorial is drawn from our own work on job vacancy mining. The tutorial consists of an exemplification of the text classification process in several roughly sequential steps, namely training data preparation, preprocessing, transformation, application of classification techniques, and validation. Concrete recommendations are provided at each step. To help researchers develop their own text classifiers, the R codes associated with each step are provided. Finally, we discussed how researchers can validate a text classification model and the associated output by either employing subject matter experts (SMEs) or triangulating the results with data from observation. A high overlap between the list of tasks collected by text classification and the list of tasks collected in the task inventory could be taken as evidence for convergent validity. Conversely, one could establish discriminant validity, or a very low correspondence with so called 'bogus tasks' that are completely unrelated to a job.

A Fresh Perspective on Job Analysis: Extracting Job Information from Job vacancies

Online vacancies provide a novel and rich source of job information that may complement traditional job and labor market analyses. Previous studies aimed at extracting relevant information from vacancy data have by and large leveraged methods that rely on counting prespecified keywords. Although keywords can be useful in tasks such as determining the frequency of certain skills or competences, they are inadequate for other tasks such as discovering new or emerging skills or competencies. Another disadvantage of using keywords is that many skills cannot be captured in a single word (e.g., full stack development). Of course, there is always the option of extending the analysis to bigrams or trigrams (i.e., two- or three-word strings), however, this may introduce unnecessary complexity in the analysis. Furthermore, there may be other pertinent information encapsulated in vacancies such as specifics about job roles and responsibilities, for which keywords

may also not be the best choice. To improve upon keyword-based extraction, use the results to identify skill groups and knowledge domains, and consequently to validate the results, we formulated the following research questions:

RQ1 What features are useful to identify work activities and worker attributes, and can these features be combined to build an automatic and accurate classifier that can distinguish between work activities and worker attributes contained in job vacancies?

RQ2 Can the extracted work activities be used to construct task groups and meaningfully cluster jobs based on tasks?

RQ3 Can the task groups be validated by comparing it to tasks as enumerated in the European Skills, Competences, and Occupations (ESCO)?

To address RQ1, this paper proposes the use of state-of-art text classification techniques coupled with a rich set of features that incorporate word-based, syntactic, and grammatical features of text found in vacancies. The results showed our approach is effective in sorting vacancy content into worker attributes and worker activities. Furthermore, we demonstrate how the extracted job information can be used to extract skills and cluster jobs, and how to validate the extracted information by comparing it to an existing job taxonomy that was independently constructed by experts.

Predicting the next job and when to change jobs

Studies on physical job changes have traditionally focused on examining underlying psychological and sociological factors influencing the decision to change jobs. Although individual characteristics have been shown to play a significant role in job changes as possible antecedents (T. W. H. Ng et al., 2007), research has much less frequently investigated the actual job transitions people make as well as the timing of these transitions. By analyzing resume data at a massive scale, we are able to examine common patterns of changes. In the study reported on in Chapter 4, we used novel analytical methods to study job transitions and set out to solve two intimately interrelated problems. First is the prediction of whether a specific candidate is going to change job within a specified time. Second is the prediction of the most likely next job. Apart from the job transition, covariates such employment history, candidates' education level, and other factors were incorporated in the analysis. The research questions are:

RQ4 How to build a model that can predict when someone is going to change jobs?

RQ5 How to build a model that can predict the most likely next job of an individual?

RQ6 How to use the models from RQ4-RQ6 to find similarities among jobs

and investigate whether industry differences play a role in mobility. That is, do some industries exhibit a relatively higher number of job transitions?

We found that it is possible to predict the timing and next job of candidates from job transitions alone. Also, the models we employed revealed the following: The likelihood that someone is going to change jobs within a period is an interplay of various factors which include the type of industry someone belongs in, the number of previous jobs transitions, time in the labor market, and educational attainment. Age and time spent in the current job are two strong predictors of the tendency to change jobs. This study contributes to the field of recruitment and job recommendation in that it does not only predict when a candidate is going to change job, but also what transitions are enacted. Another contribution is its implication to job analysis, because we demonstrated that job similarity can be deduced from the job transitions that people make. Lastly, we managed to demonstrate that topics such as job mobility can be approached from a more data-driven perspective by applying modern analytical tools and by using theory to guide the selection of variables as well as to guide the model building process.

Text mining in career studies: Generating insights from unstructured textual data

Up to this time, few studies in career literature utilize unstructured text as data. Text mining enables career researchers to augment their text analytical toolkit with powerful analytical techniques from machine learning and statistics. For this research we wanted to answer the following research questions:

RQ7 How to build a model that can predict salary from job descriptions in vacancies?

RQ8 What are the differences in pay structures among jobs and what drives these differences?

As the chapter is also about popularizing the use of text mining among career researchers, we start it by explaining the text mining process. We then proceeded to answer RQ7 and RQ8 by building a model using supervised topic modeling. We found differences in pay structure according to the nature of the job. High paying jobs appear to more often involve strategizing and managing risk and investment in finance whereas low paying jobs are dominated by call center and cleaning activities. Some skills such as data analysis, leadership, IT, and problem-solving skills are also highly valued by employers. Conversely, communication and verbal skills seem to be at the lower end of the pay scale, perhaps because vacancies that explicitly mention these are also in the low pay scale, something that has been

picked up by the classifier. Another possible explanation is that communication and verbal skills are among the most common skills, hence why their associated pay is lower.

Summary of Main Contributions

In this section, the theoretical, methodological, and empirical, contributions of this dissertation are summarized. For each contribution, the chapter in which it can be found is listed.

Methodological contributions

In this dissertation, a framework for conducting text mining in the context of organizational research which is termed as KDTD (Knowledge Discovery from Text Databases) is provided. The principal idea originated from Knowledge Discovery from Databases (KDD). Also provided are examples of research questions that may be suitable for KDTD (Chapter 2)

A model which improves upon keyword-based extraction because it managed to extract richer description of work attributes and work activities. The model uses word-based, syntactic, and grammatical features. The output from the model also showed convergent validity as evidenced by the high overlap of skills from the two sources (Chapter 3).

In studying job changes, I applied a hierarchical Bayesian probabilistic model to predict when someone will change job and a transformer-based model to predict what job someone will take next (Chapter 5)

Empirical Contribution

I provide a taxonomy of skill groups and knowledge domains extracted from vacancy data and an empirical demonstration of how to validate outcomes of text analysis, by triangulating with expert organized standard taxonomies of job information in Chapter 3.

I managed to develop an empirical derivation of job similarity through analysis of transitions in Chapter 5.

Theoretical Contribution

I have identified factors that determine the likelihood of physical job mobility. Factors include the number of previous jobs transitions, time in the labor market, and educational attainment.

Lastly, the model for predicting salaries from vacancies revealed several determinants of salary including industry type and location. Also, skill-based determinants also explained differences in pay structure.

Educational and software contribution

As part of the research carried out for the dissertation, I contributed tutorials with codes and web applications to the community. I endeavored to champion the use of text mining in organizational research by providing a tutorial with codes about text classification to the community. This hands-on approach may be expected to help interested researchers apply text classification in their own research (Chapter 2)

I also developed some web applications to illustrate the capability of the constructed models and for researchers to test the model. One app can detect language in the vacancy which is a crucial step prior to information extraction. Sentences in a vacancy are then sorted into ones containing worker attributes or activity. Also, the app is able to identify which industry the vacancy should be categorized in. (Chapter 4)

Origins

In this section, I list the publications which each chapter of this dissertation is based on.

Chapter 2 is based on the following paper:

Kobayashi VB, Mol ST, Berkers HA, Kismihók G, Den Hartog DN. Text Mining in Organizational Research. Organizational Research Methods. 2018;21(3):733-765. doi:10.1177/1094428117722619

Chapter 3 is based on the following paper:

Kobayashi VB, Mol ST, Berkers HA, Kismihók G, Den Hartog DN. Text Classification for Organizational Researchers: A Tutorial. Organizational Research Methods. 2018;21(3):766-799. doi:10.1177/1094428117719322

Chapter 4 is partly based on the following paper:

Kobayashi, V. B., Mol, S. T., Kismihók, G., & Hesterberg, M. (2016). Automatic Extraction of Nursing Tasks from Online Job Vacancies. In M. Fathi, M. Khobreh, & F. Ansari (Eds.), Professional Education and Training through Knowledge, Technology and Innovation (pp. 51–56). UniPrint - University of Siegen. http://dokumentix.ub.uni-siegen.de/opus/volltexte/2016/1057/pdf/Professional_education_and_training.pdf#page=58

Chapter 5 is based on the following paper (book chapter)

Kobayashi, V. B., Mol, S. T., Vrolijk, J., & Kismihok, G. (2021). Text mining in career studies: Generating insights from unstructured textual data. In W. Murphy & J. Tosti-Kharas (Eds.), Handbook of Research Methods in Careers. Edward Elgar Publishing.

The dissertation also indirectly builds on the following papers.

Kobayashi, V. B., Mol, S., & Kismihok, G. (2014). Labour Market Driven Learning Analytics. Journal of Learning Analytics, 1(3), 207–210. https://doi.org/10.18608/jla.2014.13.24

Kobayashi, V., Berkers, H. A., Mol, S. T., Kismihok, G., & Den Hartog, D. N. (2015, August 7). Augmenting Organizational Research with the Text Mining Toolkit: All Aboard. 75th Annual Meeting of the Academy of Management. http://proceedings.aom.org/content/2015/1/13517

Kobayashi, V., Mol, S. T., & Kismihók, G. (2015). Discovering Learning Antecedents in Learning Analytics Literature. In E. Duval, K. Verbert, J. Klerkx, M. Wolpers, A. Pardo, S. Govaerts, D. Gillet, X. Ochoa, & D. Parra (Eds.), Proceedings of the First International Workshop on Visual Aspects of Learning Analytics co-located with 5th International Learning Analytics and Knowledge Conference (LAK 2015), Poughkeepsie, NY, USA, March 16-20, 2015 (Vol. 1518, pp. 45–48). CEUR-WS.org. http://ceur-ws.org/Vol-1518/paper9.pdf

Mol, S. T., Kobayashi, V., Kismihók, G., & Zhao, C. (2016). Learning through goal setting. In D. Gasevic, G. Lynch, S. Dawson, H. Drachsler, & C. P. Rosé (Eds.), Proceedings of the Sixth International Conference on Learning Analytics & Knowledge, LAK 2016, Edinburgh, United Kingdom, April 25-29, 2016 (pp. 512–513). ACM. https://doi.org/10.1145/2883851.2883859

Theeboom, T., Van Vianen, A. E. M., Beersma, B., Zwitser, R., & Kobayashi, V. (2017). A practitioner's perspective on coaching effectiveness. In L. Nota & S. Soresi (Eds.), Counseling and coaching in times of crisis and transitions: From research to practice. Routledge Publishers.

PART 1

TEXT MINING AND TEXT CLASSIFICATION FOR ORGANIZATIONAL RESEARCHERS

Text Mining in Organizational Research

In this chapter we introduce Text Mining to organizational research by presenting a methodological framework inspired from the knowledge discovery from databases (KDD) process which we call the KDTD process. Each step is elaborated complete with practical recommendations and towards the end we provide an illustrative example for job analysis.

Organizations are increasingly turning to big data and analytics to help them stay competitive in a highly data-driven world (LaValle et al., 2013). Although difficult to assess let alone verify (Grimes, 2008), around 80 percent of data in organizations are commonly estimated to consist of unstructured text. The abundance of text data opens new avenues for research, but it also presents research challenges. One challenge is how to manage and extract meaning from massive amount of text since reading and manually coding text is a laborious exercise. To take full take advantage of the benefits of doing research with big text data, organizational researchers need to be familiarized with techniques that would enable efficient and reliable text analysis.

Text mining (TM) is "…the discovery and extraction of interesting, non-trivial knowledge from free or unstructured text" (Kao & Poteet, 2007, p. 1). Knowledge is derived from patterns and relationships and can be used to reveal facts, trends, or constructs (Gupta & Lehal, 2009; Harlow & Oswald, 2016). A related technique which organizational researchers may be more familiar with is computer-aided text analysis (CATA). To date, most studies employing text analysis in organizational research are CATA-based (Kabanoff, 1997; McKenny et al., 2013; Short et al., 2010). CATA (McKenny et al., 2013) is a special class of TM. Whereas most CATA procedures extract patterns by counting word/term frequencies, TM generally also capitalizes on other textual properties, such as grammar and structure, and employs techniques from natural language processing, computational linguistics, corpus linguistics, machine learning, and statistics. Hence, TM is more powerful and can be used for a wider range of purposes than CATA.

We have three aims for this paper. First, to illustrate how TM might enhance the field, we provide examples of questions in organizational research where TM may help generate insight. Second, we illustrate TM and its methods and provide practical recommendations at each step in the TM process. Third, we demonstrate the application of TM to job analysis by showing how TM can automatically extract job information to derive job skill constructs from job vacancies.

Extant TM reviews and tutorials (Ghosh, Roy, & Bandyopadhyay, 2012; Gupta & Lehal, 2009; Solka, 2008) have mainly targeted readers with a strong affinity with programming and machine learning, and hence have focused on technical aspects of the TM process. Here, our intended audience comprises organizational researchers. We describe the key steps in text mining research and aim to enhance organizational researchers' understanding of the concepts behind text mining, the different steps involved, and strategies for evaluating the validity of TM-based outcomes. Some technical details have been left out, though references for further reading are provided. With this paper we hope to inspire investigations that apply TM to the analysis and understanding of organizational phenomena.

Examples of the Uses of TM

Though to date TM has been largely used for exploratory purposes (i.e., focusing on describing or mapping new phenomena) it can also be applied to explanatory/theory driven research (i.e., hypothesis testing on the interrelationships between constructs). An example of exploratory TM can be found in Singh, Hu, & Roehl (2007), who identified emerging research streams in human research management (HRM) by analyzing published literature in this area. However, the latter use is likely more appealing to organizational researchers, as they will likely aspire to validate domain knowledge, models, frameworks, or theories in their research.

Existing knowledge and theory can be empirically tested using TM. An example is a study by Yarkoni (2010) in which he investigated the relationship between personality and language use. Specifically, he counted the frequency of words from 66 psychologically relevant categories (such as 'positive emotions', 'hearing', 'sexuality', and 'swear words') (LWIC; see Pennebaker, Francis, & Booth, 2001) in 694 blogs and correlated them to the Five Factor Model dimension scores obtained by surveying 576 bloggers. He showed that "personality plays a pervasive role in shaping the language people use" (Yarkoni, 2010, p. 371). Another example of such theory driven work is the study of Guo, Li, & Shao (2015) who developed features derived from the theory of cognitive situational models to cluster documents. In applying the four dimensions of the cognitive situational model (i.e., Protagonist, Temporality, Spatiality, and Activity) they managed to reduce feature size and were able to analyze complex semantics. These two studies would have been difficult to conduct with mainstream qualitative research methods, due to the laboriousness of manually coding more than 115,000 words per blog or grouping 825,992 articles.

TM could also be applied to build on work that examines the motive content of leaders or company visions (Kirkpatrick et al., 2002). Company reports or CEO statements and speeches could form the textual data to start from (see also Table 1.1). TM can be used to examine patterns found in data at a single point in time (i.e., cross-sectional), or to investigate changes in patterns over time (i.e., longitudinal) (J. Hu et al., 2015; J. Lee & Hong, 2013). An example of the latter is analyzing business model evolution from annual reports (Lee & Hong, 2013). Another application is to analyze open-ended survey responses as was done by Roberts et al. (2014) who presented two illustrations in their introduction of structural topic models. In one illustration they examined how political affiliation influences views on immigration. In another, they analyzed free text containing players' description of their strategies and related them to their game contributions. Theeboom, van Vianen, Beersma, Zwitser and Kobayashi (2017) applied TM to explore how coaching criteria differ according to length of coaching experience and whether a coach has or does not have a psychology background by analyzing coaches' responses to the question of what are the indicators of successful coaching.

Ultimately, the research question will dictate whether the use of TM is appropriate, and if so the type of text data needed, and the choice of TM technique. Researchers can draw inspiration from existing studies to decide which technique is most suited to reach their specific objective. For instance, in choosing which technique can help identify leadership themes in a corpus of company's mission and vision statements, it could be useful to examine the technique applied to organize news into news themes (Radev, Otterbacher, Winkel, & Blair-Goldensohn, 2005). Table 1.1 provides a summary of the wide range of questions to which TM can be applied. It contains brief descriptions of TM techniques along with the specific questions they are designed to answer and includes existing and potential applications of each TM technique.

Table 1.1. Summary of questions that text mining can address.

Question	Name	Definition	Specific Techniques	Example	Text Representation	Examples of Potential Applications in Organizational Research
How do I assign text to predefined categories?	Text Classification	Using an initial set of labeled text, train a classifier that can automatically sort text into existing categories.	Classification algorithms from Data Mining such as Naive Bayes, Support Vector Machines, Neural Networks, Nearest Neighbors, Random Forest, and Boosting	• Distinguishing between positive and negative product reviews (Dave, Lawrence, & Pennock, 2003; Popescu & Etzioni, 2007). • Subjective genre classification of product reviews (M. Hu & Liu, 2004; Pang & Lee, 2008). • Assigning semantic attributes to product descriptions (Ghani, Probst, Liu, Krema, & Fano, 2006). • Annotating clinical documents with semantic tags (Jang, Song, & Myaeng, 2006).	Vector space model (i.e., individual terms are used as features), Kernel based methods such as Support Vector Machines deal with text treated as strings, Can use other types of features but text is still represented as vectors.	Predicting performance and charisma using leaders' collected speeches and biographies (House et al., 1991).
How do I extract topics from a corpus of documents?	Topic modelling	Identify patterns in word frequencies and use the patterns as a basis to define "topics". For each document possible topics are determined.	Latent Dirichlet Allocation Model and Probabilistic Latent Semantic Analysis	• Topic modeling to extract latent evidence during the analysis phase of digital forensic investigations (Waal, Venter, & Barnard, 2008). • Topic models to enhance the feature set for scientific titles classification (Vo & Ock, 2015).	Vector space model where the words are weighted by their frequencies	Analyzing underlying motives or leadership themes from coded interview data, formal vision statements, and company mission or vision statements (Kirkpatrick et al., 2002)
How can I form groups of text?	Text Clustering	Define a concept of text similarity. Use the concept to group documents together. Each group is called a cluster. Documents in the same cluster are more similar than documents in different clusters	K-means, Hierarchical Clustering, Biclustering, and Nonnegative Matrix Factorization	• Clustering clinical trial records to narrow down search results about existing protocols (Korkontzelos, Mu, Restificar, & Ananiadou, 2011). • Organizing collections of legal documents and assisting automatic generation of legal taxonomies (Conrad, Al-Kofahi, Zhao, & Karypis, 2005).	Vector space model, Can use other types of features but text is still represented as vectors.	Investigating patterns of communication between different parties through the analysis of emails or other virtual communication between employees within firms (Holton, 2009; Nenkova & Bagga, 2003).

How can I summarize text and extract keywords and key sentences?	Text summarization	Measuring the importance of each sentence (word) in a document and using a threshold to determine which sentences (words) to retain and which to delete.	Content Selection using pattern matching, Hidden Markov Models, and keyword or key phrase extraction	• Biographical Summarization (Saggion & Gaizauskas, 2005). • Summarizing web page content for display in small screens of handheld devices (Buyukkokten, Garcia-Molina, & Paepcke, 2001). • Keyword extraction in publications to narrow down and organize query results to support systematic reviews (Ananiadou, Rea, Okazaki, Procter, & Thomas, 2009).	Vector space model; Text is treated in terms of strings.	Automatic summarization of companies' code of conduct to gain greater understanding of what is or is not currently included in organizational policy on ethical behavior in the workplace.
How can I analyze trends in text?	Keyword Extraction over time, Dynamic Topic Modelling, and Clustering with temporal information	Find interesting terms or topics and analyze changes of usage or prevalence in documents indexed by time.	Most frequent term extraction, and Dynamic Topic Modelling	• Analyze trends in SMS messages by tracking the use of specific keywords (Jonsson, Nugues, Bach, & Gunnarsson, 2010). • Analyzing information in software repositories to model software project progress (Hu, Sun, Lo, & Li, 2015). • Tracing significant historical trends in the field of Cognition (Cohen Priva & Austerweil, 2015).	Text is treated in terms of strings; Vector space model.	Analyzing changes in emphasis in companies' code of conduct in reaction to specific events Identifying emergent skills in a corpus of job vacancies (Smith & Ali, 2014).
How can I find other documents which are similar to the one I have?	Distance and Similarity	Given a document find other similar documents	Distance metrics, and similarity measures.	• Information retrieval (Frakes & Baeza-Yates, 1992). • Input to clustering (Jain, Murty, & Flynn, 1999).	Vector space model.	Analyzing conversation between people to facilitate exchange of rewarding information or detection of dangerous activities (Wang & Chen, 2008).
After transforming documents using the vector space model how can I cope with many variables?	Dimensionality reduction techniques	Reduce the number of variables while preserving relative similarity among documents	Feature selection techniques based on thresholding (e.g., information gain) and feature transformation techniques such as Principal Component Analysis, Latent Semantic Analysis, and Random Projection.	• Most dimensionality reduction techniques promote computational efficiency and a more compact representation of text data (Bingham & Mannila, 2001; Chen, Huang, Tian, & Qu, 2009; Forman, 2003). They have been applied to improve both classification performance and classification interpretability.	Vector space model.	The output from this can be used in other techniques such as in classification or clustering.

Key Steps in Text Mining Research

TM generally entails three steps, namely, 1) text preprocessing, 2) application of TM operations, and 3) postprocessing (Zhang, Chen, & Liu, 2015). Figure 1.1 provides a diagram of the different steps in the TM process and the steps, along which our discussion is organized. Text preprocessing may be further subdivided into text data cleaning and text data transformation (e.g., converting unstructured text into mathematical structures which can serve as input to various TM operations). TM operations refer to the application of pattern mining algorithms with the overriding goal to model characteristics of text. Finally, postprocessing involves interpreting and validating knowledge derived from TM operations. Below we explain each of the steps in detail. Hereafter, all mentions of the word "data" refer to text data. Moreover, we use the words "document" and "text" interchangeably. The appendix provides a glossary of key terms.

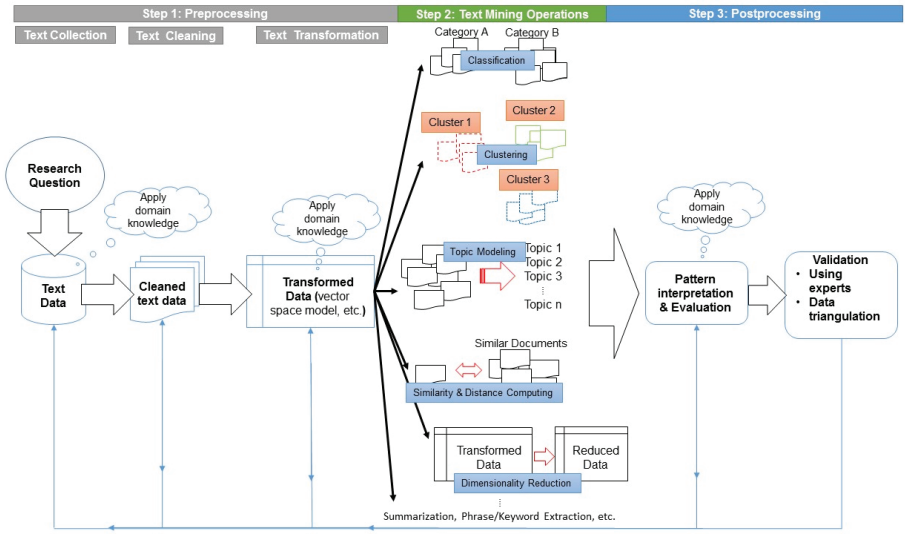


Figure 1.1. Flowchart of the Text Mining process.

Text Preprocessing

Text Data Collection. Before initiating the TM process one should have text data and the first step in data collection is to decide on the most suitable data source(s). Potential sources include the Web, enterprise documents (e.g., memos, reports, and hiring offers), personal text (e.g., diaries, e-mails, SMS messages, and tweets) (Inmon & Nesavich, 2007) and open ended survey responses. TM requires that text must be in digital form or that it can be transcribed to this form. In case it is not, non-digital text (e.g., hand-written or printed documents) may be

digitalized using Optical Character Recognition (OCR) techniques (Borovikov, 2014). Web text data are collected from websites either through web Application Programming Interfaces (APIs) or web scraping (i.e., automatic extraction of web page content) (Olston & Najork, 2010).

It is important to be aware of legal and ethical issues associated with data access, particularly web scraping. Website contents are often protected by copyright law and lawsuits may ensue if agreement under fair use is violated (see for instance "AP, Meltwater settle copyright dispute," 2013). Also, privacy issues may preclude the use of certain types of personal text data without permission, such as web forms, surveys, emails, and performance appraisals (Van Wel & Royakkers, 2004). Another potential issue to be aware of is that of data storage. In small projects, collected text data can be temporarily stored in a local file system (e.g., on a computer). However, in large scale text analytics, especially when the data come from different sources, merging, storing and managing data may require integrated database systems or data warehouses (Inmon, 1996).

Practical Recommendations. As APIs provide an efficient and legal means of obtaining data from the web, researchers who use text from the web first need to find out whether the target website offers an API. One way to find APIs is to use a search platform for APIs such as the ProgrammableWeb (ProgrammableWeb, n.d.). When an API is not available, the next option is web scraping. R has libraries that can automate the web scraping process such as "rvest". Other useful packages include "RCurl" (for http requests), "XML" (for parsing HTML and XML documents), and "stringi" (for text manipulation). If web scraping is not allowed, researchers should ask data owners if they are willing to share their data through remote connections to their databases. Text documents in databases may be fetched using standard query language (SQL).

Text Data Cleaning. Data cleaning enhances data quality which in turn enhances the validity of extracted patterns and relationships. Cleaning is done by retaining only the relevant text elements (Palmer, 2010). Standard cleaning procedures for text include deletion of unimportant characters (e.g., extra whitespaces, formatting tags, etc.), "text segmentation", "lowercase conversion", "stop word removal", and "word stemming". For open-ended survey responses (and other informally produced texts such as sms texts or personal e-mails), in our experience it may be useful to run a spelling check to correct misspelt words. For web documents, HTML or XML tags must be removed since these do not add meaningful content. Thus, the end result is text data stripped of all low content

words and characters.

Text segmentation (Huang & Zhang, 2009) is the process of dividing text into sentences and words. Stop words such as conjunctions and prepositions (e.g., and, the, of, for) are words that have low information content and do not contribute much to the meaning in the text. "Stemming" homogenizes the representation of semantically similar words (e.g., representing the words "ensures", "ensuring", and "ensured" by "ensure"). Since these techniques delete words, they also serve to reduce the size of the vocabulary.

Practical Recommendations. A popular stemming algorithm was developed by Porter (Porter, 1980; Willett, 2006). Most of the other procedures can be performed by applying string processing. For example in R, the Text Processing part *(R Programming/Text Processing, 2014)* of the R Programming wiki provides information on how to implement text processing procedures. The "tm" library, the core framework for TM in R, has functions for stop word removal and stemming. The website (RANKS NL, n.d.) provides lists of stop words for many human languages. There are cases where it is not appropriate to apply stop word removal and stemming, for example, in short text classification (Faguo et al., 2010).

An example output after applying lower case transformation, stop word removal, stemming, and punctuation removal can be found in Table 1.2(a). Instead of just deleting "/", it is replaced by a whitespace otherwise "send/receive" would be merged into the single word string "sendreceive". Extra whitespaces resulting from deleting characters or words are removed.

Text Transformation. Text transformation is a quantification strategy in which text is transformed into mathematical structures. Most analytical techniques require text to be transformed into a matrix structure, where the columns are the variables (also referred to as features) and the rows are the documents. One way to construct this matrix is to use the words or terms in the vocabulary as variables. The resulting matrix is called a "document-by-term matrix" in which the values of the variables are the "weights" of the words in that document. In many applications, this is a straightforward choice since words are the basic linguistic units that express meaning. The raw frequency of a word is the count of that word in a document. Thus in this transformation, each document is transformed into a "vector", the size of which is equal to the size of the vocabulary, with each element representing the weight of a particular term in that document (Scott & Matwin, 1999a).

Table 1.2. (a) An illustration of text preprocessing applied to original text

and (b) Document-by-term matrix constructed from the first six texts of (a)
(a)

	ORIGINAL TEXT	PROCESSED TEXT
D1	Ability or Experience in reviewing and authoring Aircraft Flight Manuals, Apps Spec and Pilot's Guides.	abil experi review author aircraft flight manual app spec pilot guid
D2	Work with KEMP Management to gain approval for new product concepts/ideas.	work kemp manag gain approv product concept idea
D3	Handle client queries and/or requests.	handl client queri request
D4	3-5 years of supervisory or product management experience required.	3-5 year supervisori product manag experi requir
D5	Understanding of XML, parsing, send/ receive and experience with web services.	understand xml pars send receiv experi web servic
D6	Responsible for developing and maintaining quality management procedures and systems.	response develop maintain quality manag procedur system
	ADDITIONAL TEXT	
D7	Experience with J2EE technology components (e.g., JSP, Servlets, XML, and Web Services) is a requirement.	
D8	Minimum 5 years of experience in Marketing or Product Management roles.	
D9	Handling consultant and client queries.	
D10	Define customer applications for the product and design product positioning to support these applications.	
D11	3-5 years of experience in the Engineering and/or Maintenance field strongly preferred.	

Note: The 6 preprocessed texts were obtained after applying stop word removal, stemming, and punctuation removal except for intra-word dashes and stripping extra whitespaces
(b)

	3-5	Abil	aircraft	app	approv	author	client	experi	manag	product
D1	0	1	1	1	0	1	0	1	0	0
D2	0	0	0	0	1	0	0	0	1	1
D3	0	0	0	0	0	0	1	0	0	0
D4	1	0	0	0	0	0	0	1	1	1
D5	0	0	0	0	0	0	0	1	0	0
D6	0	0	0	0	0	0	0	0	1	0

Note: This table is truncated due to space limitation.

Word frequency, in itself, may not be useful if the task is to make groupings or categories of documents (Kobayashi, Mol, Berkers, Kismihok, et al., 2017). Consider the word "study" in a corpus of abstracts of scientific articles. If the objective is to categorize the articles into topics or research themes, then this word is not informative as in this particular context almost all documents contain this word. A way to prevent the inclusion of terms that possess little discriminatory power is to assign weights to each word with respect to their specificity to some documents in a corpus (Lan et al., 2009). The most commonly used weighting procedure for this is the Inverse Document Frequency" *(IDF)* (Salton & Buckley, 1988). A term is not important in the discrimination process if its *IDF* is 0, implying that the word is present in every document. In fact, *IDF* can also be the basis to select stop words for the categorization task at hand. Words that have a low IDF have little discriminatory power and can be discarded. When multiplied, the word raw frequency (tf) and *IDF* yield the popular *TF-IDF*, which simultaneously takes into account the importance of a word and its specificity (Frakes & Baeza-Yates, 1992a).

Representing text as a document-by-term matrix presupposes that word order information is not crucial in the analysis. Although unsophisticated, it is noteworthy that transformations that ignore word order information perform better in many applications than transformations that account for it (Song, Liu, & Yang, 2005; Zhang, Yoshida, & Tang, 2008). The main computational challenge for document-by-term matrix representation is how to deal with the resulting dimensionality which is directly proportional to the size of the vocabulary. One can use different data dimensionality reduction methods to reduce the number of variables (e.g., variable selection and variable projection techniques) or employ specific techniques suited for data with high dimensionality. These techniques will be highlighted in the Text Mining Operations section.

Once text is transformed, techniques such as regression analysis and cluster analysis can be applied. Combining variables helps tackle substantive questions about the text (see also the Text Mining Operations section). For instance, if resumes of job applicants are used as a data source then the presence of the words "experience" and "year" together with a number can be used to deduce an applicant's work experience.

Practical Recommendations. The output from word segmentation provides the vocabulary. One may start by creating a Document-by-Term matrix. In R, the "tm" library has a function that can generate a document-by-term matrix, with an additional option for specifying weights. For example, consider the six preprocessed texts in Table 1.2(a). Part of the document-by-term matrix constructed from the

texts using raw frequency weighting is shown in Table 1.2(b). The complete matrix has 40 columns, equal to the number of unique words found in the 6 texts.

Text Mining Operations

Though text transformation precedes the application of analytical methods, these two steps are closely intertwined. The document-by-term matrix from the text transformation step serves as the input data for most of the procedures in this section. Sometimes, when results are unsatisfactory, the researcher may consider changing or enlarging the set of variables derived from the transformation step (Lewis, 1992a; Scott & Matwin, 1999a) or choosing another analytical method. Usually, different combinations of data transformation and analytical techniques are tried and tested and the one that yields the highest performance is selected.

Most TM operations fall into one of five types, namely, 1) dimensionality reduction; 2) distance and similarity computing; 3) clustering; 4) topic modeling; and 5) classification (Solka, 2008). Below we discuss each technique prior to discussing how to assess the credibility and validity of TM outcomes.

Dimensionality Reduction. Document-by-term matrices tend to have many variables. It is usually desirable to reduce the size of these matrices by applying dimensionality reduction techniques. Some of the benefits of reducing dimensionality are more tractable analysis, greater interpretability of results (e.g., it is easier to interpret variable relationships when there are few of them), and more efficient representation. Compared to working with the initial document-by-term matrices, dimensionality reduction may also reveal latent dimensions and yield improved performance (Bingham & Mannila, 2001). Two general approaches are commonly used to reduce dimensionality. One is to construct new latent variables and the second is to eliminate irrelevant variables. New variables are modelled as a (non)linear combination of the original variables and may be interpreted as latent constructs (e.g., the words "years", "experience", and "required" may be merged to express the concept of a "work experience" in job vacancies).

Singular Value Decomposition (SVD) is a classic tool which underlies techniques such as Latent Semantic Analysis (Landauer et al., 1998b) and Principal Components Analysis (PCA) (Jolliffe, 2005). The SVD method decomposes a matrix (X of the size $p \times n$) (where P is the number of variables and n is the number of documents) into a product of three matrices, i.e., $X=U\Sigma V^T$. One of these is a diagonal square matrix (Σ) which contains the singular values (Klema & Laub, 1980). Reducing the number of dimensions involves retaining the first few largest singular values. Usually, this implies choosing latent dimensions and recovering

the underlying dimensionality of the data because at times, true dimensionality is obscured by random noise.

Latent Semantic Analysis (LSA) is commonly used to detect synonymy (i.e., different words that have the same meaning) and polysemy (i.e., one word used in different yet related senses) among words. PCA is effective for data reduction as it preserves the variance of the data. Parallel analysis (Ford et al., 1986; Hayton et al., 2004; Montanelli Jr & Humphreys, 1976) is the recommended strategy to choose how many dimensions to retain in PCA. A disadvantage of both LSA and PCA is that it may be difficult to attach meaning to the constructed dimensions. Another technique is Random Projection where data points are projected to a lower dimension while maintaining the distances among points (Bingham & Mannila, 2001).

An alternative approach to reduce dimensionality is to eliminate variables by using variable selection methods (Guyon & Elisseeff, 2003). In contrast to projection methods, variable selection methods do not create new variables but rather select from the existing variables by eliminating those that are uninformative or redundant (e.g., words that occur in too many documents might not be useful for categorizing documents). Three types of methods are available: filters, wrappers, and embedded methods. Filters assign scores to variables and apply a threshold to scores in order to delete irrelevant variables. Popular filters are TF-IDF thresholding, Information Gain, and the Chi-squared statistic (Forman, 2003; Yang & Pedersen, 1997a). Wrappers select the best subset of variables in conjunction with an analytical method. In embedded methods, searching the best subset of variables is accomplished by minimizing an objective function that simultaneously takes into account model performance and complexity. Model performance can be measured for example by prediction error (in the case of classification) and complexity is quantified by the number of variables in the model. The smallest subset of variables yielding the lowest prediction error is the preferred subset.

Practical Recommendations. The dimensionality reduction stage is usually initiated by applying LSA. The LSA results (i.e., the reduced data set) can be used as input to clustering and classification. Alternatively, one can apply one of the filter methods to trim out unimportant variables. One advantage of filters as compared to LSA is interpretability since no new variables are constructed. Moreover, filter methods are faster to run. The R package "lsa" provides functionality for running LSA.

Consider the 11 texts in Table 1.2(a), which after data cleaning, are transformed into a document-by-term matrix. Running LSA on the transpose

of the document-by-term-matrix, we retained 2 dimensions. The resulting LSA space represented as a matrix is presented in Table 1.3(a). Observe that the value for "product" in Document 11 is 0.32 although Document 11 does not contain the word "product". This is due to the presence of the word "experience" in the other two documents (Document 4 and Document 8) that also contain "product". Since "experience" is present in Document 11, LSA expects to find "product" in this document. This is how LSA deduces meaning from words (which is also useful for the identification of synonyms).

Distance and Similarity Computing. Assessing the similarity of two or more documents is a key activity in many applications such as in document retrieval (e.g., document matching), and recommendation systems (e.g., for finding similar products based on product descriptions or reviews). Numerous measures that operate on vector representations may be employed to assess distance or similarity. An example of the latter is the cosine measure that is used extensively in information retrieval (Frakes & Baeza-Yates, 1992a). The values for this measure range from -1 (two vectors point in opposite direction) to 1 (two vectors point in the same direction); 0 means that the two vectors are orthogonal or perpendicular (or uncorrelated). This measure assesses the similarity of two documents based on the frequencies of terms they share, which are taken to indicate similarity of content. This measure has been applied to document matching (Frakes & Baeza-Yates, 1992a) and detecting semantic similarity (Mihalcea et al., 2006).

Of the various distance measures, Euclidean and Hamming distance measures are the most commonly employed. Unlike similarity measures, higher values for distance measures implies dissimilarity. Also, distance measures have to satisfy certain properties such as non-negativity and triangle inequality. In most cases similarity measures can be converted to distance measures and vice versa.

Clustering. Many tasks in TM involve organizing text in groups such that documents belonging to the same group are similar and documents from different groups are not (Jain et al., 1999; Steinbach et al., 2000). The process of grouping is called *clustering*. The main use of text clustering is either to organize documents to facilitate search and retrieval or to impose an automatic categorization of documents. For example, text clustering has been used to detect crime patterns (e.g., location, type of crime, weapons) in crime reports (Bsoul et al., 2013), to organize and deepen the taxonomy of legal practice areas (Conrad et al., 2005), and to improve the performance of a document retrieval system or web-based search engine by creating a taxonomy of documents and grouping the search query

results (Osinski & Weiss, 2005). In order to perform text clustering, the researcher needs to define distance between texts (e.g., Euclidean distance). The distance measure can be computed from the original set of variables or from the reduced set of variables (e.g., after application of dimensionality reduction techniques such as LSA).

Most clustering algorithms are categorized as either hierarchical or partitional (Steinbach et al., 2000). Hierarchical clustering algorithms either treat each object as its own cluster and then gradually merge clusters until all objects belong to a single cluster (i.e., agglomerative) or by first putting all objects under one cluster and recursively splitting clusters until each object is in its own cluster (i.e., divisive). The merging (or splitting) of clusters is depicted by a tree or dendrogram. For partitional clustering the user has to specify the number of clusters a priori and clusters are formed by optimizing an objective function that is usually based on the distances of the objects to the centers of the clusters to which they have been assigned. The popular k-means algorithm is an example of partitional clustering (Derpanis, 2006). One key challenge in clustering is the determination of how many clusters to form. Since clustering is an exploratory technique, a common strategy is to experiment with different numbers of clusters and use cluster evaluation measures to decide. Examples of quality measures are the Dunn index and the Silhouette coefficient (Rendón et al., 2011).

Practical Recommendations. One can start with k-means or a hierarchical approach such as the complete linkage or Ward's method (El-Hamdouchi & Willett, 1989). If a researcher has a clear idea of how many clusters to create, then k-means is a good start. If a researcher has no idea as to how many clusters to construct, then she may use hierarchical clustering to see whether interpretable groupings emerge. The "cluster" and "mclust" packages in R run most of the clustering techniques described here and the "proxy" package offer various distance and similarity measures.

Using the reduced dimension set from LSA and a distance metric derived from cosine similarity we applied hierarchical clustering based on Ward's method on the 11 texts (see Table 1.2(a)). The resulting dendrogram is shown in Figure 1.2. The dendrogram basically shows two clusters: one cluster is about customer and product management and the other pertains to technical requirements on technology use.

Topic Models. Topic models automatically extract topics from documents. These topics can indicate underlying constructs or themes. In machine learning

and natural language processing, topic models are probabilistic models that are used to discover topics by examining the pattern of term frequencies (Blei et al., 2003). Its mathematical formulation has two premises: a topic is characterized by a distribution of terms and each document contains a mixture of different topics. The most likely topic of a document is therefore determined by its terms. For example, when an open-ended survey response, contains words such as "pay", "compensation", "salary", and "incentive", one might label its topic as "rewards or pay systems".

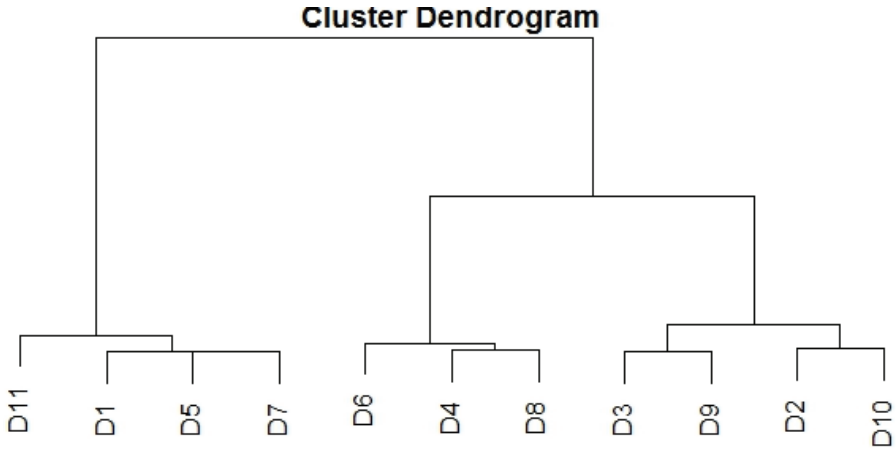


Figure 1.2. Cluster dendrogram of 11 texts

Perhaps the most popular topic models are the Latent Dirichlet Allocation (LDA) (Blei et al., 2003) model and the Correlated Topic Model (CTM) (Blei & Lafferty, 2007). LDA and CTM both operate on the document-by-term matrix (Porteous et al., 2008). CTM will yield almost the same topics as LDA. The main difference between the two is that in LDA topics are assumed to be uncorrelated, whereas in CTM topics can be correlated. In comparing LSA with LDA, the latter has been found to be particularly suitable for documents containing multiple topics (Lee, Baker, Song, & Wetherbe, 2010).

Table 1.3. (a) Lower rank approximation of the term-by-document matrix obtained from Table 1.2 using LSA by retaining 2 dimensions and (b) Sample topics extracted from the 11 texts for LDA and CTM.
(a)

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11
3-5	0.23	0.16	0.00	0.35	0.26	0.07	0.34	0.31	0.00	0.22	0.25
abil	0.15	-0.01	0.00	0.11	0.18	0.01	0.23	0.08	0.00	-0.1	0.12
aircraft	0.15	-0.01	0.00	0.11	0.18	0.01	0.23	0.08	0.00	-0.1	0.12
experi	0.88	0.22	0.00	0.94	1.02	0.15	1.3	0.78	0.00	0.05	0.81
product	-0.03	0.95	0.00	0.94	-0.08	0.28	-0.07	0.95	0.00	1.97	0.32
web	0.41	-0.05	0.00	0.28	0.49	0.02	0.62	0.21	0.00	-0.3	0.32
work	-0.01	0.13	0.00	0.12	-0.03	0.04	-0.03	0.13	0.00	0.28	0.04
xml	0.41	-0.05	0.00	0.28	0.49	0.02	0.62	0.21	0.00	-0.3	0.32
year	0.31	0.29	0.00	0.55	0.35	0.11	0.46	0.49	0.00	0.45	0.37

Note: This table is truncated

(b)

	Latent Dirichlet Allocation			Correlated Topic Model		
	Topic 1	Topic 2	Topic 3	Topic 1	Topic 2	Topic 3
Terms	product manag applic experi year	abil aircraft app approv author	experi client handl queri servic	experi manag year product 3-5	abil aircraft app author develop	product applic handl queri client
Documents	4,6,8,10	1,2	3,5,7,9,11	2,4,7,8,11	1,6	3,5,9,10

Practical Recommendations. For topic extraction, the recommended initial approach is to try LDA. It may also be useful to investigate the assignment of documents to topics. Code to run Topic Models is available in the "topicmodels" package in R. For example, we ran LDA and CTM on the example text above in Table 1.2(a) (see Table 1.3(b) for the results). The top terms listed in each topic form the basis for topic interpretation. For example, the top terms of Topic 1 indicate that this topic is about product management, whereas Topic 2 is more about ability on aircraft apps, and Topic 3 about the handling of clients. Examining the most likely topic for each document we observe that Documents 3, 9, and 11 have Topic 3 as the most likely topic since these documents are focused on dealing with customers.

Classification. Classification is the assignment of objects to predefined classes or categories. Logistic regression is perhaps the best known classification method. The goal is to construct a model that can predict the category of a given document. Example applications of text classification are spam or ham classification of emails (Youn & McLeod, 2007), authorship identification (Houvardas & Stamatatos, 2006), thematic categorization (Phan et al., 2008), and identification of sentiments in product reviews (Dave et al., 2003a; M. Hu & Liu, 2004a; Pang & Lee, 2008; Popescu & Etzioni, 2007). For a fuller discussion and tutorial of text classification we refer the reader to Kobayashi et al. (2017).

Evaluation. Model evaluation helps us choose which among competing models best explains the data (Alpaydin, 2014). Model evaluation needs to address issues related to underfitting and overfitting. Underfitting happens when the model does not adequately represent the relationships present in the data (i.e., high variance). Overfitting occurs when a model performs well on data used to build it but poorly on new data (i.e., high bias). Hence, a model generalizes well if it also demonstrates good performance on new data (Mitchell, 1997). A common way to assess the quality of the model's generalizability is to use hold-out data (Alpaydin, 2014). The procedure involves repeatedly splitting the corpus of documents to create a training and a test set either by randomly sampling documents from the corpus or by partitioning the corpus. Documents in the training set are used to fit the model and the generalizability of this model is assessed using the documents in the test set. Procedures that evaluate a model by partitioning the corpus are K-fold cross validation and a resampling procedure called bootstrapping (Kohavi, 1995). Measures to assess generalizability are commonly referred to as evaluation metrics. Since different values of the metric for each unique split will be obtained, values are usually averaged across splits. Using cross-validation and bootstrapping, one can build confidence intervals and assess the true performance of the model. The choice of metric is dependent on the task and application domain. However, it should be kept in mind that conclusions generated are conditioned on the data; that is, a model is good only insofar as the data are representative of the population. Second, there are other criteria to judge the merit of a model, such as the time it takes to build the model and its interpretability.

Practical Recommendations. In topic modelling, one can use the aggregate topic probabilities of unseen documents (Wallach et al., 2009) as an evaluation metric. In clustering, internal and external evaluation criteria are used. External criteria use previous knowledge about the data (i.e., prior information) and internal

criteria only use the data. We already mentioned two criteria in the Clustering section which are the Dunn's index and the silhouette coefficient (Rendón et al., 2011).

Both dimensionality reduction and distance and similarity computing are usually evaluated on their impact on the text classification and text clustering performance (Forman, 2003). That is, an effective dimensionality reduction technique must contribute to the improvement of classification or clustering performance. An analogous comment can be made for distance and similarity computing, since these measures often serve as input to the clustering (e.g., k-means) and classification task (e.g., nearest neighbor), although there are applications where distance and similarity measures are used as a standalone method (Houvardas & Stamatatos, 2006; Lewis, 1992b; Mihalcea et al., 2006). An example of the latter is comparing (parts of) leader and subordinate resumes to operationalize Person-Supervisor Similarity. In information retrieval, where the task is to match queries to document content, performance metrics for distance or similarity measures are precision, recall, and the F-measure.

Postprocessing

The postprocessing step may involve domain experts to assist in determining how the output of the models can be used to improve existing processes, theory, and/or frameworks. Two major issues are usually addressed here. The first is to find out whether the extracted patterns are real and not just random occurrences due to the sheer size of the data (e.g., by applying Bonferroni's principle). The second is, as with all empirical research, whether data and results are valid. Establishing the reliability, validity (e.g., content, predictive, and discriminant validity), and credibility of the output of TM models is particularly important for TM to gain legitimacy in organizational research. It is important to note here that it is not the TM procedures that need to be validated but the output (in the same manner that we do not validate Factor Analysis), for example, the predictions of a TM based classifier.

Prior to being applied to support decision making and knowledge generation, the validity of TM based findings will need to be established. When TM is used to identify and operationalize constructs, using different forms of data triangulation will help generate construct validity evidence. For example, in our job analysis example of TM application, which follows below, we enlisted the help of job analysts and subject matter experts in evaluating the output of the TM of vacancy texts. In other cases, TM outcomes could be compared to survey

data, such as for the aforementioned study on the role of personality in language use (Yarkoni, 2010). More generally, TM based models will require a comparative evaluation in which (part of) the TM output is correlated with independent data sources or other 'standards' (such as the aforementioned survey or expert data). Though it is easy to view TM as a mechanistic means of extracting information from data, the input of domain experts is critically important. Finally, there is no reason why validity assessment procedures, such as those outlined by Binning and Barrett (1989a) to establish the validity of personnel decisions, cannot be applied to TM output.

Practical Recommendations. A straightforward practice for construct validation is to have independent experts validate TM output. For example, in text classification, subject matter experts (SMEs) may be consulted from time to time to assess whether the resulting classifications of text are correct or not. A high agreement between the experts and the model provides an indication of the content-related validity of the model. The agreement is usually quantified using measures such as the Cohen's kappa or intra-class correlation coefficient.

Another way to validate TM output is through replication, data triangulation, and through an indirect inferential routing (Binning & Barrett, 1989a). The standard can be established by obtaining external data using accepted measures or instruments that may provide theory based operationalizations that should or should not be correlated to the model. Such correlations give an indication of validity. For example, to validate experience requirements extracted from job vacancies, one can administer questionnaires to job incumbents asking them about their experience. Validity is then ascertained through the correlation between both operationalizations. This can be replicated on various types of text to assess if the TM model consistently generates valid experience requirements for a particular occupation. In theory, one could even compute full multi-trait multi-method correlation matrices (D. T. Campbell & Fiske, 1959a) to compare the measurements obtained from TM with established instruments, although in practice it may be difficult to obtain the fully crossed dataset that it requires.

Text Mining Applied to Job Analysis

To illustrate the key steps in TM we provide an example from job analysis. Job analysis aims to collect and analyze any type of job-related information to describe and understand a job in terms of behaviors necessary for performing the

job (Sanchez & Levine, 2012a; Voskuijl, 2005). Job analytic data is traditionally collected through interviews, observations, and surveys among subject matter experts (SMEs), including job holders, supervisors, and job analysts (Morgeson & Dierdorff, 2011). Here, we apply a TM approach to automatically classify job information from vacancies and assess whether the worker attributes necessary for effective job performance emerge from the vacancies to show that TM might be useful tool to job analysts.

Job analysis may be a fertile ground for TM due to the abundance of textual sources of job information, a prime example being online job vacancies. The ability to analyze a big corpus of job vacancies addresses several limitations of existing job analysis data collection strategies. Job vacancies provide up to date job information, offer the potential to capture the dynamism of jobs in the contemporary workplace (Sanchez & Levine, 2012a), and may be used to reduce bias inherent in existing data collection strategies for job analysis (Dierdorff & Morgeson, 2009; Morgeson et al., 2004; Morgeson & Campion, 1997, 2000; Sanchez & Levine, 2000a). Also, job vacancies are inexpensive and relative easy to obtain and since TM techniques are based on algorithms that are optimized for performance, job information extraction can be made efficient and reliable (McEntire et al., 2006).

Previous studies in the field of Information Technology have extracted skills from vacancies by counting the frequency of preselected keywords related to computer programming (e.g., Java, Python, etc.) (Smith & Ali, 2014; Sodhi & Son, 2010). One limitation of this approach is that it may not be effective in detecting emergent skills, because researchers may fail to specify the appropriate keywords. Here we assess whether TM is of use in analyzing the content of vacancies. In line with the work of Sacket and Laczo (2003a), it seems useful to be able to disentangle information referring to worker attributes on the one hand and work activities on the other. Also, worker attribute requirements may differ across job professions and/or job industries. Determining key worker attributes (e.g., technical skills) for specific jobs and how these worker attributes compare and contrast across jobs could be of use in job classification (Harvey, 1986), training needs analysis (Arthur Jr et al., 2003), compensation (Verwaeren et al., 2016), recruitment (Abdessalem & Amdouni, 2011), and the generation of synthetic validity evidence (Scherbaum, 2005).

Classification of Job Information Types

Preprocessing. We partnered with two organizations, namely, Monsterboard and Textkernel, which provided access to vacancy data from various employment websites. Of the different fields that vacancies usually contain, our focus is on the *job description* field which usually lists activities associated with the job and the attributes required from the applicants.

Since our analysis operates at the sentence level and some of our variables are derived from the words, we started by applying sentence and word segmentation. Once words and sentences were identified, we converted letters to lower case and removed stop words. The criteria used to determine whether a word is a stop word or not were based on the standard English language (RANKS NL, n.d.) stop word list and our own inductive identification of words that did not appear to be associated with the types of job information we were interested in detecting. Hence, conjunctions, articles, and prepositions were deleted. We retained the following stop words: "to", "have", "has", "had", "must", "can", "could", "may", "might", "shall", "should", "will", "would", because these were useful for the classification task. Specifically, sentences containing "to" and "will" often contain job activities, whereas, "have", "has", "had", "should", and "must" are suggestive of worker attributes. Having deleted the irrelevant stop words, we removed punctuation except for intra-word dashes to avoid separating words which together express a single meaning (e.g., problem-solving, customer-oriented, pro-active, etc.). Finally, we stripped the extra whitespaces that resulted from the deletion of particular characters. The output was a collection of sentences in which all letters were in lower cases from which the irrelevant stop words and punctuation had been stripped.

For the transformation step we deviated from the approach of using solely words as variables. We generated a list of variables that would potentially be able to predict the category membership of sentences i.e., into either work activities (e.g., tasks) or worker attributes (e.g., skills). We used knowledge from the job analysis field and eye-balling coupled with statistical tests to preselect these variables. Based on definitions of tasks, for example, we deduced that often, these are indicated by sentences that consist of an action verb, the object of the action, the source of information or instruction, and the results (Morgeson & Dierdorff, 2011; Voskuijl, 2005). We also expected verbs to be more prevalent in activity sentences than in attribute sentences. Using part-of-speech tagging, we computed features such as

the percentage of verbs in a sentence and the part-of-speech of the first word. For the POS labels, we based the tags on the "Penn part-of-speech tags" ("Penn Part of Speech Tags," n.d.).

For the purposes of our analysis, we put all noun, verb, adjective, and adverb related tags together. We grouped related tags under one general derived tag, since we did not require detailed information about each tag. For example, singular or mass noun (NN), plural noun (NNs), singular proper noun (NNP), plural proper noun (NNPS) were all subsumed under the "noun" tag. Other noteworthy tags are TO (to), CD (cardinal number) and MD (modal), these tags appeared important for discriminating between work activities and worker attributes. The TO tag is indicative of job activity (e.g., "to ensure project stays on track for assigned client projects"), because it reflects either the results of an action or the indefinite form of a verb in a task. The presence of a CD tag most often points to the years of education or work experience required from job applicants, and hence is indicative of the worker attribute category. The complete list of variables can be found in Table 1.4.

A total of 168 variables were constructed. In the future, in order to create finer (sub)classifications of job information types, additional features will likely be needed. We computed the 168 variables for each sentence (our unit of analysis) and constructed vectors that represent each sentence. We then collected the vectors in a data matrix.

Application of Classification Techniques. The data matrix served as the input data for the classification of job information. In order to construct models, we needed labeled training data. We examined each sentence and employed standard definitions from the job analysis literature (these were accumulated in a coding manual that is available in the supplement) to label each sentence as either a work activity or a worker attribute. In establishing the labeled training data, mixed sentences containing both activity and attribute information were split and buffer sentences not containing any relevant information were dropped. For the construction of the classification model, we added a 169th column to the data matrix. This column contained the classification of sentences into either job attribute (0) or job activity (1) as derived from the manually labeled sentences.

Table 1.4. The 168 variables for the Vacancy Mining task.

Feature	Number of derived features	Variable Type
Part of speech (POS) tag of the first word	1	Categorical (actual POS)
Is the first word in this sentence unique in work activity sentences (based on the labeled data)	1	Numeric
Is the first word in this sentence unique in worker attribute sentences (based on the labeled data)	1	Numeric
Is the last word in this sentence unique in work activity sentences (based on the labeled data)	1	Numeric
Is the last in this sentence unique in worker attribute sentences (based on the labeled data)	1	Numeric
Proportion of adjectives	1	Numeric
Proportion of verbs	1	Numeric
Proportion of word "to"	1	Numeric
Proportion of modal verbs	1	Numeric
Proportion of numbers	1	Numeric
Proportion of adverbs	1	Numeric
Proportion of nouns	1	Numeric
Proportion of nouns, verbs, adjectives, adverbs, and other part of speech tags followed by another verb	5	Numeric
Proportion of unique words found only in work activity sentences (based on the labeled data)	1	Numeric
Proportion of unique words found only in worker attributes sentences (based on the labeled data)	1	Numeric
Frequency of keywords for work activity and worker attributes sentences	149	Numeric

For the classifier, three techniques were tested, namely Naive Bayes (NB), Support Vector Machines (SVM), and Random Forest (RF). We chose these as they are purportedly the most effective classifiers for text classification (Aggarwal & Zhai, 2012a). We built each classifier and assessed its performance through 10-fold cross-validation using accuracy and F-measure as performance metrics. These performance metrics reflect our objective of creating an accurate classifier that favors either one of the categories (attribute or activity).

The parameter set for each technique and the classification results are summarized in Table 1.5. The mean of the two metrics from the 10-fold cross validation suggests that SVM and RF perform better than NB. A comparison of the mean accuracies using a One-Way ANOVA found that at least one mean accuracy was different from the rest [$F(2.27) = 15.94$, $p = .000$]. A post-hoc

analysis using Tukey's Honest Significant Difference (HSD) method revealed that the mean accuracy of NB is significantly different from the other two techniques [p = .001 (RF) and p = .000 (SVM)], whereas SVM and RF did not significantly differ from one another (p = .988). These high accuracies can be explained by the appropriateness of the extracted variables and the suitability of these classifiers for studying text data. To make predictions even more valid, one can aggregate them (e.g., by means of majority voting).

Table 1.5. Parameters and Performance metrics for the three classifiers.

	Parameters	Accuracy (%)	F-measure for Job Activity	F-measure for Job Attribute
Support Vector Machine	Dot product kernel Cost of misclassification = 1 Number of trees grown= 500	97.30	.9703	.9751
Random Forest	Number of variables sampled at each split = 4	97.31	.9700	.9750
Naive Bayes	Laplace smoothing = 0.01	96.60	.9463	.9554

We then ran the classifier on over a million sentences and obtained an additional 270,000 work activity sentences and 317,000 worker attribute sentences. These are the sentences in which all three classifiers agree and have high confidence in their predictions.

Postprocessing. Since it is difficult to find job experts that have expertise across job professions the following discussion of validity is based solely on nursing jobs and experts in those. Specifically, we wanted to assess whether the extracted work activities for nurses correspond to actual nursing tasks. We validated the TM application to job analysis in two ways. First, we asked a nursing expert (i.e., training coordinator) to examine the condensed list of 76 nursing tasks that we extracted from the nursing vacancies for consistency with the actual tasks executed in practice. The 76 nursing tasks were obtained by first extracting task sentences from vacancies, and then applying clustering to group similar tasks together. Hence, we only presented core nursing tasks to the expert.

The subject matter expert (SME) classified 93.3% of the extracted tasks as representative of actual nursing jobs. The expert validation provided initial support for the content validity of the TM model as the collected information from the vacancies appears to accurately reflect the job. Second, we compared

the TM results with traditional job analysis, namely a task inventory, to validate our results by data triangulation. The task inventory consisted of four interviews and a two-day observation with SMEs (i.e., nurses and head nurse) from two German hospitals. More information about the task inventory is available in the supplement). Tasks from both lists were rated as synonyms (i.e., exactly the same), similar (i.e., different wording, same meaning), or dissimilar (i.e., different wording and meaning) based on the decision rules of Tett, Guterman, Bleier and Murphy (2005). Based on this comparison 55.6% of all tasks were found in both lists, whereas 29.1% were unique to the task inventory and 15.2% to the online vacancies. The relatively high correspondence (≥ 50%) between the list of tasks collected by text mining and the list of tasks collected in the task inventory further established convergent validity.

Topic Modeling on Worker Attributes

We now proceed with our second aim of analyzing all of the extracted worker attributes (i.e., not restricted to solely those of the nurses). Our goal is to summarize the worker attributes and find worker attribute constructs and use these to cluster jobs. For this purpose, we applied topic modeling using LDA to the extracted worker attribute sentences. We set the number of topics equal to 140 based on two criteria. One criterion is based on topic distances as discussed in the paper of Cao, Xia, Li, Zhang, & Tang (2009) and the other is based on the idea that LDA is a matrix factorization mechanism and the quality of the factorization depends on choosing the right number of topics (for additional information we refer the reader to the paper of Arun, Suresh, Madhavan, & Murthy (2010)). We use Variational Expectation Maximization to estimate the parameters of the LDA model. For the interest of space and purpose of illustration we show in Table 1.6 a subset of twelve topics generated from LDA. Looking at the top 8 words, Topics 75, 18, 45, 108, and 129 appear to point to behavioral/personal qualities. Topic 75 could be interpreted as interpersonal communication skills, Topic 18 as self-motivation, Topic 45 seems to pertain to attention to detail, Topic 108 seems to be about analytical and problem-solving skills, and Topic 129 about team-working. Topics 132 and 16 are attributes that were seldom considered in job analysis studies (e.g., Harvey (1986)) and may as well reflect new worker attributes sought by contemporary organizations. Topic 132 seems to be about willingness to travel and the ability to operate on a flexible work schedule and Topic 16 about data analytical skills. The rest of the Topics seem to be about technical skills specific to certain professions such as sales for Topic 20 and software/programming for Topics 100 and 60. Topic 61 pertains to a specific requirement and is about having

a valid driving license. Interestingly, even without giving LDA prior information about which worker attributes to expect it still appears to recover both technical and soft skill requirements. Though it is a bit difficult to interpret Topics 86, 105 and 15 they seem to be topics pertaining to generic personal qualities such as the ability to learn new things quickly (86), goal-setting and leadership (105), and possessing a positive, energetic, and enthusiastic attitude (15). We can visualize the correlations among words within each topic to aid interpretation. We show this in Figure 1.3 where an edge between words indicates a correlation of at least 0.1 and the thickness of an edge indicates the strength of correlation. The word-networks are in line with our interpretations and show that a topic could capture more than 2 worker attributes; the model put them in one topic because they tend to co-occur. From the topics we can generate hypotheses about which behavioral/personal characteristics are actually required to carry out a particular job, which could then be tested in an empirical study.

Table 1.6. Some of the topics obtained from applying LDA to worker attribute sentences.

TOPIC 100	**TOPIC 86**	**TOPIC 132**	**TOPIC 75**
development	new	travel	communication
software	learn	willingness	written
agile	quickly	willing	oral
methodologies	willingness	work	verbal
application	adapt	time	interpersonal
scrum	technologies	needed	presentation
design	internet	internationally	effective
life	desire	international	listening
TOPIC 18	**TOPIC 45**	**TOPIC 20**	**TOPIC 105**
highly	detail	sales	results
motivated	attention	selling	leadership
oriented	oriented	salesforcecom	others
self	organizational	outside	goals
driven	accuracy	crm	achieve
organized	multitask	success	influence
starter	follow	account	motivate
selfstarter	details	inside	deliver
TOPIC 60	**TOPIC 15**	**TOPIC 55**	**TOPIC 108**
scripting	attitude	design	problem
python	positive	adobe	solving
linux	can	creative	analytical
programming	energetic	photoshop	solver
java	team	user	troubleshooting
perl	flexible	illustrator	approach
languages	enthusiastic	graphic	abilities
unix	professional	production	capabilities
TOPIC 61	**TOPIC 129**	**TOPIC 81**	**TOPIC 16**
license	work	management	data
valid	team	time	analysis
drivers	independently	project	quantitative
driving	part	organizational	research
record	environment	change	statistics
transportation	pressure	people	economics
reliable	members	planning	statistical
vehicle	member	pm	modeling

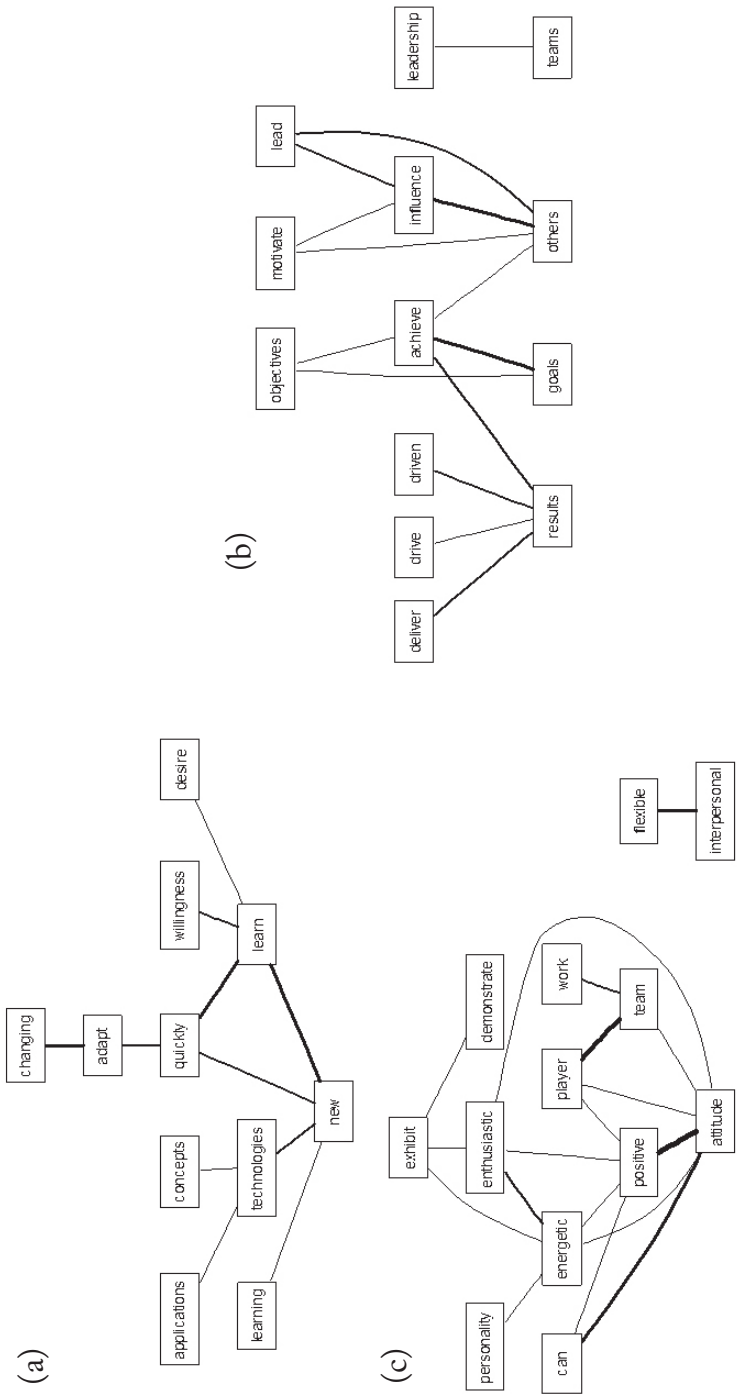


Figure 1.3. Word correlation networks for (a) Topic 86, (b) Topic 105, and (c) Topic 15.

Investigating the relationship between topics provides a way to assess the convergent/divergent validity of the topical content. Here we cannot directly use correlation since topics are assumed to be uncorrelated, however, we can use the "distance" between topics. To get a better idea about how to judge whether an association is low or high, we suggest using simulation techniques such as Monte Carlo or permutation tests. In this case, magnitude is always application dependent. In order to compute distance, we use the Jensen-Shannon divergence which measures the distance between probability distributions. Here we focus the discussion on Topic 75 which we previously interpreted as interpersonal communication skills. Topic 75 is closer to Topics 13, 30, 51, 88, 111, 129, and 103 (please refer to the supplement for the complete list of topics). Topics 13 (effective oral and written communication), 30 (professional demeanor), 129 (teamwork), and 103 (analytical and problem solving skills) all relate to interpersonal skills hence these qualities are expected to relate to interpersonal communication. A noteworthy similarity exists between Topics 132, 77 and 119 which are willingness to travel, ability to work on a flexible schedule, and work relocation, respectively. We can further explore this relationship by performing a more inference driven investigation by comparing the findings here to the results obtained by interviewing SMEs or job holders, which will further help in establishing construct validity. Aside from similar topics there are also less similar ones, for example Topic 75 (interpersonal communication skills) is least similar to Topics 31 (finance) and 89 (programming languages). Possible interpretations include range restriction (that is, if job incumbents in a position do not vary on certain characteristics these characteristics may not be mentioned in the vacancies), but it could also mean that interpersonal communication is not essential to perform jobs requiring those specific technical skills, or that incumbents who excel in those jobs have low interpersonal communication skills.

In order to examine the relationship among all topics simultaneously, we applied multidimensional scaling and projected the topics in 2 dimensions. Figure 1.4(a) shows the projections of topics on 2 dimensions. Topics 7, 8, 9, 6, 25, and 35 (bottom rightmost, 4th quadrant) are close together because they all relate to programming or software skills. This also holds for Topics 123, 124, 128, 107, and 133 (bottom leftmost, 3rd quadrant) which are about written and oral communication skills. Topics 46, 52, 50, 83, and 31 (upper between 1st and 2nd quadrant) are about how someone should work (fast paced and dynamic), and the qualities needed to perform the work (adaptable, able to multitask, and can work independently or in a team).

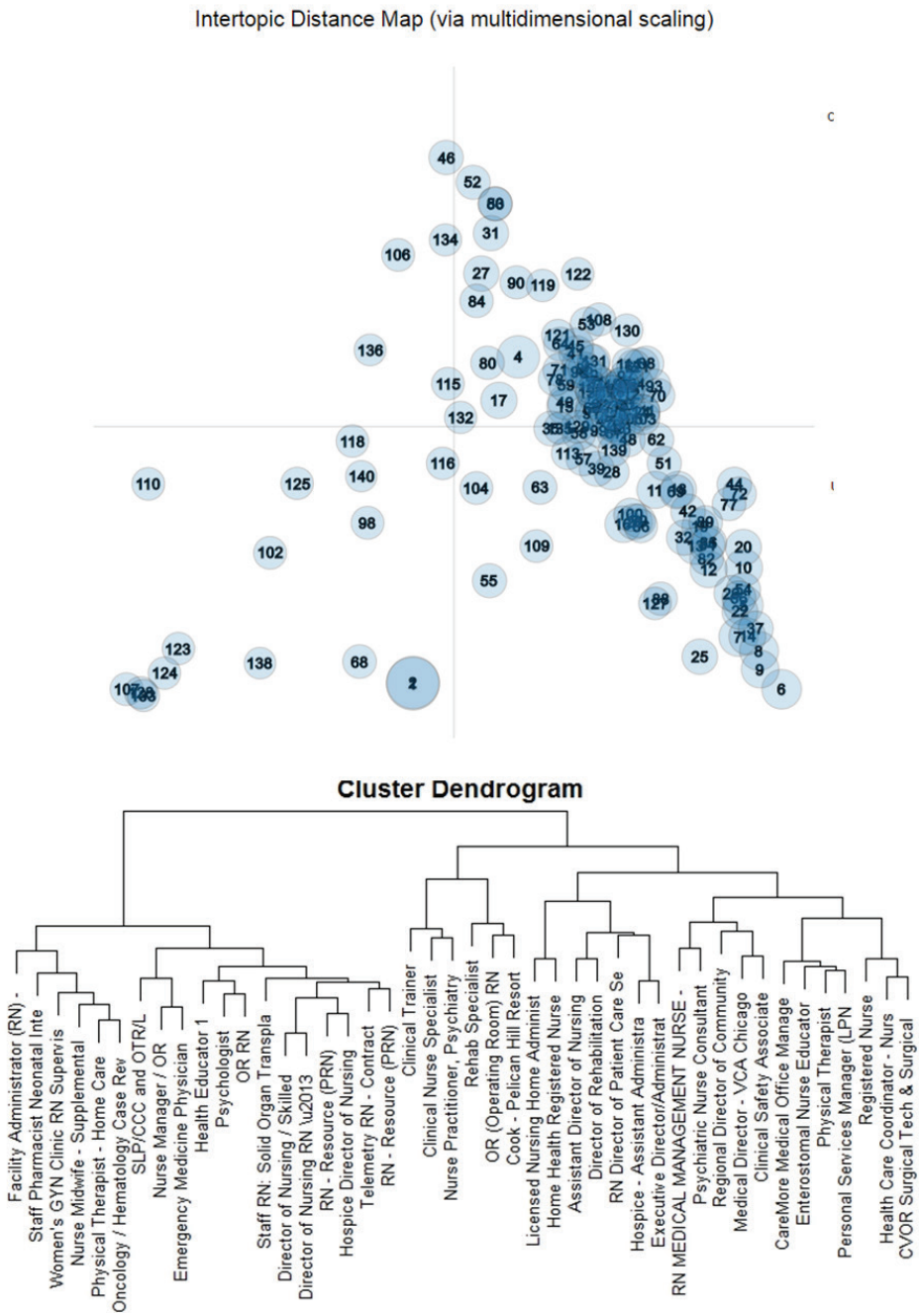


Figure 1.4. (a) Intertopic Distance Map and (b) Cluster Dendrogram of medically related jobs.

The output from LDA allows us to determine the most likely topic for each document. Here we want to find the most likely worker attribute for each job. Consider Topics 16 and Topic 18. Most jobs under Topic 16 are quantitatively oriented jobs such as Data Scientist, Statistician, and Financial Analyst. On the other hand, jobs under Topic 18 appear to pertain mostly to sales, marketing, and customer management. Note that in LDA, each document can have more than topics (each document is a mixture of topics), we can utilize all topic probabilities for each document and construct a hierarchical clustering of jobs. In Figure 1.4(b) we show part of the cluster dendrogram highlighting medically related jobs.

Terms associated with topics give us an idea about the possible interpretation of topics, however, we need to examine the relationship graph to help us surmise the context in which these words are used. Also, topic modeling showed that it is not only possible to accurately classify job information from vacancies but that we can also derive behavioral characteristics that are valued or required by employers from potential or existing job holders. We further made use of the extracted job information by summarizing the worker attributes on 140 dimensions, defining "job similarity" based on topic mixtures, and then clustering the jobs. Further analysis can be performed such as analyzing trends of worker attributes required by organizations across time, occupations, companies, and geographical regions given that these types of information are generally provided in the vacancies. Also, one can build a network of work activities to examine relationships among tasks.

Data collection, through TM, is faster, cheaper, and more reliable than traditional job analytic methods (McEntire et al., 2006). For our work on nursing tasks extraction, data triangulation showed that a substantial amount of TM extracted may be characterized as context-specific (e.g., caring for patients with spine surgery, caring for mentally ill patients) and that not all nurses perform these tasks. These tasks reflect idiosyncrasies in jobs that may be overlooked with data collection from SMEs because it would be impossible to interview, observe, and/or survey all nurses. Due to context-specificity, traditional ways of data collection have compromised the reliability of job-analytic data, causing bias (Dierdorff & Morgeson, 2009; Morgeson et al., 2004; Morgeson & Campion, 1997, 2000; Sanchez & Levine, 2000a). Our application of TM, however, showed that this information can be extracted automatically from vacancies to complement, enrich, and strengthen traditional methods of job analysis.

Of course, there are also validity concerns of using online vacancies as a data source. First, there are noticeable differences in the quality of the information across sources. For example, vacancies posted by recruitment agencies

are often lower in quality (e.g., level of detail, clarity of information) compared to vacancies posted by organizations. Data triangulation for the nurses also showed that specificity varied a lot between TM and task inventory data. There are for example five tasks about medication (i.e., prepare medication, arrange medication new patients, check medication, and hand out medication), all with extensive descriptions in the task inventory, whereas the TM counterpart is only 'administration of medication'. Thus, the level of detail is much lower there. Second, online data, as all secondary data, are often produced with vastly different purposes other than the research purpose it may subsequently be repurposed for, in this case job analysis. For example, online vacancies are aimed at recruiting employees, which means that the included information might be biased through advertising only certain, mainly positive, aspects of the job and/or not mentioning very mundane tasks. Tasks unique to the traditional task inventory included, for example more mundane and less positive, but very frequently occurring tasks in the nursing profession (e.g., washing patients, changing patients, cleaning beds, checking temperature). Third, not all jobs are advertised online (Sodhi & Son, 2007), potentially leaving out relevant information and jobs. Our recommendation to further validate the relationships is to compare the results we obtained with alternative sources of information such as interviewing SMEs or job incumbents, and computing measures traditionally used in inter-rater reliability as what we did with nursing tasks.

Discussion and Summary

This paper presented TM steps and associated methodologies to provide a sense of the applicability of TM methodologies within the field of organizational research. When confronted with a large volume of text data, TM can reduce personnel and cost constraints (i.e., hiring manual coders). Besides discussing steps and techniques, the practical recommendations sections offered tips on how to start TM and which tools to use, and we illustrated how TM can be applied in the field of job analysis.

When incorporating TM in organizational research, domain knowledge or theory can help supplement the more inductive approach often followed in TM and we tried to illustrate the role and importance of such knowledge and theory to a number of TM steps (see Figure 1.1). TM also allows flexibility and opportunity to recover potentially useful patterns which have previously been inaccessible from large amounts of text. Yet, for the expansion of TM to areas where research goals are not only to classify or to cluster but also to explain, using existing knowledge or theory and incorporating this into the analysis from the start is vital (see also

George, Haas, & Pentland, 2014).

In establishing or evaluating the reliability and validity of a given study using TM, a key question is whether we should adopt our evaluative criteria from the qualitative research tradition (cf. Yu et al., 2011), the quantitative research tradition, or perhaps even both. Of course, insofar as a TM study can withstand scrutiny from both methodological perspectives, this only serves to increase its credibility. Yet, the relevance of specific quality measures is likely to be contingent upon the epistemological orientation and specific objectives of the researcher. That is, for more exploratory or descriptive studies, such as those relying on topic modelling and clustering (see Table 1.1), it is not mandatory to impose strategies designed for establishing the validity of inferences. "[B]y definition 'inference' is an act of expanding the conclusion from a smaller subset to a broader set (e.g., from the sample statistics to the population parameter), but most qualitative studies do not aim to make 'valid inferences'." (C. H. Yu et al., 2011, p. 736). Krippendorf (2012) echoed this for CATA stating that "Deductive and inductive inferences are not central to content analysis" (p.36). Nevertheless, TM output can also provide a starting point for studies aiming to take an inferential route.

TM also has limitations and constraints. TM requires specific expertise and resources. Not all organizations/researchers have the computing resources to develop massive TM applications or the necessary expertise to execute these appropriately. The expertise and computing resources constraint could be addressed by outsourcing the task to companies and people who specialize in TM. Another limitation is the question of the representativeness of the information found in text data. The quality of the data will matter for the outcomes as with any type of data. The limitation of text data as an incomplete source of information could be mitigated by supplementing the analysis with additional types of data. For instance, in our job vacancy analysis we could triangulate our findings against the Occupational Information Network (Jeanneret & Strong, 2003), or other data sources that provide rich job information.

The different legal and ethical considerations that come with using particular forms of text data form a final limitation. Some text data are proprietary or contain privacy sensitive information that may be difficult to anonymize. The difficulty of obtaining permission to use text data can be addressed in part by implementing safeguards to protect the confidentiality of the data and to perform the analysis securely. Wider ethical concerns (Van Wel & Royakkers, 2004) on the use of 'Big' data, urgently need further and wider development and discussion.

We hope our discussion of TM helps foster dialogue and collaboration between organizational researchers and data scientists, particularly text miners.

Though most discussions here have centered on how TM can help organizational research, TM as a field also has something to gain from organizational research. The richness of problems that organizational research is trying to analyze can stimulate the creation of novel TM methodologies, thereby contributing to its advancement. In sum, the deluge of text data, the need to combine qualitative approaches with their quantitative counterparts, and the resulting progress for the two fields (organizational research and TM) brought by the interplay of theory and methods make the inclusion of TM methods ever more relevant to organizational research.

Appendix: Glossary of Terms as used by the Text Mining Community

Corpus: A collection of documents.

Document: A sequence of characters or a string. In this context, it is better understood as a file containing words, punctuations and special characters in a particular language. It is synonymous with the word text. Examples of documents are hiring offers, email messages, company mission statements, responses to open-ended survey questions, journal articles, and books.

Feature: A variable used to capture a characteristic of text data. The word feature is usually synonymous with (input) variable (i.e., terms), although at times input variables may be pre-processed to compute a feature (Guyon & Elisseeff, 2003), somewhat akin to the process by means of which survey items may be pre-processed to yield a score for a construct. The application of feature selection techniques yields a subset of features that is then used to construct a classification model.

Labeled Data: In classification, labeled data refer to documents whose category membership is known. For the purposes of constructing and evaluating the algorithm, these are respectively split into training and test data.

Term: A unit in a document. It can be a word, a phrase, or a sentence. Punctuation marks can also be considered as terms.

Test Data: Human labeled data used to evaluate the performance of a model.

Training Data: Human labeled data used to construct the classification model.

Vector: An arranged array of numbers that represent the scores on features for a particular document

Vocabulary or Lexicon: The set of all unique terms in a corpus.

Text Classification for Organizational Researchers

The previous Chapter laid the foundations and offered practical recommendation on how to apply text mining. This Chapter focuses on a specific yet pervasive method of text mining, that is, text classification. Many problems in organizational research can be formulated as text classification problems. Hence, we provide here a tutorial complete with codes on text classification.

Text data are pervasive in organizations. Digitization (Cardie & Wilkerson, 2008) and the ease of creating online information (e.g., e-mail messages) (Berry & Castellanos, 2008) contributes to the vast quantities of text generated each day. Embedded in these texts is information that may improve our understanding of organizational processes. Thus, organizational researchers increasingly seek ways to organize, classify, label, and extract opinions, experiences, and sentiments from text (Pang & Lee, 2008; Wiebe et al., 2005). Up until recently, the majority of text analyses in organizations relied on time consuming and labor-intensive manual procedures, which are impractical and less effective for voluminous collections of *documents* especially when resources are limited (we refer the reader to the Glossary provided in Kobayashi, Mol, Berkers, Kismihók, & Den Hartog (2017) for the definition of text mining related terms). Hence, automatic (or computer-assisted) strategies are increasingly employed to accelerate the analysis of text (Berry & Castellanos, 2008).

Similar to content analysis (Duriau et al., 2007; Hsieh & Shannon, 2005; Scharkow, 2013) and template analysis (Brooks et al., 2015), a common objective of text analysis is to assign text to predefined categories. Manually assigning large collections of text to categories is costly and may become inaccurate and unreliable due to cognitive overload. Furthermore, idiosyncrasies among human coders may creep into the labeling process resulting in coding errors. One workaround is to code only part of the corpus as opposed to coding all documents. However, this comes at the expense of possibly omitting relevant information, which may lead to bias and a degradation of the internal and external validity of the findings. Another option is to hire multiple human coders but this adds cost (e.g., cost of hiring and training coders) and effort pertaining to determining inter-rater reliability and consensus seeking (Sheng et al., 2008). A final (and more affordable) option is to solicit the help of the public to label text, for instance through the Amazon Mechanical Turk

platform (Buhrmester et al., 2011). However, this may only be effective in labeling objective information (e.g., names of people, events, etc.) since it is often difficult to establish consistency on subjective labels (e.g., sentiments) (Wiebe et al., 2004). Hence, automatic text analysis procedures that reliably, efficiently, and effectively assign text elements to classes are both necessary and advantageous especially in dealing with a massive corpus of text.

This paper focuses on *automatic text classification* for several reasons. First, although text classification (henceforth TC), has been applied in various fields, such as in political science (Atteveldt et al., 2008; B. Yu et al., 2008), occupational fraud (Holton, 2009), law (Gonçalves & Quaresma, 2005), finance (Chan & Chong, 2017; Chan & Franklin, 2011; Kloptchenko et al., 2004), and personality research (Shen et al., 2013), so far its uptake in organizational research is limited. Second, the use of TC is economical both in terms of time and cost (Duriau et al., 2007). Third, many of the techniques that have been developed in TC, such as sentiment analysis (Pang & Lee, 2008), genre classification (Finn & Kushmerick, 2006), and sentence classification (Khoo et al., 2006) seem particularly well suited to address contemporary organizational research questions. Fourth, the acceptance and broader use of TC within the organizational research community can stimulate the development of novel TC techniques.

Tutorials or review-tutorials on TC that have been published so far (Harish, Guru, & Manjunath, 2010; Yong H. Li & Jain, 1998; Sebastiani, 2002) were targeted mainly towards researchers in the field of machine learning and data mining. This has resulted in a skewed focus on technical and methodological details. In this paper our goal is to balance the discussion among techniques, theoretical concepts, and validity concerns to increase the accessibility of text classification to organizational researchers.

Below we first discuss the TC process, by pointing out key concerns and providing concrete recommendations at each step. Previous studies are cited to enrich the discussion and to illustrate different use cases. The second part is a hands-on tutorial using part of our own work as a running example. We applied TC to automatically extract nursing job tasks from nursing vacancies to augment nursing job analysis (Kobayashi, Mol, Kismihók, & Hesterberg, 2016). The findings from this study were used in the EU funded Pro-Nursing (http://pro-nursing.eu) project which aimed to understand, among others, how nursing tasks are embedded in the nursing process. We also address validity assessment because the ability to demonstrate the validity of TC outcomes will likely be critical to its uptake by organizational researchers. Thus, we discuss and illustrate how to establish validity for TC outcomes. Specifically, we address assessing the predictive

validity of the classifier and triangulating the output of the classification with other data sources (e.g., expert input and output from alternative analyses).

Text Classification

Text classification (TC) is defined as the automatic assignment of text to one or more predefined classes (Y. H. Li & Jain, 1998a; Sebastiani, 2002). Formally, the task of TC is stated as follows. Given a set of text and a set of categories, construct a model of the form: $Y = f(X, \theta) + \varepsilon$ from a set of documents with known categories. In the preceding formula, is a suitably chosen text representation (e.g., a vector), $\boldsymbol{\Theta}$ is the set of unknown parameters associated with the function f (*also known as the classifier or classification model*) that need to be estimated using the training data, and $\mathcal{E}$ is the error of the classification. The error is added to account for the fact that f is just an approximation to the true but unknown function h such that $Y = h(X)$. Hence, the smaller $\mathcal{E}$ is, the more *effective* the classifier f is. The term usually takes numerical values indicating the membership of text to a particular category. For example, when there are only 2 categories, such as in classifying the polarity of relations between political actors and issues as either positive or negative (Atteveldt et al., 2008), can take the values of and , respectively signifying positive and negative sentiment. We further discuss how to deal with each part of the formula, such as how to choose X and f, below. Once the classification model has been constructed it is then used to predict the category of new text (Aggarwal & Zhai, 2012b).

An ideal classifier would mimic how humans process and deduce meaning from text. However, there are still many challenges before this becomes reality. Natural languages contain high-level semantics and abstract concepts (Harish, Guru, & Manjunath, 2010; Popping, 2012) that are difficult to articulate in computer language. For instance, the meaning of a word may change depending on the context in which it is used (Landauer, Foltz, & Laham, 1998). Also, lexical, syntactic, and structural ambiguities in text are continuing challenges that would need to be addressed (Hindle & Rooth, 1993; Popping, 2012). Another issue is dealing with typographical errors or misspellings, abbreviations, and new lexicons. Strategies for dealing with ambiguities all need to be explicated during classifier development. Before a classifier is deployed it thus needs several rounds of training, testing, fine tuning (of parameters), and repeated evaluation until acceptable levels of performance and validity are reached. The resulting classifier is expected to approximate the performance of human experts in classification tasks (Cardie & Wilkerson, 2008), but for a large corpus its advantage is that it will be able to do so in a faster, cheaper, and more reliable manner.

TC: The Process

The TC process consists of six interrelated steps, namely (1) text preprocessing, (2) text representation or transformation, (3) dimensionality reduction, (4) selection and application of classification techniques, (5) classifier evaluation, and (6) classifier validation. As with any research activity, before starting the TC process, we begin by formulating the research question and identifying text of interest. Here, we assume that classes are predefined and that the researcher has access to, or can gather, documents with known classes i.e., the *training data*. For example, in a study about identifying disgruntled employee communications, researchers used posts from intra-company discussion groups. Subsequently, using criteria on employee disgruntlement, two people manually classified 80 messages into either disgruntled or non-disgruntled communication (Holton, 2009). Another study focused on the detection of personality of users from their email messages. Researchers first administered a 120-item questionnaire to 486 users to identify their personalities after which their email messages over a 12-month period were collected (Shen et al., 2013). Compared to the study on disgruntlement, it is more straightforward to label the associated text in this latter study because the labels are based on the questionnaire. Researchers are often faced with the decision of how many documents to label, an issue we will return to in the "Other TC issues" section below. Once the training dataset has been compiled, the next step is to preprocess the documents.

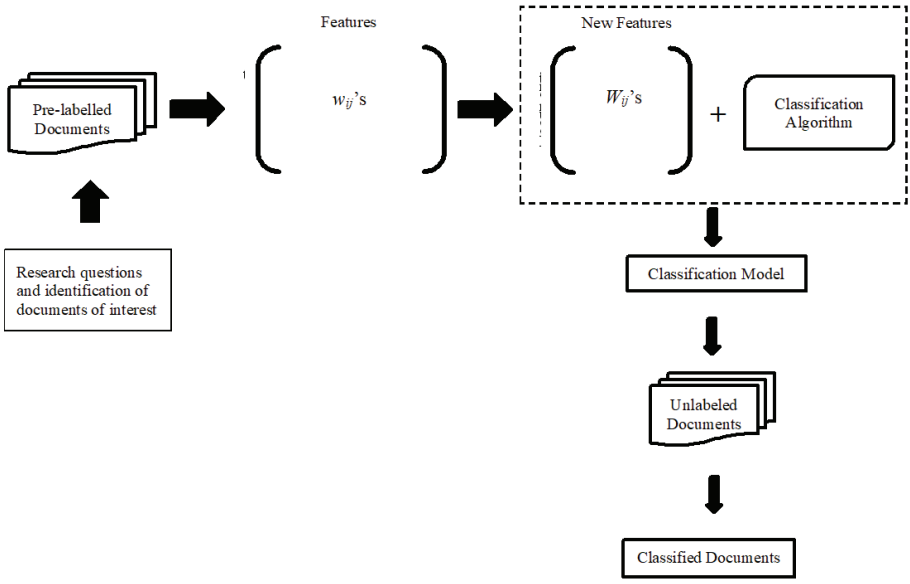


Figure 2.1. Diagrammatic depiction of the text classification process.

Text Preprocessing for Classification

The purpose of preprocessing is to remove irrelevant bits of text as these may obscure meaningful patterns and lead to poor classification performance and redundancy in the analysis (Uysal & Gunal, 2014). During preprocessing we first apply *tokenization* to separate individual terms. Terms may be words, punctuation marks, numbers, tags, and other symbols (e.g., an emoticon). In written English, terms are usually separated by spaces.

Punctuations and numbers, if deemed irrelevant to the classification task at hand are removed, although in some cases these may be informative and thus retained (exclamation marks or emoticons, for instance, may be indicative of sentiment). Dictionaries or lexicons are used to apply spelling correction, and to resolve typos, and abbreviations. Words that are known to have low information content such as conjunctions and prepositions are typically deleted. These words are called *stopwords* (Fox, 1992), examples of pre-identified stopwords in the English language are "and", "the", and "of" (see *http://www.ranks.nl/stopwords* for different lists stopwords in various languages). When the case of the letters is irrelevant it is advisable to transform all upper case letters into lower case.

During preprocessing *stemming,* which is defined as the process of obtaining the base or stem form of words (Frakes, 1992; Porter, 1980), is also commonly applied. A key assumption in stemming is that words that have similar root forms are identical in meaning. Stemming is performed by removing suffixes which may not correspond to an actual base form of the word (Willett, 2006). For example, the words *calculate, calculating, calculated* will be rewritten to calculate although the actual base form is calculate (Toman et al., 2006). If one wants to recover the actual base form, then one can use *lemmatization* instead of stemming. However, lemmatization is more challenging than stemming (Toman et al., 2006) and the added complexity of applying lemmatization may offset its benefits. Both lemmatization and stemming lead to a loss of inflection information in words (e.g., tense, gender, and voice). Inflection information may be important in some applications, such as in identifying the sentiment of product reviews, since as it turns out, most negative reviews are written in the past tense (Dave et al., 2003a). Stemming and lemmatization are part of a broad class of preprocessing techniques called *normalization* (Dave et al., 2003b; Toman et al., 2006). The aim of normalization is to merge terms that express the same idea or concept under a single code called a *template.* For example, another normalization strategy is to use the template POST_CODE to replace all occurrences of postcodes in a collection of documents. This can be useful when it is important to consider if a

document does or does not contain a postcode (i.e., contains an address), but the actual postcode is irrelevant.

A practical question is: what preprocessing techniques to apply for a given text? The answer is largely determined by the nature of text (e.g., language and genre), the problem that we want to address, and the application domain (Uysal & Gunal, 2014). Any given preprocessing procedure may be useful for a specific domain of application or language but not for others. Several empirical studies demonstrated the effect of preprocessing on classification performance. For example, stemming in the Turkish language does not seem to make a difference in classification performance when the size of the training data set is large (Torunoğlu et al., 2011). In some applications stemming even appears to degrade classification performance, particularly in the English and Czech languages (Toman et al., 2006). In the classification of English online news, the impact of both stemming and stopword removal is negligible (Song, Liu, & Yang, 2005). In general, the classification of English and Czech documents benefits from stopword removal but may suffer from word normalization (Toman et al., 2006). For the Arabic language, certain classifiers benefit from stemming (Kanaan et al., 2009). In spam email filtering, some words typically seen as stopwords (e.g., "however" or "therefore") were found to be particularly rare in spam email, hence these should not be removed for this reason (Méndez et al., 2006).

Recommendation. In using English documents, our general recommendation is to apply word tokenization, convert upper case letters to lower case, and apply stopword removal (except for short text such as email messages and product titles) (Méndez et al., 2006; H.-F. Yu et al., 2012). Since the effects of normalization have been mixed, our suggestion is to apply it only when there is no substantial degradation on classification performance, since it can increase classification efficiency by reducing the number of terms. When in doubt whether to remove numbers or punctuation (or other symbols), our advice is to retain them and apply the dimensionality reduction techniques discussed in the below section on text transformation.

Text Transformation (x)

Text transformation is about representing documents so that they form a suitable input to a classification algorithm. In essence, this comprises imposing structure on a previously unstructured text. Most classification algorithms accept vectors or matrices as input. Thus, the most straightforward way is to represent a document as a vector and the corpus as a matrix.

The most common way to transform text is to use the so-called vector *space model* (VSM) where documents are modeled as elements in a vector space (Raghavan & Wong, 1986; Salton, Wong, & Yang, 1975). The features in this representation are the individual terms found in the corpus. This somehow makes sense under the assumption that words are the smallest independently meaningful units of a language. The size of the vector is therefore equal to the size of the vocabulary (i.e., the set of unique terms in a corpus). Hence, we can represent document j as $x_j = (x_j^1 \quad x_j^2 \quad \cdots \quad x_j^M)$ where M is the size of the vocabulary, and the element x_j^i is the weight of term i in document j. Weights can be the count of the terms in a document ($x_j^i = TF(j,i)$) or, when using binary weighting, a 1 (presence of a term) or 0 (absence of a term). Applying the transformation to the entire corpus will lead to a *Document-by-Term matrix* (DTM), where the rows are the documents, the columns are the terms, and the entries are the weights of the terms in each document.

Other weighting options can be derived from basic count weighting. One can take the logarithm of the counts in order to dampen the effect of highly frequent terms. Here we need to add 1 to the counts so that we avoid taking the logarithm of zero counts. It is also possible to normalize with respect to document length by dividing each count by the maximum term count in a given document. This is to ensure that frequent terms in long documents are not overrepresented. Apart from the weights of the terms in each document, terms can also be weighted with respect to the corpus. Common corpus-based weights include the inverse document frequency (IDF) which assesses the specificity of terms in a corpus (Algarni & Tairan, 2014). Terms that occur in too few (large IDF) or in too many (IDF close to zero) documents have low discriminatory power and are therefore not useful for classification purposes. The formula for IDF is: $IDF(i) = \log\left(\frac{N}{df(i)}\right)$ where $df(i)$ stands for the document frequency of term *i*, i.e., the number of documents containing term *i*. Document- and corpus –based weights may also be combined so that that the weights simultaneously reflect the importance of a term in a document and its specificity to the corpus. The most popular combined weight measure is the product of term frequency (TF) and the inverse document frequency (IDF) ($x_j^i = TF(j,i) \times IDF(i)$) (Aizawa, 2003).

Although the vector space model (VSM) ignores word order information, it is popular due to its simplicity and effectiveness. Ignoring word order means losing some information regarding the semantic relationships between words. Also, words alone may not always express true atomic units of meaning. Some

researchers improve the VSM by adding adjacent word pairs or trios (*bigrams* and *trigrams*) as features. For example, "new" followed by "york" becomes "new york" in a bigram. Although this incorporates some level of word order information it also leads to feature explosion thereby increasing noise and redundancy. Also, many bigrams and trigrams do not occur often, thus their global contributions to the classification are negligible and will only contribute to sparsity and computational load. A workaround is to use only the most informative phrases (e.g., frequent phrases) (Scott & Matwin, 1999b). Strategies for selecting key phrases include the noun phrase (Lewis, 1992b) and key phrase (Turney, 1999) extraction algorithms. However, this does add additional complexity in the analysis which may again not result in a significant improvement in the classification. Studies have consistently shown that using bigrams only marginally improved classification performance and in some cases degraded it, whereas the use of trigrams typically yielded improvement (Dave et al., 2003a; Ragas & Koster, 1998). Using syntactic phrases typically does not improve performance much compared to single term features (Moschitti & Basili, 2004; Scott & Matwin, 1999b). Thus, the recommendation is to rely on single term features rather than phrases unless there is a strong rationale to use phrases.

Text transformation plays a critical role in determining classification performance. Inevitably some aspects of the text are lost in the transformation phase. Thus, when the resulting classification performance is poor, we recommend that the researcher re-examines this step. For example, while term-based features are popular, if performance is poor one could also consider developing features derived from linguistic information (e.g., parts-of-speech) contained in text (Gonçalves & Quaresma, 2005; Kobayashi, Mol, Berkers, et al., 2017; Moschitti & Basili, 2004) or using consecutive characters instead of whole words (e.g., n-grams) (Cavnar et al., 1994).

Reducing dimensionality. Even after preprocessing, transformation through VSM is still likely to result in a large feature set. Too large a number of features is undesirable because it may increase computational time and may degrade classification performance, especially when there are many redundant and noisy features (Forman, 2003; Guyon & Elisseeff, 2003; Joachims, 1998). The size of the vector and hence the size of feature set is referred to as the dimensionality of the VSM representation. When possible, one should reduce *dimensionality* either by selectively eliminating features or by creating latent features from existing ones without sacrificing classification performance (Burges, 2010; Fodor, 2002; van der Maaten et al., 2009). A reduced feature set has advantages such as higher efficiency and, in some cases, improved classification performance.

One way to eliminate features is to first assign scores to each feature and then remove features by setting a cut-off value, this is called *thresholding* (Lan, Tan, Su, & Lu, 2009; Salton & Buckley, 1988). Weights from the transformation steps are sometimes used to score features. An example is to remove rare terms, that is, terms with high IDF or low DF since they are non-informative for category prediction or not influential in global performance. In some cases, rare terms are noise terms (e.g., misspellings).

Another group of strategies to score features is to make use of class membership information in the training data. These methods are called *supervised scoring methods*. Examples of these methods are mutual information (MI), chi-squared (CHI), Gini index (GI), and information gain (IG) (Yang & Pedersen, 1997b). Supervised scoring methods are expected to be superior to unsupervised ones (e.g., DF) although in some cases DF thresholding has yielded performance comparable to supervised scoring methods such as CHI and GI (Yang, 1999) and even exceeded the performance of MI.

An alternative to scoring methods is to create latent orthogonal features by combining existing features. Methods that construct new features from existing ones are known as *feature transformation methods*. Techniques include Principal Component Analysis (PCA) (Sirbu et al., 2016; Zu et al., 2003), Latent Semantic Analysis (LSA) (Landauer et al., 1998), and Nonnegative Matrix Factorization (Zurada et al., 2013). These methods construct high level features as a (non) linear combination of the original features with the property that the new features are uncorrelated. They operate on the DTM by applying a matrix factorization method. The text is scored (or projected) on the new features, or factors, and these new features are used in the subsequent analysis. LSA improves upon the vector space model through its ability to detect synonymy (Landauer, Foltz, & Laham, 1998). Words that appear together and load highly on a single factor may be considered to be synonyms.

Recommendation. Our recommendation is start with the traditional vector space model, that is, transform the documents into vectors using single terms as features. For the unsupervised scoring, compute the DF of each term and filter out terms with very low and very high DF, customarily those terms belonging to the lower 5th and upper 99th percentiles. For the supervised scoring try CHI and IG and for the feature transformation try LSA and Nonnegative Matrix Factorization. Compare the effect on classification performance of the different feature sets generated by the methods and choose the feature set that yields the highest performance (e.g., accuracy). We also suggest to try combining scoring and transformation methods. For example, one can first run CHI and perform LSA

on the terms selected by CHI. Note that the quality of the feature set (and that of the representation) is assessed based on its resulting classification performance (Forman, 2003).

For LSA and Nonnegative Matrix Factorization, we need to decide how many dimensions to retain. For LSA, Fernandes, Artífice, & Fonseca (2017) offered this formula as a rough guide $K = N^{\left\{\frac{1}{1+\frac{\log(N)}{10}}\right\}}$, where N is the size of the corpus, K is the number of dimensions to retain and the logarithm is base 10. For example, if there are 500 documents, then retain approximately 133 latent dimensions. In the case of Nonnegative Matrix Factorization, an upper bound for choosing K is that it must satisfy this inequality $(N+M)K < NM$, where M is the number of original features (Tsuge et al., 2001). Hence if there are 500 documents and 1000 terms, K should not be greater than 333. Of course, one has to experiment with different sizes of dimensionality and select the size that yields the maximum performance. For example, the formula gave 133 dimensions for 500 documents, but one may also try experimenting values within ± 30 around 133.

Application of TC Algorithms (f)

The transformed text, usually the original DTM or the dimensionality reduced DTM, serves as input to one or more classification techniques. Most techniques are from the fields of machine learning and statistics. There are three general types of techniques: (1) geometric, (2) probabilistic, and (3) logical (Flach, 2012).

Geometric algorithms assume that the documents can be represented as points in a hyperspace, the dimensions of which are the features. This means that distances between documents and lengths of the documents can be defined as well. In this representation, nearness implies similarity. An example of a geometric classifier is *K-nearest neighbors* in which classification is done by first finding the closest K documents (using a distance measure) from the training data (Jiang et al., 2012) then the majority class of the K closest documents is the class to which the new document is assigned. The parameter K is chosen to be an odd number to prevent ties from occurring. Another geometric classifier is *support vector machines* (Joachims, 1998) in which a hyperplane is constructed that provides the best separation among the text in each class. The hyperplane is constructed in such a way that it provides the widest separation between the two nearest observations of each class.

Probabilistic algorithms compute a joint probability distribution between the observations (e.g., documents) and their classes. Each document is assumed to

be an independent random draw from this joint probability distribution. The key point in this case is to estimate the posterior probability $P(Y_m \mid X)$. Classification is achieved by identifying the class that yields the maximum posterior probability for a given document. The posterior probability is estimated in two ways. Either one can marginalize the joint distribution $P(\mathrm{X}, \mathrm{Y}_m)$ or one may compute $P(\mathrm{X} \mid \mathrm{Y}_m)$ and $P(Y_m)$ separately and apply Bayes theorem. Both *Naive Bayes* (Eyheramendy et al., 2003) and logistic regression (J. Zhang et al., 2003) are examples of probabilistic algorithms.

The third type of algorithm is the logical classifier, which accomplishes classification by means of logical rules (Dumais et al., 1998; Rokach & Maimon, 2005). An example of such a rule in online news categorization is: "If an article contains any of the terms "vs", "earn", "loss" and not the words "money", "market open", or "tonn" then classify the article under category "earn" (Rullo et al., 2007). The rules in logical models are readable and thus facilitate revision, and, if necessary, correction of how the classification works. An example of a logical classifier is a *decision tree* (Rokach & Maimon, 2005).

Naive Bayes and Support Vector Machines are popular choices (Ikonomakis et al., 2005; Joachims, 1998; Y. H. Li & Jain, 1998a; Sebastiani, 2002). Both can efficiently deal with high dimensionality and data sparsity, though in Naive Bayes appropriate smoothing will need to be applied to adjust for terms which are rare in the training data. The method of K-nearest neighbor works well when the amount of training data is large. Both logistic regression and discriminant analysis yield high performance if the features are transformed using LSA. The performance of decision trees has been unsatisfactory. A number of researchers therefore recommend the strategy of training and combining several classifiers to increase classification performance, which is known as ensemble learning (Breiman, 1996; Dietterich, 1997; Dong & Han, 2004; Polikar, 2012). This kind of classification can be achieved in three ways. The first is using a single method and training it on different subsets of the data. Examples include Bagging and Boosting which both rely on resampling. Random forest is a combination of bagging and random selection of features that uses decision trees as base learners. Gradient Boosted Trees, a technique which combines several decision trees, has been shown to significantly increase performance as compared with that of individual decision trees (A. J. Ferreira & Figueiredo, 2012). The second is using a single method but varying the training parameters such as, for example, using different initial weights in neural networks (Kolen & Pollack, 1990). The third is using different classification techniques (Naive Bayes, Decision Trees, or SVM) (Y. H. Li & Jain, 1998a) and combining their predictions using, for instance, the majority vote.

Recommendation. Rather than using a single technique, we suggest applying different methods, by pairing different algorithms and feature sets (including those obtained from feature selection and transformation) and choosing the pair with the lowest error rate. For example, using the DTM matrix, apply SVM, Naive Bayes, Random Forest bagging, and Gradient Boosted Trees. When feature transformation has been applied (e.g., LSA and Nonnegative Matrix Factorization), use logistic regression or discriminant analysis. When the training data is large (e.g., hundreds of thousands of cases), use K-nearest neighbors. Rule-based algorithms are seldom used in text classification, however, if readability and efficiency are desired in a classifier, then these can be trialed as well.

Evaluation Measures

Crucial to any classification task is the assessment of the performance of classifiers using evaluation measures (Powers, 2011; Yang, 1999). These measures indicate whether a classifier models the relationship between features and class membership well and may thus be used to indicate the extent to which the classifier is able to emulate a human coder. The most straightforward evaluation measure is the accuracy measure, which is calculated as the proportion of correct classifications. Accuracy ranges from 0 to 1 (or 0 to 100 when expressed as a percentage). The higher the accuracy the better the classifier (1 corresponds to perfect classification). However, in case of imbalanced classification (i.e., when there is one class with only a few documents) and/or unequal costs of misclassification, accuracy may not be appropriate. An example is detecting career shocks in job forums. Since it is likely that only a small fraction of these postings pertain to career shocks (suppose 0.05), a classifier can still have a high accuracy (equal to .95) even if that classifier classifies all discussion as containing no career shocks content.

Alternative measures to accuracy are precision, recall, F-measure (Powers, 2011), specificity, break-even point, and balanced accuracy (Ogura et al., 2011). In binary classification, classes are commonly referred to as positive and negative. Classifiers aim to correctly identify observations in the positive class. A summary table which can be used as a reference for computing these measures is presented in Figure 2.2. The entries of the table are as follows: TP stands for true positives, TN for true negatives, FP for false positives (i.e., negative cases incorrectly classified into the positive class), and FN for false negatives (i.e., positive cases incorrectly classified into the negative class). Hence the five evaluation measures are computed as follows:

$$precision = \frac{TP}{TP + FP} \quad , \quad recall = \frac{TP}{TP + FN} \quad , \quad specificity = \frac{TN}{TN + FP} \quad ,$$

$$F - measure = \frac{2 \times recall \times precision}{recall + precision} \text{, and } Bal.Accu = \left(\frac{TP}{TP + FP} + \frac{TN}{TN + FN} \right) \Big/ 2 .$$

The break-even point is the value at which $precision = recall$. F-measure and balanced accuracy are generally preferred in case of imbalanced classification, because they aggregate the more basic evaluation measures.

		Predicted Classes	
		Positive	Negative
True Classes	Positive	TP	FN
	Negative	FP	TN

Figure 2.2. Confusion matrix as a reference to compute the evaluation measures. Note. TP = True positive; FN = False negative; FP = False positive; TN = True negative.

Evaluation measures are useful to compare the performance of several classifiers (Alpaydin, 2014). Thus, one can probe different combinations of feature sets and classification techniques to determine the best combination (i.e., the one which gives the optimal value for the evaluation measure). Apart from classification performance, one can also take the parsimony of the trained classifier into account by examining the relative size of the different feature sets, since they determine the complexity of the trained classifier. In line with Occam's razor, when two classifiers have the same classification performance, the one with the lower number of features is to be preferred (Shreve et al., 2011).

Evaluation measures are computed from the labeled data. It is not advisable to use all labeled data to train the classifier since this might result in *overfitting* which is the case when the classifier is good at classifying the observations in the training data but performs poorly on new data. Hence, part of the labeled data should be set aside for evaluation so that we can assess the degree to which the classifier is able to predict accurately in data that were not used for training.

Cross-validation can be applied by computing not only one value for the evaluation measure but several values corresponding to different splits of the data. A systematic strategy to evaluate a classifier is to use k-fold cross-validation (Kohavi, 1995). This method splits the labeled dataset into k parts. A classifier is

trained using parts and evaluated on the remaining part. This is repeated so that each of the parts had been used as a test data. Thus for equals 10, there are 10 partitions of the labeled data and corresponding 10 values for a given measure, the final estimate is just the average of the 10 values. Another strategy is called bootstrapping which is accomplished by computing an average of the evaluation measures for N bootstrap samples of the data (sampling with replacement).

Recommendation. Since accuracy may give misleading results when classes are imbalanced we recommend using measures sensitive to this, such as F-measure or balanced accuracy (Powers, 2011). For the systematic evaluation of the classifier we advise using K-fold cross validation and setting K to 5 or 10 when data is large, as this ensures sufficient data for the training. For smaller data sets, e.g., fewer than 100 documents, we suggest bootstrapping or choosing a higher K for cross-validation.

Model Validity

Figure 2.1 illustrates that a classification model consists of features and the generic classification algorithm (Domingos, 2012). Thus, the validity of the classification model depends both on the choice of features and the algorithm.

Many TC applications use the set of unique words as the feature set (i.e., vector space model). For organizational researchers this way of specifying the initial set of features may seem counterintuitive since features are constructed in an ad hoc and inductive manner, that is, without reference to theory. Indeed, specifying the initial set of features, scoring features, transforming features, evaluating features, and modifying the set of features in light of the evaluation constitutes a data-driven approach to feature construction and selection (Guyon et al., 2008). The validity of the features is ultimately judged in terms of the classification performance of the resulting classification model. But this does not mean that researchers should abandon theory based approaches. If there is prior knowledge or theory that supports the choice of features then this can be incorporated (Liu & Motoda, 1998). Theory can also be used as a basis for assigning scores to features such as using theory to rank features according to importance. Our recommendation, however, would be to have theory complement, as opposed to restrict, feature construction, because powerful features (that may even be relevant to subsequent theory building and refinement) may emerge inductively.

The second component, the classification algorithm, models the relationship between features and class membership. Similar to the features, the validity of the algorithm is ultimately determined from the classification performance and is also for the most part data driven. The validity of both the

features and the classification algorithm establishes the validity of the classification model.

A useful strategy to further assess the validity of the classification model is to compare the classifications made by the model with the classification of an independent (group of) human expert(s). Usually agreement between the model and the human expert(s) is quantified using measures of concordance or measures of how close the classification of the two correspond to one another (such as Cohen's kappa for inter-rater agreement where one 'rater' is the classifier). Using expert knowledge, labels can also be checked against standards. For example, in job task extraction from a specific set of job vacancies one can check with experts or job incumbents to verify whether the extracted tasks correspond to those tasks actually carried out on the job and whether specific types of tasks are under or over represented.

Once model validity is established one may start applying the classification model to unlabeled data. However, the model will still need to be reevaluated from time to time. When the performance drops below an acceptability threshold, there are four possible solutions: (1) add more features or change existing features, (2) try other classification algorithms, (3) do both, and/or (4) collect more data or label additional observations.

Other Issues in TC

In this section we discuss how to deal with multi-class classification, where there is an increased likelihood of classes being imbalanced, and provide some suggestions on determining training size and what to do when obtaining labeled data is both expensive and difficult.

Multi-class classification. Multi-class classification pertains to dealing with more than 2 categories. The preprocessing and representation parts are the same as in the binary case. The only changes are in the choices of supervised feature selection techniques, classification techniques and evaluation measures. Most supervised feature selection techniques can be easily generalized to more than 2 categories. For example, when calculating CHI, we just need to add an extra column to the two-way contingency table. Most techniques for classification we discussed previously have been extended to multi-class classification. For example, techniques suited for binary classification problems (e.g., SVM) are extended to the multi-class case by breaking the multi-class problem into several binary classification problems in either one-against all or one-against-one approaches. In the former approach we build binary classifiers by taking each category as the positive class and merging the others into the negative class. Hence, if there are K

categories, then we build K binary classifiers. For the latter approach, we construct a binary classifier for each pair of categories resulting to $\frac{K(K-1)}{2}$ classifiers. Since several classifiers are built, and thus there are several outputs, final category membership is obtained by choosing the category with the largest value for the decision function for the one-against-all case or by a voting approach for the one-against-one case (Hsu & Lin, 2002).

The four evaluation measures can also be extended to classifications with more than 2 classes by computing these measures per category, the same as in one-against-all, and averaging the results. An example is the extension of F-measure called the *macro F-measure* which is obtained by computing the F-measure of each category, and then averaging them.

Imbalanced classification. By and large, in binary classification, when the number of observations in one class represent less than 20% of the total number of observations then the data can be seen as imbalanced. The main danger of imbalanced classification is that we may train a classifier with a high accuracy even if it fails to correctly classify the observations in the minority class. In some cases, we are more interested in detecting the observations in the minority class. At the same time however, we also want to avoid many false detections.

Obvious fixes are to label more observations until the classes are balanced as was done by Holton (2009) or by disregarding some observations in the majority class. In cases where classification problems are inherently imbalanced and labeling additional data is costly and difficult, another approach is to oversample the minority class or to undersample the majority class during classifier training and evaluation. A strategy called the Synthetic Minority Oversampling Technique (SMOTE) is based on oversampling but instead of selecting existing observations in the minority class it creates synthetic samples to increase the number of observations in the minority class (Chawla et al., 2002). Preprocessing and representation remain the same as in balanced classes. The parts that make use of class membership need to be adjusted for imbalanced data.

There are options for supervised dimensionality reduction for imbalanced classification such as those provided by Ogura et al. (2011). For the choice of classification techniques, those discussed previously can be used with minor variations such as adjusting the costs of misclassification, which is known as cost-sensitive classification (Elkan, 2001). Traditional techniques apply equal costs of misclassification to all categories, whereas, for cost-sensitive classification we can assign large cost for incorrect classification of observations in the minority class. For the choice of evaluation measures, we suggest using the weighted F-measure

or balanced accuracy. One last suggestion is to treat imbalanced classification as an anomaly or outlier detection problem where the observations in the minority class are the outliers (Chandola et al., 2009).

Size of the training data. A practical question that often arises is how many documents one should label to ensure a valid classifier? The size of the training dataset depends on many considerations such as the cost and limitations associated with acquiring pre-labeled documents (e.g., ethical and legal impediments) and the kind of learning framework we are using. In the Probably Approximately Correct (PAC) learning framework, which is perhaps the most popular framework for learning concepts (such as the concept of spam emails or party affiliation) training size is determined by the type of classification technique, the representation size, the maximum error rate one is willing to tolerate, and the probability of not exceeding the maximum error rate. Under the PAC learning framework, formulae have been developed to determine the lower bound for the training size, an example being the one by Goldman (2010) $\Omega\left(\frac{1}{\varepsilon}\log_2\frac{1}{\delta}+\frac{\mathrm{VCD}(C)}{\varepsilon}\right)$ where is the maximum error rate, indicates the probability that the error will not exceed ε, and *VCD (C)* of the classifier C. VCD stands for Vapnik-Chervonenski dimension of the classifier C which can be interpreted as the expressive power of the classifier which depends on the representation size and the form of the classifier (e.g., axis parallel rectangle, closed sets, or half-spaces). As an illustration suppose we want to learn the concept of positive sentiment from English text, we then represented each document as a vector of 50,000 dimensions (number of commonly used English words) and our classification technique constructs a hyperplane that separates positive and negative observations (e.g., SVM using ordinary dot product as kernel). If we want to ensure with probability 0.99 that the error rate will not exceed 0.01, then the minimum training size is: $\frac{1}{0.01}\log_2\frac{1}{0.01}+\frac{50001}{0.01}=5000764$. This means we would need at least 5 million documents. Here we calculated the VCD of C using the formula $d+1$, where d is the dimensionality of the representation, since we consider classifiers that construct hyperplane boundaries (half-space) in 50000-dimensions. Of course, in practice dimensionality reduction can be applied still get adequate representation. If one would manage to reduce dimensionality to 200 then the lower bound for the training size is dramatically reduced to 20765. We can tweak this lower bound by adjusting the other parameters.

Although formulae provide theoretical guarantees, determining training size is largely empirically-driven and involves a good deal of training, evaluation, and validation. To give readers an idea of training sizes as typically found in practice, Table 2.1 provides information about the training data sizes for some

existing text classification studies.

Table 2.1. Training sizes, number of categories, evaluation measures, and evaluation procedures used in various text classification studies.

Authors	Subject matter	Training Size	Number of Categories	Evaluation Measure	Evaluation procedure
(Phan et al., 2008))	Domain disambiguation for Web search results	12340	8	Accuracy	5-fold CV
(Moschitti & Basili, 2004; Phan et al., 2008)	Disease classification for medical abstracts	28145	5	Accuracy	5-fold CV
(Vo & Ock, 2015)	Titles of scientific documents	8100	6	Accuracy & F-measure	5-fold CV
(Khoo et al., 2006)	Response emails of operators to customers	1486	14	Accuracy & F-measure	10-fold CV
(J. Chen et al., 2009; Y. H. Li & Jain, 1998b; Moschitti & Basili, 2004; Ogura et al., 2011; Scott & Matwin, 1999b; Song et al., 2005; Toman et al., 2006; Torunoğlu et al., 2011; Uysal & Gunal, 2014; Yang & Pedersen, 1997b; W. Zhang et al., 2008)	News items	764-19997	4-93	Accuracy, micro averaged break-even points, F-measure, recall, precision, break-even point	1 training and 2 test , single train-test, 20 splits with inter-corpus evaluation, 4-fold CV, 10-fold CV, 20 splits
(Atteveldt et al., 2008)	Sentiment analysis of actor-issue relationship	5348	2	F-measure	Did not mention
(Dave et al., 2003a)	Product reviews sentiments	31574	7	Accuracy	2 test
(Scott & Matwin, 1999b)	Song lyrics	6499	33	Micro-averaged break even points	Single-train test
(J. Chen et al., 2009)	Chinese news texts	2816	10	Accuracy & F-measure	Single-train test
(Kanaan et al., 2009)	Arabic news documents	1445	9	F-measure	4-fold CV
(Ragas & Koster, 1998)	Dutch documents	1436	4	Accuracy	Single train-test
(Méndez et al., 2006; Panigrahi, 2012; Uysal & Gunal, 2014; Youn & McLeod, 2007)	Emails	400-9332	2	F-measure	Single train-test
(Torunoğlu et al., 2011; Uysal & Gunal, 2014)	Turkish news items	1150-99021	5-10	F-measure	
(Moschitti & Basili, 2004)	Italian news items	16000	8	F-measure & break even point	20 splits with inter-corpus validation
(Toman et al., 2006)	Czech news items	8000	5	F-measure	4-fold CV
(Thelwall et al., 2010)	Cyberspace comment sentiment analysis	1041	5	Accuracy	10-fold CV
(Holton, 2009)	Disgruntled employee communications	80	2	Accuracy	Single train –test varying proportion
(Shen et al., 2013)	Personality from emails (this is a multi-label classification problem)	114907	3 categories per personality	Accuracy	10 fold CV and single train-test

Suggestions when labeled data are scarce. In many classification problems, labeled data are costly or difficult to obtain. Fortunately, even in this case, principled approaches can be applied. In practice, unlabeled data is plentiful and we can apply techniques to make use of the structure and patterns in the unlabeled data. This approach of using unlabeled data in classification is called *semi-supervised classification* (Zhu, 2005). Various assumptions are made in order to make semi-supervised classification feasible. Examples are the smoothness assumption which says that observations near each other are likely to share the same label, and the cluster assumption which states that if the observations form clusters then observations in the same cluster are likely to share the same label (Zhu, 2005).

Another approach is to use classification output to help us determine which observations to label. In this way, we take a targeted approach to labeling by labeling those observations which are most likely to generate better classifiers. This is called *active learning* in the machine learning literature (Settles, 2010). Active learning is made possible because some classifiers give membership probabilities or confidence rather than a single decision as to whether to assign to one class or not. For example, if a classifier assigns equal membership probabilities to all categories for a new observation then we call an expert to label the new observation. For a review of active learning techniques we refer the reader to (Fu et al., 2013).

Tutorial

We developed the following tutorial to provide a concrete treatment of TC. Here we demonstrate TC using actual data and codes. Our intended audience are researchers who have little or no experience with TC. This tutorial is a scaled down version of our work on using TC to automatically extract job tasks from job vacancies. Our objective is to build a classifier that automatically classifies sentences into task or non-task categories. The sentences were obtained from German language nursing job vacancies.

We set out to automate the process of classification because one can then deal with huge numbers (i.e., millions) of vacancies. The output of the text classifier can be used as input to other research or tasks such as job analysis or the development of tools to facilitate personnel decision making. We used the R software since it has many ready-to-use facilities that automate most TC procedures. We provide the R annotated scripts and data to run each procedure. Both codes and data can be downloaded as a Zip File from Github; the URL is https://github.com/vkobayashi/textclassificationtutorial. The naming of R scripts are in the following format: CodeListing (CL) <number>.R and in this tutorial we referenced them as CL <number>. Thus, CL 1 refers to the script CodeListing_1.R. Note that the CL files

contain detailed descriptions of each command, and that each command should be run sequentially.

All the scripts were tested and are expected to work on any computer (PC or Mac) with R, RStudio, and the required libraries installed. However, basic knowledge including how to start R, open R projects, run R commands, and install packages in RStudio are needed to run and understand the codes. For those new to R we recommend following an introductory R tutorial (see, for example, DataCamp (www.datacamp.com/courses/free-introduction-to-r), or tutorialspoint (www.tutorialspoint.com/index.htm) for free R tutorials).

This tutorial covers each of the previously enumerated TC steps in sequence. For each step we first explain the input, elaborate the process, and provide the output, which is often the input for the subsequent step. Table 2.2 provides a summary of the input, process, and output for each step in this tutorial. Finally, after downloading the codes and data, open the t*ext_classification_tutorial.Rproj* file. The reader should then run the codes for every step as we go along, so as to be able to examine the input and the corresponding output.

Table 2.2. Text classification based on the input-process-output approach.

	Text Preprocessing						
	Text Preparation	**Text Cleaning**	**Text Transformation**	**Dimensionality Reduction**	**Classification**	**Evaluation**	**Validation**
Input	Raw html files	Output from Text Preparation	Output from Text Cleaning	Document-by-Term Matrix	Output from Dimensionality Reduction	Classification Model, test data, and an evaluation measure	Classification from the model
Process	Parsing, sentence segmentation	Punctuation, number, and stopword removal, lower case transformation	Word tokenization, constructing the Document-by-Term matrix where the words are the features and the entries are raw frequencies of the words in each document	Latent Semantic Analysis and/or supervised scoring methods for feature selection.	Apply classification algorithms such as Naive Bayes, Support Vector Machines, or Random Forest	Classify the documents in test data and compare with the actual labels. Calculate the value of the evaluation measure	Compute classification performance using an independent validation data set or compare the classification to the classification of domain experts
Output	Raw text file (one sentence per line)	Raw text file sentences where all letters are in lower cases and without punctuation, number and stopwords.	Document-by-Term matrix	Matrix where the columns are the new set of features or the reduced Document-by-term matrix	Classification Model	Value for the evaluation	Measure of agreement (one can quantify the agreement through the use existing evaluation measure)

Preparing Text

The input for this step consists of the raw German job vacancies. These vacancies were obtained from Monsterboard (url: www.monsterboard.nl) Since the vacancies are webpages, they are in Hypertext Markup Language (HTML), the standard mark-up language for representing content in web documents (Graham, 1995). Apart from the relevant text (i.e., content), raw HTML pages also contain elements used for layout. Therefore, a technique known as HTML parsing is used to separate the content from the layout.

In R, parsing HTML pages can be done using the **XML** package. This package contains two functions, namely, `htmlTreeParse()` that parses HTML documents and `xpathSApply()` that extracts specific content from parsed HTML documents. CL 1 (see the annotations in the file for further details as to what each command does), installs and loads the **XML** package, and applies the `htmlTreeParse()` and `xpathSApply()` functions. In addition, the contents of the HTML file *sample_nursing_vacancy.html* in the folder data is imported as a string object and stored in the variable htmlfile. Subsequently, this variable is provided as an argument to the `htmlTreeParse()` function. The parsed content is then stored in the variable rawpagehtml, which is in turn is the doc argument to the `xpathSApply()` function which searches for the tags in the text that we are interested in. In our case this text can be found in the div tag of the class content. Tags are keywords surrounded by angle brackets `(e.g., <div> and </div>).` The `xmlValue` in the `xpathSApply()` function means that we are obtaining the content of the HTML element between the corresponding tags. Finally the `writeLines()` function writes the text content to a text file named *sample_nursing_vacancy.txt* (in the folder parsed).

In order to extract text from several HTML files, the codes in CL 1 are put in a loop in CL 2. The function `htmlfileparser()` in CL 2 accepts two arguments and applies the procedures in CL 1 to each HTML file in a particular folder. The first argument is the name of the folder and the second argument is the name of the destination folder where the extracted text content is to be written. Supposing these html files are in the folder *vacancypages* and the extracted text content is to be saved in the folder *parsedvacancies*, these are the arguments we provide to `htmlfileparser()`. Expectedly, the number of text files generated corresponds to the number of HTML files, provided that all HTML files are well-defined (e.g., correct formatting). The text files comprise the output for this step.

Preprocessing Text

The preprocessing step consists of two stages. The first identifies sentences in the vacancies, since the sentence is our unit of analysis, and the second applies text

preprocessing operations on the sentences. We used sentences as our unit of analysis since our assumption is that the sentence is the right resolution level to detect job task information. We did not use the vacancy as our unit of analysis since a vacancy may contain more than one task. In fact if we chose to treat the vacancy as the unit of analysis it would still be important to identify which of the sentences contain task information. Another reason to select sentence as the unit of analysis is to minimize variance in document length. Input for the first stage are the text files generated from the previous step, and the output sentences from this stage serve as input to the second stage. CL 3 contains functions that can detect sentences from the parsed HTML file in the previous section *(i.e., sample_nursing_vacancy.html).*

The code loads the `openNLP` package. This package contains functions that run many popular natural language processing (NLP) routines including a sentence segmentation algorithm for the German language. Although the German sentence segmenter in `openNLP` generally works well, at times it may fail. Examining such failures in the output can provide ideas for the inclusion of new arguments in the algorithm. For example, if the segmenter encounters the words bzw. and inkl. (which are abbreviations of "or" and "including" respectively in German) then the algorithm will treat the next word as the start of a new sentence. This is because the algorithm has a rule that when there is a space after a period the next word is the start of a new sentence. In order to adjust for these and other similar cases, we created a wrapper function named `sent_tokens()`. Another function `sent_split()` searches for matches of the provided pattern within the string and when a match is found it separates the two sentences at this match. For example, some vacancies use bullet points or symbols such as "|" to enumerate tasks or responsibilities. To separate these items we supply the symbols as arguments to the function. Finally, once the sentences are identified the code writes the sentences to a text file where one line corresponds to one sentence.

For multiple text files, the codes should again be run in a loop. One large text file will then be generated containing the sentences from all parsed vacancies. Since we put all sentences from all vacancies in a single file, we attached the names of the corresponding text vacancy files to the sentences to facilitate tracing back the source vacancy of each sentence. Thus, the resulting text file containing the sentences has two columns: the first column contains the file names of the vacancies from which the sentences in the second column were derived.

After applying sentence segmentation on the parsed vacancy in *sample_nursing_vacancy.txt*, the sentences are written to the file *sentencelines_nursing_vacancy.txt* located in the folder *sentences_from_sample_vacancy.* The next task is to import the sentences into R so that additional preprocessing (e.g., text cleaning) can be performed. Other preprocessing steps that may be applied are lower case transformation, punctuation removal, number removal, stopword removal, and stemming. For this we use the tm package in R. This package automatically applies

word tokenization, so we do not need to create separate commands for that.

The sentences are imported as a data frame in R (see CL 4). Since the sentence is our unit of analysis, hereafter we refer to these sentences as documents. The first column is temporarily ignored since it contains only the names of the vacancy files. Since the sentences are now stored in a vector (in the second column of the data frame), the `VectorSource()` function is used. The source determines where to find the documents. In this case the documents are in `mysentences[,2].` If the documents are stored in another source, for example in a directory rather than in a vector, one can use `DirSource().` For a list of supported sources, invoke the function `getSources().` Once the source has been set, the next step is to create a corpus from this source using the `VCorpus()` function. In the tm package, corpus is the main structure for managing documents. Several preprocessing procedures can be applied to the documents once collected in the corpus. Many popular preprocessing procedures are available in this package. Apart from the existing procedures, users can also specify their own via user-defined functions. The procedures we applied are encapsulated in the `transformCorpus()f`unction. They include number, punctuation, and extra whitespace removal, and lower case conversion. We did not apply stemming since previous work recommends not to use stemming for short documents (H.-F. Yu et al., 2012). The output consists of the cleaned sentences in the corpus with numbers, punctuation, and superfluous whitespaces removed.

Text Transformation

CodeListing 5 details the structural transformation of the documents. The input in this step is the output from the preceding step (i.e., the cleaned sentences in the training data). In order to quantify text characteristics, we use the vector space model because this is the simplest and perhaps most straightforward approach to quantify text and thus forms an appropriate starting point in the application of TC (Frakes & Baeza-Yates, 1992; Salton et al., 1975). For this transformation, the `DocumentTermMatrix()` of the tm package has a function that may be used to build features based on the individual words in the corpus.

The `DocumentTermMatrix()` function transforms a corpus into a matrix where the rows are the documents, the columns are features, and the entries are the weights of the features in each document. The default behavior of the `DocumentTermMatrix()` function is to ignore terms with less than 3 characters. Hence, it is possible that some rows consist entirely of 0s because after preprocessing it may be the case that in some sentences all remaining terms have less than 3 characters. The output in this step is the constructed DTM. This matrix is then used as a basis for further analysis. We can further manipulate the DTM, for instance by adjusting the weights.

We mentioned previously that for word features one can use raw counts

as weights. The idea of using raw counts is that the higher the count of a term in a document the more important it is in that document. The `DocumentTermmatrix()` function uses the raw count as the default weighting option. One can specify other weights through the weighting option of the control argument. To take into account documents sizes, for example, we can apply a normalization to the weights although in this case it is not an issue because sentences are short.

Let us assign a "weight" to a feature that reflects its importance with respect to the entire corpus using the DF. Another useful feature of DF is that it provides us with an idea of what the corpus is about. For our example the word with the highest DF (excluding stopwords) is plege (which translates to "care") which makes sense because nursing is about the provision of care. Terms that are extremely common are not useful for classification.

Another common text analysis strategy is to find keywords in documents. The keywords may be used as a heuristic to determine the most likely topic in each document. For this we can use the TF-IDF measure. The keyword for each document is the word with the maximum TF-IDF weight (ties are resolved through random selection). The codes in CL 6 compute the keyword for each document. For example, the German keyword for Document 4 is `aufgabenschwerpunkte` which translates in English to "task focal points".

The final DTM can be used as input to dimensionality reduction techniques or directly to the classification algorithms. The process from text preprocessing to text transformation culminated in the DTM that is depicted in Figure 2.3.

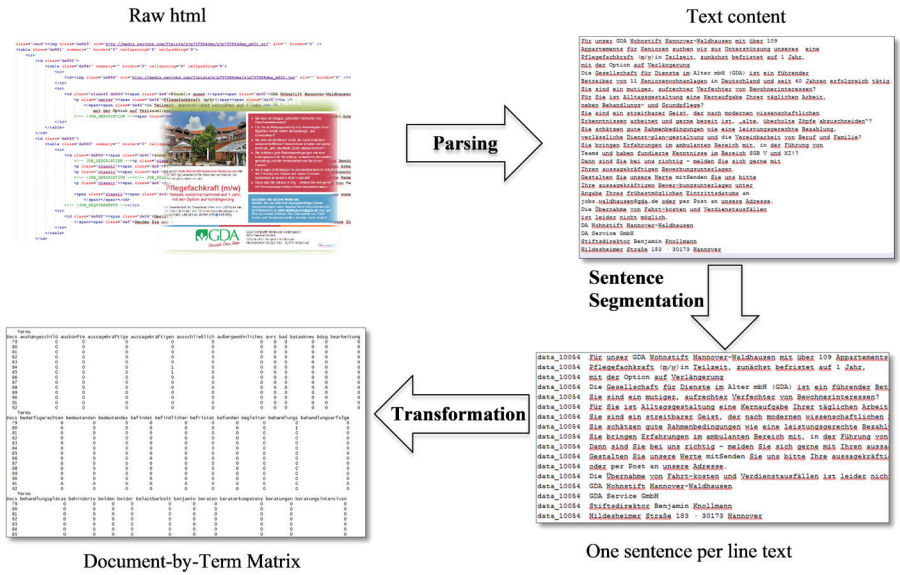


Figure 2.3. Illustration of text preprocessing from raw HTML file to Document-by-Term matrix.

Dimensionality Reduction

Before running classification algorithms on the data, we first investigate which among the features are likely most useful for classification. Since the initial features were selected in an ad hoc manner, that is, without reference to specific background knowledge or theory, it may be possible that some of the features are irrelevant. In this case, we applied dimensionality reduction to the DTM.

LSA is a commonly applied to reduce the size of the feature set (Landauer et al., 1998). The output of LSA yields new dimensions which reveal underlying patterns in the original features. The new features can be interpreted as new terms that summarize the contextual similarity of the original terms. Thus, LSA partly addresses issues of synonymy and in some circumstances, polysemy (i.e., when a single meaning of a word is used predominantly in a corpus). In R, the `lsa` package contains a function that runs Latent Semantic Analysis.

To illustrate LSA we need additional vacancies. For illustrative purposes we used 11 job vacancies (see the *parsedvacancies* folder). We applied sentence segmentation to all the vacancies and obtained a text file containing 425 sentences that were extracted from the 11 vacancies (see the *sentences_from_vacancies* folder). After applying preprocessing and transforming the sentences in sentenceVacancies.txt into a DTM, we obtained 1079 features and retained 422 sentences. We selected all sentences and ran LSA on the transposed DTM (i.e., the Term-by-Document Matrix) (see CL7). We applied normalization to the term frequencies to minimize the effect of longer sentences.

Documents and terms are projected onto the constructed LSA space in the `projdocterms` matrix. The entries in this matrix are readjustments of the original entries in the Term-by-Document Matrix. The readjustments take into account patterns of co-occurrence between terms. Hence, terms which often occur together will roughly have the same values in documents where they are expected to appear. We can apply the cosine measure to identify similar terms. Similarity is interpreted in terms of having the same pattern of occurrence. For example, terms which have the same pattern of occurrence with `sicherstellung` can be found by running the corresponding commands in CL 8.

The German word `sicherstellung` (which means "to guarantee" or "to make sure" in English) is found to be contextually similar to `patientenversorgunng` (patient care) and `reibungslosen` (smooth or trouble-free) because these two words appeared together with `sicherstellung` (to guarantee) in the selected documents. Another interesting property of LSA is that it can uncover similarity between two terms even though the two terms may never be found to co-occur in a

single document. Consider for example the word `koordinierung` (coordination), we find that `kooperative` (cooperative) is a term with which it is associated even though there is not one document in the corpus in which the two terms co-occur. This happens because both terms are found to co-occur with `zusammenarbeit` (collaboration), thus when either one of the terms occurs then LSA expects that the other should also be present. This is the way LSA addresses the issue of synonymy and polysemy. One can also find the correlation among documents and among terms by running the corresponding commands in CL 8.

Since our aim is to reduce dimensionality, we project the documents to the new dimensions. This is accomplished through the corresponding codes in CL 8. From the LSA, we obtain a total of 107 new dimensions from the original 1079 features. It is usually not easy to attach natural language interpretations to the new dimensions. In some scenarios, we can interpret the new dimension by examining the scaled coefficients of the terms on the new dimensions (much like in PCA). Terms with higher loadings on a dimension have a greater impact on that dimension. Figure 2.4 visualizes the terms with high numerical coefficients on the first 6 LSA dimensions (see CL 8 for the relevant code). Here we distinguish between terms found to occur in a task sentence (red) or not (blue). In this way, an indication is provided of which dimensions are indicative for each class (note that distinguishing between tasks and non-tasks requires the training data, which is discussed in greater detail below).

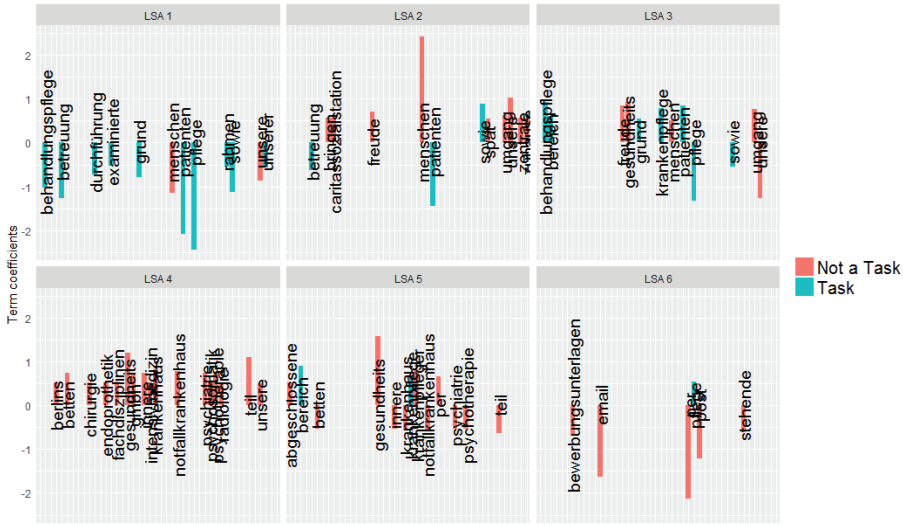


Figure 2.4. Loadings of the terms on the first 6 LSA dimensions using 422 sentences from 11 vacancies.

Another approach that we could try is to downsize the feature set by eliminating those features that are not (or less) relevant. Such techniques are collectively called filter methods (Guyon & Elisseeff, 2003). They work by assigning scores to features and setting a threshold whereby features having scores below the threshold are filtered out. Both the DF and IDF can be used as scoring methods. However, one main disadvantage of DF and IDF is that they do not use class membership information in the training data. Including class membership (i.e., through supervised scoring methods) ought to be preferred, as it capitalizes on the discriminatory potential of features (Lan et al., 2009).

For supervised scoring methods, we need to rely on the labels of the training data. In this example, the labels are whether a sentence expresses task information (1) or not (0). These labels were obtained by having experts manually label each sentence. For our example, experts manually assigned labels to the 425 sentences. We applied three scoring methods, namely, Information Gain, Gain ratio, and Symmetric Uncertainty (see CL 12). Due to the limited number of labeled documents, these scoring methods yielded less than optimal results. However, they still managed to detect one feature that may be useful for identifying the class of task sentences, that is, the word `zusammenarbeit` (collaboration), as this word most often occurred in task sentences. The output from this step is a column-reduced matrix that is either the reduced version of the DTM or the matrix with the new dimensions. In our example we applied LSA and the output is a matrix in which the columns are the LSA dimensions.

Classification

The reduced matrix from the preceding section can be used as input for classification algorithms. The output from this step is a classification model which we can then use to automatically classify sentences in new vacancies. We have mentioned earlier that reducing dimensionality is an empirically driven decision rather than one which is guided by specific rules of thumb. Thus, we will test whether the new dimensions lead to improvement in performance as compared to the original set by running separate classification algorithms, namely Support Vector Machines (SVMs), Naive Bayes, and Random Forest, on each set. These three have been shown to work well on text data (Dong & Han, 2004; Eyheramendy et al., 2003; Joachims, 1998).

Accuracy is not a good performance metric in this case since the proportion of task sentences in our example data is low (less than 10%). The baseline accuracy (computed from the model which assigns all sentences to the dominant class), would be 90% which is high, and thus difficult to improve upon. More suitable performance metrics are the F-measure (Ogura et al., 2011; Powers, 2011) and balanced accuracy

(Brodersen et al., 2010). We use these two measures here since the main focus is on the correct classification of task sentences and we also want to control misclassifications (non-task sentences put into the task class or task sentences put into the non-task class).

In assessing the generalizability of the classifiers, we employed 10 times 10-fold cross-validation. We repeated 10-fold cross-validation 10 times because of the limited training data. We use one part of the data to train a classifier and test its performance by applying the classifier on the remaining part and computing the F-measure and balanced accuracy. For the 10 times 10-fold cross-validation, we performed 100 runs for each classifier using the reduced and original feature sets. Hence, for illustration we ran about 600 trainings since we trained 6 classifiers in total. All performance results reported are computed using the test sets (see CL 10).

From the results we see how classification performance varies across the choice of features, classification algorithms, and evaluation measures. Figure 2.5 presents the results of the cross-validation. Based on the F-measure, Random Forest yielded the best performance using the LSA reduced feature set. The highest F-measure obtained is 1.00 and the highest average F-measure is 0.40 both from Random Forest. SVM and Naive Bayes have roughly the same performance. This suggests that among the three classifiers Random Forest is the best classifier using the LSA reduced feature set, and F-measure as the evaluation metric. If we favor the correct detection of task sentences and we want a relatively small dimensionality, then Random Forest should thus be favored over the other methods. For the case of using the original features, SVM and Random Forest exhibit comparable performance. Hence, when using F-measure and the original feature set, either SVM or Random Forest would be the preferred classifier. The low values of the F-measures can be accounted for by the limited amount of training data. For each fold, we found that there are about 3-4 task sentences, thus a single misclassification of a task sentence leads to sizeable reduction in precision and recall which in turn results in a low F-measure value.

When balanced accuracy is the evaluation measure, SVM and Random Forest consistently yield similar performance when using either the LSA reduced feature set or the original feature set, although, Random Forest yielded a slightly higher performance compared to SVM using the LSA reduced features set. This seems to suggest that for balanced accuracy and employing the original features, one can choose between SVM and Random Forest, and if one decides to use the LSA feature set then Random Forest is to be preferred. Moreover, notice that the numerical value for balanced accuracy is higher than F-measure. Balanced accuracy can be increased by the accuracy of the dominant class, in this case the non-task class.

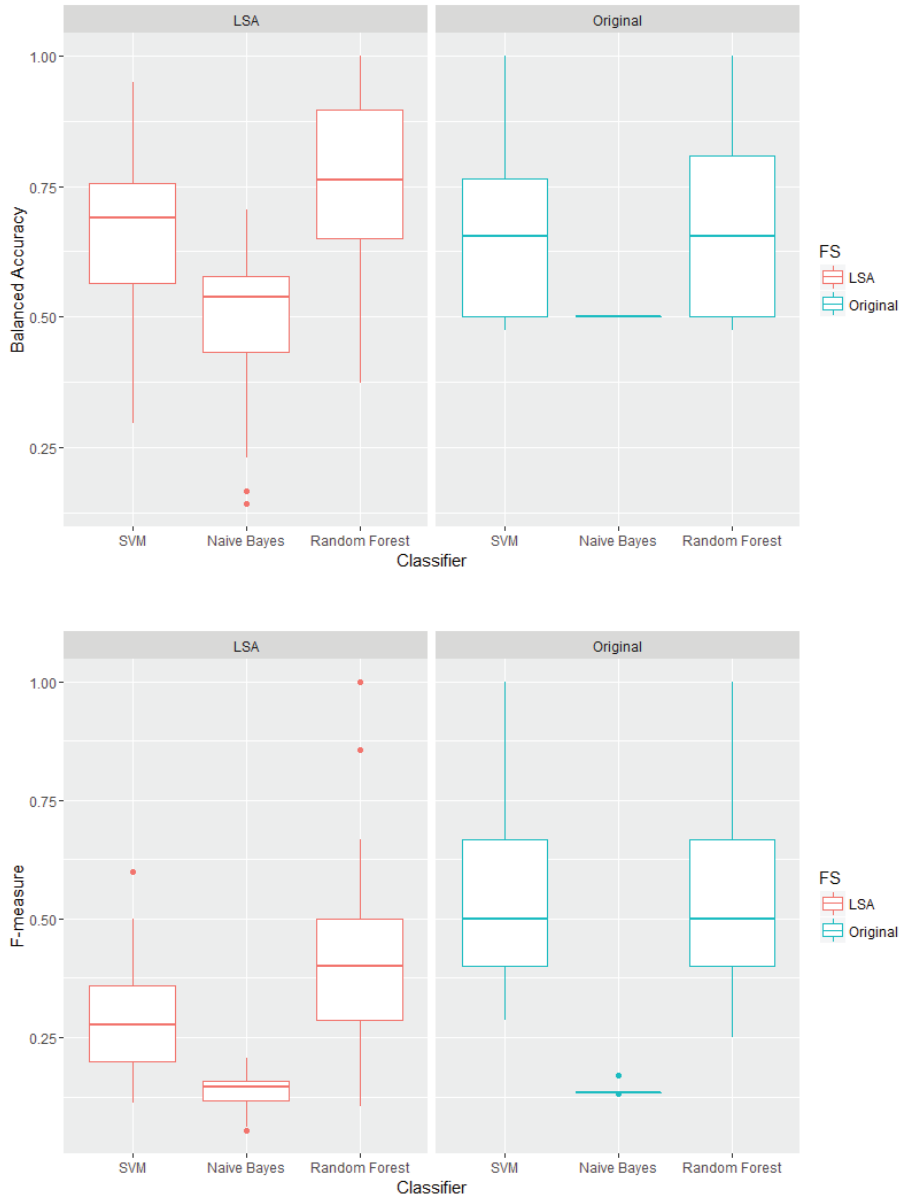


Figure 2.5. Comparison of classification performance among three classifiers and between the term-based and LSA-based features.

This classification example reveals the many issues that one may face in building a suitable classification model. First is the central role of features in classification. Second is how to model the relationship between the features and the class membership. Third is the crucial role of choosing an appropriate evaluation measure or performance metric. This choice should be guided by the nature of the problem, the objectives of the study, and the amount of error we are willing to tolerate. In our example, we assign equal importance to both classes, and we therefore have slight preference for balanced accuracy. In applications where the misclassification cost for the positive class is greater than that for the other class, the F-measure may be preferred. For a discussion of alternative evaluation measures see Powers (2011).

Other issues include the question of how to set a cut-off value for the evaluation measure to judge whether a model is good enough. A related question is how much training data is needed for the classification model to generalize well (i.e., how to avoid overfitting). These questions are best answered empirically through systematic model evaluation, such as by trying different training sizes and varying the threshold, and then observing the effect on classifier performance. One strategy is to treat this as a factorial experiment where the choices of training size and evaluation measures are considered as factor combinations. Additionally, one has to perform repeated evaluation (e.g., cross-validation) and validation. Aside from modeling issues there are also practical concerns such as the cost of acquiring training data and the interpretability of the resulting model. Classification models with high predictive performance are not always the ones that yield the greatest insight. Insofar as the algorithm is to be used to support decision making, the onus is on the researcher to be able to explain and justify its workings.

Classification for Job Information Extraction

For our work on Job Task information extraction three people hand labeled a total of 2,072 out of 60,000 sentences. It took a total of 3 days to label, verify and relabel 2,072 sentences. From this total, 132 sentences were identified as task sentences (note that the task sentences were not unique). The proportion of task sentences in vacancy texts was only 6%. This means that the resulting training data is imbalanced. This is because not all tasks that are part of a particular job will be written in the vacancies, likely only the essential and more general ones. This partly explains their low proportion.

Since labeling additional sentences will be costly and time-consuming we employed a semi-supervised learning approach called label propagation (Zhu & Ghahramani, 2002). For the transformation and dimensionality reduction we respectively constructed the DTM and applied LSA. Once additional task sentences were obtained via semi-supervised learning we ran three classification algorithms, namely, SVM, Random Forest, and Naive Bayes. Instead of choosing a single classifier we combined the predictions of the three in a simple majority vote. For the

evaluation measure we used the Recall measure since we wanted to obtain as many task sentences as possible. Cross-validation was used to assess the generalization property of the model. The application of classification resulted to identification of 1179 new task sentences. We further clustered these sentences to obtain unique nursing tasks since some sentences pointed to the same tasks.

Model Reliability and Validity

We set out to build a classification model that can extract sentences containing nursing tasks from job vacancies. Naturally, a subsequent step is to determine whether the extracted task sentences correspond to real tasks performed by nurses. An approach to establish construct validity is to use an independent source to examine the validity of the classification. Independent means that the source should be blind from the data collection activity, initial labeling procedure, and model building process. Moreover, in case ratings are obtained, these should be provided by Subject Matter Experts (SME's), that is, individuals who have specialist knowledge about the application domain. If found to be sufficiently valid the extracted sentences containing job tasks may then be used for other purposes such as in job analysis, identifying training needs or developing selection instruments.

We enlisted the help of SME's, presented them the task sentences predicted by the text classifier, and asked them to check whether the sentences are actual nursing tasks or not so as to be able to compute the precision measure. Specifically, we compute precision as the ratio of the number of sentence tasks confirmed as actual nursing tasks to the total number of sentences tasks predicted by the model. We reran the classification algorithm in light of the input from the experts. The input is data containing the correct label of sentences which were misclassified by the classifier. We performed this in several iterations until there was no significant improvement in precision. This is necessarily an asymmetric approach since we use the expert knowledge as the "ground truth".

A more elaborate approach would be to compare the extracted tasks from vacancies to tasks collected using a more traditional job analysis method, namely a task inventory. The task inventory would consist of interviews and observations with subject matter experts (SMEs) to collect a list of tasks performed by nurses. Based on this comparison, a percentage of tasks would be found in both lists, a percentage of unique tasks would only be found in the task inventory, and a percentage of unique tasks would only be found in the online vacancies. A high correspondence between the list of task collected by text mining and the list of tasks collected in the task inventory (which would be considered to accurately reflect the nursing job) could be taken as evidence for convergent validity. Converely, one could establish discriminant validity, or a very low correspondence with so called 'bogus tasks' that are completely unrelated to the nursing job.

We apply the less elaborate approach by first training a classification model, making predictions using the model, and presenting the task sentences to an

SME. The expert judged whether the sentences are actual nursing tasks or not. The precision measure was used to give an indication of the validity of the model. The first round of validation resulted in a precision of 65% (precision range: 0% to 100%) and we found out that some of the initial labels we assigned did not match the labels provided by the independent expert (that is some of the labels in the initial labels were judged to be erroneous by the expert). In light of this, we adjusted the labels and conducted a second round of validation in which precision increased to 89%. This indicates that we gained classification validity in the classification model. A total of 91 core tasks were validated. Table 2.3 contains validated tasks under the basic care and medical care clusters. In practice, it is difficult to obtain 100% precision since forcing a model to give high precision comes at the expense of sacrificing its recall. High precision and low recall imply the possibility that many task sentences will be dismissed though we can put more confidence on the sentences that are labeled as a task. As a last note, TC models are seldom static, that is, as new documents arrive, we have to continually assess the performance of the model on new observations and adjust our model if there is significant degradation in performance.

Table 2.3. Basic care and medical care core nursing tasks extracted from nursing vacancies by applying text classification.

Task	German Translation	Task Cluster
Monitoring the patients' therapy	Überwachung der Therapie des Patienten	Basic Care
Caring for the elderly	Pflege von älteren Menschen	Basic Care
Providing basic or general care	Durchführung der Allgemeinen Pflege	Basic Care
Providing palliative care	Durchführung von Palliativpflege	Basic Care
Caring for mentally ill patients	Pflege von psychisch kranken Menschen	Basic Care
Caring for children	Pflege von Kindern	Basic Care
Assisting at intake of food	Hilfe bei der Nahrungsaufnahme	Basic Care
Supporting of rehabilitation	Unterstützung der Rehabilitation	Basic Care
Providing holistic care	Durchführung ganzheitlicher Pflege	Basic Care
Accompanying patients	Begleitung von Patienten	Basic Care
Assisting at surgical interventions	Assistenz bei operativen Eingriffen	Medical Care
Doing laboratory tests	Durchführung von Labortests	Medical Care
Participating in resuscitations	Beteiligung an Reanimationsmaßnahmen	Medical Care
Conducting ECG	Durchführung von EKG	Medical Care
Collecting blood	Durchführung der Blutabnahme	Medical Care
Preparing and administer intravenous drugs	Vorbereitung und Verabreichung von intravenösen Medikamenten	Medical Care
Assisting at diagnostical interventions	Assistenz bei diagnostischen Maßnahmen	Medical Care
Operating the technical equipment	Bedienung der technischen Geräteschaften	Medical Care
Assisting at endoscopic tests	Assistenz bei endoskopischen Maßnahmen	Medical Care
Assisting at examination	Assistenz bei Untersuchungen	Medical Care

Conclusion

This paper provided an overview of TC and a tutorial on how to conduct actual TC on the problem of job task information extraction from vacancies. We discussed and demonstrated the different steps in TC and highlighted issues surrounding the choices of features, classification algorithms, and evaluation metrics. We also outlined ways to evaluate and validate the resulting classification models and prediction from these models. Text classification is an empirical enterprise where experimentation with choices of representation, dimensionality reduction and classification techniques are standard practice. By building several classifiers and comparing them, the final classifier is chosen based on repeated evaluation and validation. Thus, text classification is not a linear process; one has to revisit each step iteratively to examine how choices in each step affect succeeding steps. Moreover, classifiers evolve in the presence of new data. TC is a wide research field and there are many other techniques that were not covered here. An exciting new area is the application of deep learning techniques for text understanding (for more on this we refer the reader to Maas et al., 2011; Mikolov, Chen, Corrado, & Dean, 2013; X. Zhang & LeCun, 2015).

TC models are often descriptive as opposed to explanatory in nature, in the sense that they capture the pattern of features and inductively relate these to class membership (Bird et al., 2009). This contrasts with explanatory models whose aim is to explain why the pattern in features leads to the prediction of a class. Nevertheless, the descriptive work can be of use for further theory building too as the knowledge of patterns can be used as a basis for the development of explanatory models. For example, in the part about feature selection we found out that the word sicherstellung (to guarantee or to make sure) is useful in detecting sentences containing nursing tasks. Based on this we can define the concept of "task verb", that is, a verb that is indicative of a task in the context of job vacancy. We could then compile a list of verbs that are "task verbs" and postulate that task verbs pair with noun or verb phrases to form task sentences. Further trials could then be designed to validate this concept and establish the relationship between features and patterns. In this way, we are not only detecting patterns but we also attempt to infer their properties and their relationship to class membership.

Whether a descriptive model suffices or whether an explanatory model is needed depends on the objectives of a specific study. If the objective is accurate and reliable categorization (e.g., when one is interested in using the categorized text as input to other systems) then a descriptive model will suffice although the outcomes still need to be validated. On the other hand, if the objective is to explain how patterns lead to categorization or how structure and form lead to meaning then an explanatory model is required.

In this paper we tried to present TC in such a manner that organizational

researchers can understand the underlying process. However, in practice, organizational researchers will often work with technical experts to make choices on the algorithms and assist in tweaking and tuning the parameters of the resulting model. The role of organizational researchers then is to provide research questions, help select the relevant features, and provide insights considering the classification output. These insights might lead to further investigation and ultimately to theory development and testing.

Finally, we conclude that TC offers great potential to make the conduct of text-based organizational research fast, reliable, and effective. The utility of TC is most evident when there is a need to analyze massive text data, in fact in some cases TC is able to recover patterns that are difficult for humans to detect. Otherwise, manual qualitative text analysis procedures may suffice. As noted, the increased use of TC in organizational research will likely not only contribute to organizational research, but also to the advancement of TC research, because real problems and existing theory can further simulate the development of new techniques.

PART 2

TEXT MINING APPLICATIONS IN JOB ANALYSIS

A Fresh Perspective on Job Analysis: Extracting Job Information from Job Vacancy

In this Chapter, we describe the development of an automatic method for extracting job information from job vacancies. Specifically, we are interested in sorting vacancy content into work activities and work attributes. For this, we engineered features and tested several machine learning classifiers. We address the following research questions:

RQ1 What features are useful to identify work activities and worker attributes and can these features be combined to build an automatic and accurate classifier, one that can distinguish between work activities and worker attributes contained in job vacancies?

RQ2 Can the extracted work activities be used to construct task groups and meaningfully cluster jobs based on tasks?

RQ3 Can the task groups be validated by comparing it to tasks as enumerated in the European Skills, Competences, and Occupations (ESCO)?

The collection and analysis of work-related information is called Job Analysis (Sackett & Laczo, 2003b). The purpose of conducting job analysis may either be one or a combination of the following human resource management activities: selection (Jeanneret & Strong, 2003), job design (Berg et al., 2013), training (Arthur Jr et al., 2003), and compensation (Siddique, 2004; P. Singh, 2008), among others. The types of information for job analysis fall into one of the three categories (also known as *job descriptors*) (Campion et al., 1999). The first category is work activities (or work data), which includes tasks and responsibilities. The second category is worker attributes, which are characteristics of workers required to successfully discharge the tasks and perform their responsibilities. Examples of worker attributes are knowledge, skills, abilities, and other characteristics (KSAOs). The third category is the work context defined as the physical and social factors that influence the nature of work, and which may include the context of work, and the physical work environment (Morgeson & Dierdorff, 2011).

Data relevant for job analysis are traditionally collected through observation, interviews, and questionnaires and surveys administered to job incumbents or subject matter experts (Brannick et al., 2007a; Sackett & Laczo, 2003a). However, most of these methods are time consuming, suffer from reporting bias, lacks reliability (Lindell et al., 1998) and accuracy (Dierdorff & Wilson, 2003), and limit inter- and intra-job comparisons. Although job analysis researchers have tried to address these challenges with varying degrees of success, there are also emerging challenges deriving from new realities of the job market such as globalization, technological

development, and alternative work arrangements.

Other rich sources of job data are documentations such as job descriptions, published information about the job, and previous work analyses (Brannick et al., 2007a). Documentations have the distinct advantages of being easy to collect once access has been granted, and data sets need a moderate number of preprocessing steps since information has been previously categorized. Furthermore, job information obtained in this way has the advantage of being relatively easy to replicate. There are also disadvantages though, such as information being outdated (for example, it may have been a long time since the information was last collected about a particular job) and/or have insufficient depth or breadth (P. Singh, 2008).

New ways of gathering information that are rapid, reliable, and valid may help mitigate some of the challenges. Comprehensive job analysis requires not only an analysis at the basic level of organization, but also one done across organizations and even countries to get a general overview of the roles, responsibilities and requirements associated with each job. Also, as some jobs change fast, methods should be able to keep track of and analyze how jobs evolve, to inform educational curriculum changes and employee training. A novel source of job information has become apparent in recent years, one that could help mitigate the above challenges. These are online job vacancies.

Job vacancies posted to various online employment platforms (Duggan, 2010; Malinowski, Keim, Wendt, et al., 2006; Mang, 2012) present an opportunity to greatly enhance job analysis for the following reasons. First, in contrast to traditional sources of job information, vacancies are numerous, inexpensive, relatively easy to obtain (once access has been granted) and likely to contain up-to-date information about jobs (Amato et al., 2015; Harper, 2012). Second, since vacancies are written for different jobs and originate from various organizations, vacancies permit the comparison of jobs within and across organizations (Addom et al., 2011; Boselli et al., 2017; Russ et al., 2016). Third, vacancies can be compared and analyzed over time, which facilitates the understanding of how different jobs evolve in terms of required skills, and activities performed (Albitz, 2002) and offer the potential to capture job changes in the workplace (Addom et al., 2011; Albitz, 2002; Russ et al., 2016; Sanchez & Levine, 2012b). Fourth, by aggregating vacancies for the same job, idiosyncratic 'noise' (i.e., job irrelevant information) due to regional, temporal, and/or organizational circumstances can be filtered out in an effort to provide comprehensive, generalizable, and valid information about that job. Fifth, besides complementing job analysis, vacancies may also be used as a rich source of information for labor market analysis or curriculum planning, through the provision of relevant labor market information (e.g., projected compensation and occupational outlook) (Dusi et al., 2015; Kobayashi et al., 2014). For example, vacancies have been used to inform education and training policies by comparing and forecasting the skills demand across European labor markets (Cedefop, 2019) and to provide insights into

the matching of skills and jobs (Messum et al., 2017). And sixth, job vacancy data can be easily matched with other web based data, promoting interlinking of data and rich knowledge representations (Hosen & Alfina, 2016) .

In light of the potential of job vacancies to revitalize job analysis, from a methodological perspective, the key challenge is how to effectively and efficiently extract job information, particularly in light of the fact that manual extraction of such information is time consuming, requires a lot effort, and may indeed be downright impractical (Cedefop, 2019; Kureková et al., 2013). Fortunately, with advancements in computing and text mining, the extraction of job information from vacancies can, at least in part, be automated leading to efficiency, reliability, and replicability (McEntire et al., 2006). We refer to the automatic extraction of job information from vacancies as Vacancy Information Extraction (VIE).

Most proposed algorithms for VIE rely on keyword search, which involves prespecifying a set of words and developing an algorithm to search and count those words as they appear in vacancies (Addom et al., 2011; Dusi et al., 2015; Fabo et al., 2017; Fabo & Kahanec, 2020; Sodhi & Son, 2010). Although keywords can be useful in tasks such as determining the frequency of certain skills or competences, they are inadequate for other tasks such as extracting new or emerging skills or competencies. Another disadvantage of using keywords is that many skills cannot be captured in a single word (e.g., full stack development). Of course, there is always the option of extending the analysis to bigrams (or trigrams), however, this may introduce unnecessary complexity in the analysis. Finally, there may be other pertinent information encapsulated in vacancies such as specifics about job roles and responsibilities, for which keywords may also not be the best choice because keyword alone cannot highlight contextual differences, for example, the difference between *using* and *developing* a web development framework.

Although the potential of using vacancies is nontrivial, questions pertaining to the accuracy, reliability, quality, and validity of the information from vacancies must be addressed. Hence, the validity of information extracted from vacancies will need to be established (Kobayashi et al., 2018). Integrating different sources of information through data triangulation will establish a construct validity evidence (Binning & Barrett, 1989b; Sanchez & Levine, 2000b).

In this paper, different means of automating the extraction and sorting of job information from vacancies using text classification procedures are discussed and compared. The paper also shows how the extracted information can be applied to job analysis and labor market analysis. The following are demonstrated: (1) how the extracted job information can be applied to automatically determine knowledge domains and skill sets that in turn may be used to create job taxonomies, and (2) how to validate the extracted information by comparing it to an existing taxonomy of job information, specifically the European Skills, Competences, Qualifications and Occupations (ESCO).

Research Questions and Objectives

We asked the following research questions:

RQ1 What features are useful to identify work activities and worker attributes and can these features be combined to build an automatic and accurate classifier that can distinguish between work activities and worker attributes contained in job vacancies?

RQ2 Can extracted work activities be used to construct task groups and meaningfully cluster jobs based on tasks?

RQ3 Can the extracted job information be validated by comparing it to tasks as enumerated in the ESCO?

To address the research questions above, we specify the following primary and specific objectives.

The primary objectives of this paper were (i) to explore and compare different features and different text classification algorithms, to sort job information from vacancies into work activities and worker attributes, and to show how the extracted work activity information can be used to identify task groups and to cluster jobs. Knowledge about tasks performed in the job could be used for recruitment (e.g., person-job matching), for micro-level analysis of how employers perceive a job, and to compare labor markets in terms of job tasks and the skills required to perform the tasks (Kureková et al., 2013). Other applications include curriculum planning (i.e., injecting in curricula training that are relevant to the labor market) and job matching by public employment services (Messum et al., 2017). The latter could also encourage unemployed people to go back to work or increase participation in the labor market (Zhou et al., 2016) Job clustering is useful for creating job taxonomies and occupational coding, which may be applied to determining compensation, training needs analysis, career path and succession planning (Colihan & Burger, 1995; Denisi, 1976; Taylor, 1978). Moreover, job clustering is a good starting point to study for vacancy driven job mobility.

To accomplish these objectives, we defined the following roughly sequential and iterative work packages (which are discussed in greater detail below):

1) Engineering features useful for the classification of job information.
2) Developing and trialing different classification algorithms to categorize information from job vacancies.
3) Applying topic modeling to determine task groups from the extracted work activities.
4) Applying a hierarchical clustering algorithm to cluster jobs using the output from the topic modeling.

Text classification techniques for vacancy extraction

Most studies on vacancy mining to date have used predefined keywords to extract job information. For example, IT skills are extracted from vacancies by leveraging pre-selected keywords related to computer programming (e.g., C++, Visual Basic, etc.) (Smith & Ali, 2014). Other studies have examined the trends in required job skills for IT professionals (Gallivan et al., 2002) and required computer skills across jobs (Fabo & Kahanec, 2020). These researchers established that vacancies in IT tend to be asymmetrically focused on technical skills despite the increasing emphasis on hiring well rounded individuals with business knowledge and proficiency in "soft skills". Nevertheless, online job ads appear to list a stronger mix of technical, humanistic, and business skills compared to what can be obtained from other sources (Huang et al., 2009).

One limitation of using keyword approach is that it does not explicitly distinguish between work activity and worker attribute job information. The distinction is important since job-analytic data need to be evaluated according to the reliability and validity of the inferences derived from them. Work activities are considered observable aspects of jobs useful for objectively measuring job performance, whereas worker attributes are construals that are determinants of successful job performance and are usually useful during recruitment and for explaining workers' behavior. Another limitation is that the keyword approach may fail to detect new skills or competencies and may entirely ignore work activities. Due to the preceding limitations, the approach that was taken here was to cast the problem of vacancy information extraction as a text classification problem, that is, contents of a vacancy are first sorted into work activities, worker attributes, and others (or the rest category).

For the classifier, we experimented with several classification modelling techniques, starting with classical ones, namely *Penalized Logistic Regression, Random Forest, Gradient Boosting Machines, Feed Forward Neural Network* and an ensemble technique called Stacking. We then compared the performance of the aforementioned 5 classifiers to a more recent deep learning technique called *Bidirectional Encoder Representation from Transformers* (BERT) (Devlin et al., 2018). These classifiers were chosen because they have been shown to demonstrate adequate performance in many classification tasks (Baskin et al., 2017; Kowsari et al., 2019; Ramraj et al., 2018; Thangaraj & Sivakami, 2018). For the Penalized Logistic Regression and Random Forest methods, we leveraged the approaches as explained in Kobayashi et al. (2018a). The Gradient Boosting Machines, Feed Forward Neural Network, and BERT are discussed next.

Gradient Boosting Machine

Gradient Boosting Machine (GBM) is like Random Forest in that both are ensemble techniques that combine several weak classifiers to come up with a strong learning algorithm and that both use decision trees as the base learners. Here we use a

type of GBM called Gradient Boosted Trees. The difference is that whereas Random Forest uses bagging to combine several decision trees, GBM is based on iteratively training classifiers, wherein each classifier is an improvement of its predecessor.

Feed Forward Neural Network

Feed Forward Neural Network or simply Neural Network (NN) is another classification technique that is loosely inspired on our current understanding of how the human brain works. It consists of interconnected nodes that form layers. A neural network usually consists of an input layer, 1 or more hidden layers, and one output layer (see Figure 3.1). Training a neural network model comes down to estimating the weights among node connections.

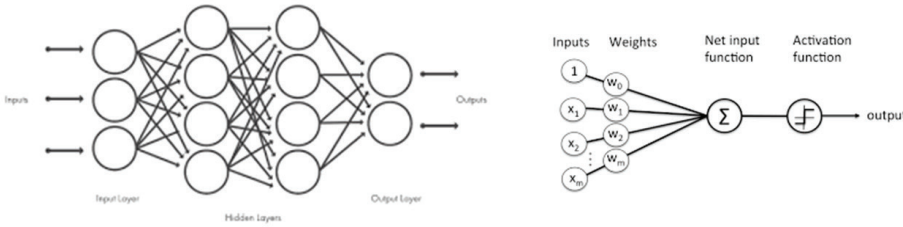


Figure 3.1. (Left) Architecture of a neural network model with two hidden layers and (Right) illustration of a single neuron or node.

In building NN several considerations come into play. First is the architecture of the neural network which typically refers to the number of hidden layers and the number of nodes per layer. Another is the transformation function, also called the activation function, to be used in the nodes. In terms of learning the weights, we need to set several hyperparameters depending on what type of algorithm is used. The most fundamental hyperparameters are the learning rate, the *adaptive rate, rho, epsilon* and *momentum start*. These hyperparameters determine how fast the learning would converge and how good are the weight estimates. Other hyperparameters such as i*nput dropout ratio, L1 and L2 regularization parameters*, and the l*earning rate annealing* pertain to improving the generalization ability of the neural network.

BERT-Logistic regression classifier

BERT is a Transformer encoder stack, which is basically an architecture inspired by a transformer (which is a type of deep neural network architecture) but only uses stacked encoders (a series of self-attention and feed-forward neural networks). The input for BERT is a sequence of words, in our case the sentence, and the output consists of contextualized embeddings of each word in the input sequence.

The first input token called CLS is appended at the start of each word sequence. Also, the output embedding for CLS, which is a vector of certain size, is the input for the chosen classifier. One can choose any classifier such as another Feed Forward Neural Network with softmax output layer or a logistic regression. In this study, we chose a logistic regression model since it is a popular type of choice classification. Furthermore, we used a pretrained BERT model called bert-base-cased (Devlin et al., 2018).

Classification of Job Information Types in English vacancies

In this section we discuss the development of machine learning based classifiers for the extraction and classification of job information from English online job vacancies

Data collection and Preprocessing

Vacancy data were obtained from various employment websites. The vacancies were subsequently parsed sorted into the following fields: *title, location, type of employment* (full-time vs part-time), *name of employer, name of industry,* and *job description.* Our focus was on the job description field, which contains job activities and candidate attributes required from the applicants. These vacancies were provided to us in Hypertext Markup Language (HTML) format and hence they needed to be preprocessed to get at the content to the task at hand. Standard HTML parsing techniques were employed (Burke, 2002; *Multilingual CV and Job Parsing,* n.d.).

Training data preparation

We collected 100,000 vacancies. The gathered vacancies were in different languages, but we retained only the English vacancies. In total there were 47,856 English vacancies. Our analysis operates at the sentence level. Identifying job information at the sentence level allows for more flexibility than using the word or phrase level, in terms of the job information that can be extracted. For example, in extracting worker attributes, the information extracted is not limited by predefined keywords (Smith & Ali, 2014) and may allow the identification of new worker attributes. Hence, we needed to identify the sentences and words which were achieved by applying sentence and word segmentation algorithms.

In the English language, words are typically separated by white space characters and sentences start with an upper-case letter and end with full stops. Additional rules were specified to capture special cases, such as words connected by "/" (as in "send/receive" above) and proper nouns that always start with upper case letters even when they do not begin a sentence.

After applying sentence segmentation to the English vacancies, 650,698 sentences were extracted. To prepare the labeled training data, we randomly selected 7,000 sentences from this corpus, which constituted roughly 5% of the total number of sentences. These sentences were manually labeled by three independent experts. A coding guide was provided to ensure that labels were consistent (Brannick et al., 2007a) across these experts. Each sentence was labeled as either containing a work activity, a worker attribute, or other. Sentences containing both activity and attribute information were split into parts so that the two are clearly delineated, and buffer sentences that did not contain skill nor task information were put in a new category marked as "other". Example sentences of this type are, *"You will preferably reside locally as you will be expected to participate in call-out cover rota"* and *"Supportive team environment"*. Each expert was tasked with labeling all of the 7,000 sentences. From these sentences, 2790 sentences were labeled as containing attribute information, 3,120 sentences were labeled as containing activity information, and the rest were put into the "other" category. Hence, 5,910 labeled sentences were used as training data. The other 1120 sentences were discarded. As can be noticed, the total number of sentences labeled is 7030, although we originally started with 7000. This is because some sentences contain both attribute and activity information, hence these were split so that the labeled sentences in each category are homogeneous. Later we are going to discuss how the "other" category can be automatically obtained by means of the developed classifiers.

The resulting labels were then compared across experts. We computed *Fleiss' Kappa,* which is an inter-rater reliability measure for instances where there are more than 2 raters. The computed Fleiss's Kappa is .89 (p-value=0.00). Kappa of .89 indicates strong agreement between raters. All disagreements in the labels were resolved by having experts deliberate among themselves which final label to assign.

The sentences in the training data were further preprocessed. All text were converted to lower case and stop words were deleted. We use the standard stop word lists for the English language (Fox, 1989) but we retain the following words: "to", "have", "has", "had", "must", "can", "could", "may", "might", "shall", "should", "will", "would", because these were found useful for the classification task. Sentences containing "to" and "will" often describe tasks, duties, and responsibilities, whereas, "have", "has", "had", "should", and "must" point to knowledge, skills, and abilities (KSAs). We additionally derived context specific stop words by inductively identifying words that do not discriminate between the types of job information we wanted to detect. We then removed all punctuation. Finally, extra white spaces were stripped. To further reduce the number of words, we applied *stemming*. Table 3.1 shows sample output of the text preprocessing on 6 sentences extracted from vacancies.

Table 3.1. Applying preprocessing steps to 6 sentences (with stemming).

	ORIGINAL TEXT	PROCESSED TEXT
D1	Collaborate with the UA Digital Product Team to provide visual design support on new projects.	collabor ua digit product team to provid visual design support project
D2	Ability to lift and mobilize medium to large items up to 75 lbs. while utilizing appropriate equipment and safety techniques.	abil to lift mobil medium larg item to 75 lbs util equip safeti techniqu
D3	Key mandate of CoE is to create mobile assets while generating greater speed and velocity.	key mandat coe to creat mobil asset generat greater speed veloc
D4	3+ years experience in digital design	3 year experi digit design
D5	Must have strong programming and analytical skills and be willing to code full time.	must have strong program analyt skill to code full time
D6	Experience with J2EE technology components (e.g., JSP, Servlets, XML, and Web Services) is a requirement.	experi j2ee technolog compon jsp servlet xml web servic requir

The preprocessed sentences were transformed into vectors by defining several variables. To ensure reliable classification of sentences in either category we used characteristics of both worker attribute and work activity sentences. Hence, we inductively came up with a list of variables that we think would be able to predict the category of sentences in job descriptions. We used ideas from the job analysis field and expert judgement to identify these variables. Based on definitions of tasks, for example, we know that it usually has an action, the object of the action, the source of information or instruction, and the results (Morgeson & Dierdorff, 2011; Voskuijl, 2005). Put this in contrast with worker attribute sentences which is usually a phrase long or even a single word. Thus, we suspected that many activity sentences are likely to be longer than attribute sentences. We therefore added a variable that assessed sentence length. Verbs are particularly prevalent in sentences containing work activities because work activity is typically associated with an action and they also start with a verb (e.g., Collaborate with the UA Digital Product Team to provide visual design support on new projects.). Part-of-speech tagging was used to identify verbs in a sentence and whether the sentence starts with a verb.

We incorporated three general types of features in our study: i) Features based on sentence- and word-level characteristics; ii) features based on syntactic characteristics of words; and iii) features resulting from grammatical patterns. The first two types of features are relatively easy to obtain as they only use the directly observable characteristics of words and sentences.

The first set of features refers to the count of highly frequent words for each job information category. We constructed two-word frequency features using the labeled sentences. The first counts the frequency of words relating to job activity in a sentence and the second the frequency of words relating to job attributes.

Syntactic features leveraged the position and order of words in a sentence. In many sentences in our study of job vacancies, the first word signals the type of job information it communicates. For example, if a sentence starts with the word "develop" or "responsible", then most likely that sentence contains information about a work activity as opposed to a worker attribute. Another feature similarly examines the last word. This feature is most relevant in identifying sentences that express a job attribute, since based on our analysis using the labeled sentences many job attribute sentences end in "skills" or "preferred" or "plus". Apart from POS-derived features, we also included features based on specific words. Our approach was to select those words that best discriminate between the two categories. This was accomplished by inspecting which terms occur frequently for each category. Lastly, we defined two features which determine the proportion of words that are frequent in either activity or attribute sentences. This last set of features seems to rely on word-features; indeed, this is the case but unlike other studies, these words were selected inductively and not priori.

The third type of features involved the application of the part-of-speech (POS) tagging algorithm (Voutilainen, 2003). For the POS labels, we based the tags on the Penn part-of-speech tags (*Penn Part of Speech Tags*, n.d.). This algorithm uses a probabilistic model to look up the POS label of that word in a standard lexical database and the context or the characteristics of the neighboring words. The algorithm was chosen because of its relatively good performance compared to non-probabilistic taggers (Voutilainen, 2003). For the purposes of our analysis, we decided to rely on six **derived POS tags**, namely, **noun, verb, adjective, adverb, TO** (the word to), and **CD** (cardinal number such as 1, 52, 300) by grouping related tags under one general derived tag since this level of information suffices. For example, **NN** (noun, singular or mass), **NNS** (noun, plural), **NNP** (proper noun, singular), and **NNPS** (proper noun, plural) all refer to the noun tag. The same was done for other tags that can be grouped, such as the POS tags that can be combined under **verb, adjective, and adverb.** In the end, the **TO** and **CD** were retained because based on an exploratory analysis they appeared important to discriminating between work activities and worker attributes. The **TO** tag is indicative of job activity (e.g., "to ensure project stays on track for assigned client projects"). The presence of a **CD** tag when followed by the word "year" in a job description may suggest education or work experience required from job applicants, and thus is indicative of the worker attribute category. Other tags that do not belong to these six derived tags were ignored as they would complicate the analysis. In machine learning, adding unnecessary features can make classification performance worse (Guyon & Elisseeff, 2003).

Grammatical pattern variables were extracted from the POS tags of the words, specifically the variable that examines the POS tag of the first word in a sentence. This variable stems from the fact that job activity sentences usually start with **TO** and a verb (both express action) and job attribute sentences often start

with an adjective or **CD.** The **CD** usually starts sentences on duration (e.g., work experience). Likewise, the more verb and **TO** and the less **adjective** and **noun** in a sentence, the more likely it refers to a work activity. Next, five variables were based on our observation that it was useful to examine the **POS** tag of a word that follows a verb. Hence, we obtained five features that count how many **noun, verb, adjective, adverb,** as well as other tags follow a **verb** in a sentence. Another reason why we considered grammatical pattern-based features is that this allowed us to exploit the structure of the language which is more invariable compared to identifying keywords, which based on our experience are perpetually changing. Another group of POS-derived features are the proportion of nouns, adjectives, verbs, etc. in each sentence. We also incorporated features that capture word order information that are akin to bigrams but instead of words we considered the POS tags. This was done because experts noted that even if a sentence has a high proportion of adjectives (indicative an of attribute), if many adjectives are followed by a verb, then it signals that the sentence is likely to pertain to work activity information.

To assess the predictive strength of each feature, we determined the proportion of sentences which only that feature could classify. As expected, no single feature provided a perfect separation of sentences into the two job information categories. However, using analytical methods, features can be combined in a way that yields a higher predictive power.

A total of 168 features were identified. Using these 168 features we constructed vectors that represent each sentence (our unit of analysis). The complete list of variables can be found in Table 3.2.

Table 3.2. The 168 Variables for the Vacancy Information Extraction Task

Feature Type	Number of derived features	Variable Type	Variable code
Part of speech (POS) tag of the first word	1	Categorical (actual POS)	V1
Is the first word in this sentence unique in work activity sentences (based on the labeled training data)	1	Numeric	V2
Is the first word in this sentence unique in worker attribute sentences (based on the labeled training data)	1	Numeric	V3
Is the last word in this sentence unique in work activity sentences (based on the labeled training data)	1	Numeric	V4
Is the last in this sentence unique in worker attribute sentences (based on the labeled training data)	1	Numeric	V5

Proportion of adjectives	1	Numeric	V6
Proportion of verbs	1	Numeric	V7
Proportion of word "to"	1	Numeric	V8
Proportion of modal verbs	1	Numeric	V9
Proportion of numbers	1	Numeric	V10
Proportion of adverbs	1	Numeric	V11
Proportion of nouns	1	Numeric	V12
Proportion of nouns, verbs, adjectives, adverbs, and other part of speech tags followed by another verb	5		V13-V17
Proportion of unique words found only in work activity sentences (based on the labeled data)	1	Numeric	V18
Proportion of unique words found only in worker attributes sentences (based on the labeled data)	1	Numeric	V19
Frequency of keywords for work activity and worker attributes sentences	149	Numeric	V20-V168

More formally, we consider a corpus of job vacancies which we denote by $D = \{1,\ldots, N\}$. Each vacancy in the corpus has a number of constituent sentences, which we denote by s_{ij} , that is, the *i*"th " sentence in vacancy *j*. Since our unit of analysis is the sentence, and therefore it serves as our input, we transform and represent each sentence into a vector $s_{ij} = [w_{ij}^1 \ \ldots \ w_{ij}^K]$, where the w's are the features we define or construct for the sentences. For each sentence vector we associate a label $t_{ij} \in \{0,1\}$, where 0 refers to a sentence containing work activity and 1 if the sentence contains a worker attribute. Note that as was mentioned before it is possible that a sentence contains neither a work activity nor a worker attribute, we did not assign a specific category for this type of sentence instead we inspected the prediction of the classifiers and we set a threshold. If a classifier predicts a value between 0.45-0.55, we put the sentence into the "other" category since it implies that the classifier has difficulty in determining how to classify sentence. Thus, it had been necessary that classifiers considered should give predictions between 0 and 1 rather than just output the specific category.

Application of Classification Techniques

The data matrix served as the input data for the classification of job information. For the construction of the classification model, we added the169th column to the data matrix. This column contained the classification of sentences into either job attribute (0) or job activity (1) as obtained from the manually labeled sentences.

We built each classifier and assessed its performance through 10-fold cross-validation using AUC as performance metric. This performance metric reflects our

objective of creating an accurate classifier that favors either one of the categories (attribute or activity). Unlike the other 5 classifier which used the derived features we described previously, BERT uses sequences of words as input, produces an embedding for the sequence of words, after which the embeddings are fed to the classifier, in our case a logistic regression model (we will return to this later).

We also experimented on the different values for the hyperparameters for each technique. Below, we present the results of our experiments for each technique in the succeeding sections.

Training, testing, and validation. Before we ran the experiments, we first split the data into a training, a testing, and a validation set. The split was 80% train, 10% validation and 10% test. For each run, we further performed 10-fold cross validation using the training set to tune parameters. Once the "best" hyperparameters were found we ran the model on the training data and used the validation set to assess the performance. Hence all reported performances here refer to the performance on the validation set, unless otherwise specified. The test set was not used in model training and was solely reserved to compare performances among the different classifiers.

Penalized logistic regression. There are two hyperparameters for penalized logistic regression, namely, *alpha* and *lambda*. The alpha hyperparameter determines the type of penalization used. An alpha of 1 means lasso and 0 means ridge. An alpha between 0 and 1 results in a form of penalization called *elastic net* and balances lasso and ridge. The lambda on the other hand is the parameter that controls the overall strength of the penalty. Lambda can be set to any nonnegative number.

We simultaneously performed a hyperparameter search on alpha from 0 to 1 (with.0001 increments) and lambda 0-0.1 (with .000001 increments). The lambda range was determined by first investigating how AUC changed as we varied the lambda from 0 to 1 which was done to reduce the search space. Figure 3.2 shows the AUCs using various lambdas. The optimal lambda was found to be .004048273.

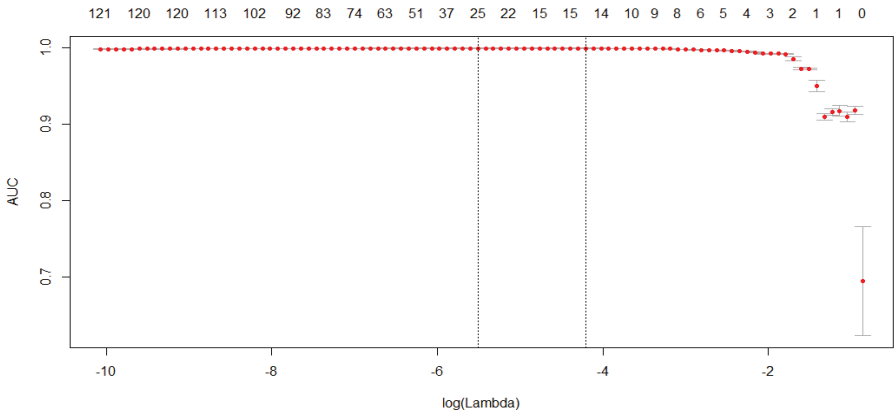


Figure 3.2. Lambdas and the corresponding AUCs. The x-axis shows the logarithm of lambdas instead of the actual lambda values.

For searching the best hyperparameter combination, we did not test all possible combinations of alpha and lambda, but instead performed a random discrete search on the grid. The best hyperparameter combination found was alpha= .768 and lambda =.028196 with an AUC of .9988988. The ROC Curve is shown in Figure 3.3(a).

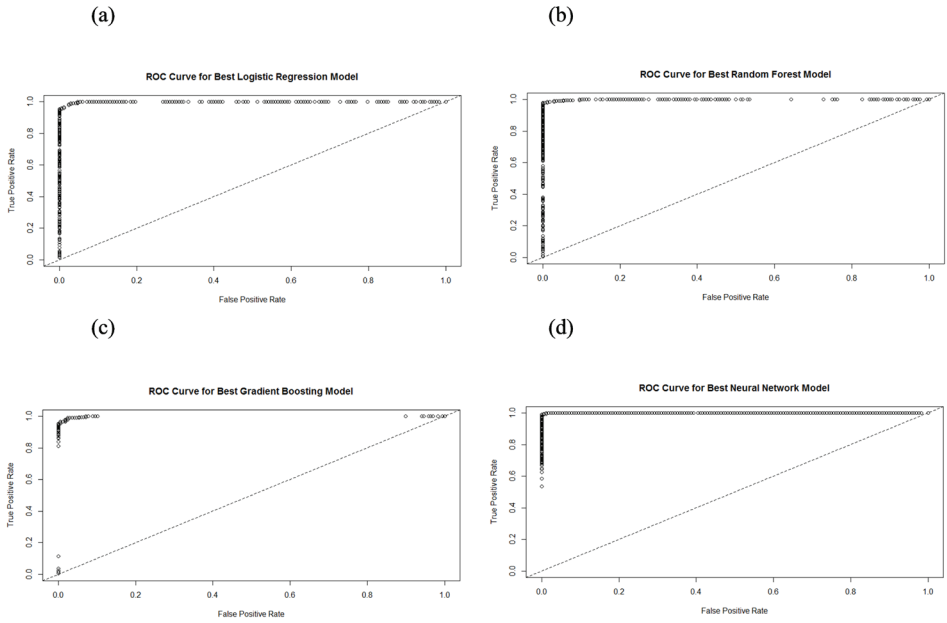


Figure 3.3. ROC curves for the best (a) logistic regression model, (b) random forest model, (c) Gradient Boosting Model, and (d) Neural Network Model

Aside from the AUC it is also of interest to examine variable importance, that is, which variables were found to be important in the model. This can be obtained in Penalized Regression Models, in this case the top 13 important variables are summarized in Table 3.3(a). The last column indicates which category the variable is good at predicting. For ACT, it is highly predictive of activity and for ATTR it is highly predictive of attribute. For example, the variables "lead" and "support" are indicative of work activity and the variables "ability", "skills", and "experience" signal that a sentence contained worker attribute information.

Table 3.3. Variable Importance from (a) Logistic Regression and (b) Random Forest Model (Top 13 features).

(a)

Rank	Names	Feature Type	coefficients	Predict
1	V2	Is the first word in this sentence unique in work activity sentences (based on the labeled training data)	1.36	ACT
2	V3	Is the first word in this sentence unique in worker attribute sentences (based on the labeled training data)	0.84	ATTR
3	V18	Proportion of unique words found only in work activity sentences (based on the labeled data)	0.71	ACT
4	V19	Proportion of unique words found only in worker attributes sentences (based on the labeled data)	0.48	ATTR
5	V4	Is the last word in this sentence unique in work activity sentences (based on the labeled training data)	0.41	ACT
6	V5	Is the last in this sentence unique in worker attribute sentences (based on the labeled training data)	0.34	ATTR
7	ability		0.25	ATTR
8	skills		0.20	ATTR
9	experience		0.07	ATTR
10	lead		0.06	ACT
11	strong		0.06	ATTR
12	excellent		0.03	ATTR
13	support		0.01	ACT

(b)

Rank	Variable	Relative importance	Scaled importance	Percentage (%)
1	V2	961.32	1.00	24
2	V18	756.66	0.79	19
3	V3	713.32	0.74	17
4	V19	429.71	0.45	11
5	V4	343.37	0.36	8
6	V7	159.63	0.17	4
7	V5	125.78	0.13	3
8	V1	115.05	0.12	3
9	skills	112.56	0.12	3
10	V8	83.61	0.09	2
11	ability	65.27	0.07	2
12	years	34.91	0.04	1
13	experience	33.76	0.04	1

Random Forest. As was done in the penalized logistic regression, we experimented on the different hyperparameters to find the best Random Forest model. Specifically, we varied the following: (1) max depth of each tree from 5 to 15; (2) the min number of observations per leaf (5,10, 20, 50, 100); (3) the number of bins for continuous variables (2, 3, 5,10); (4) the number of bins for the categorical variables (3, 5); (5) the row sample rate for each tree (0.7,1); and the number of variables to sample (8, 13, 16, 32,64).

The best combination was determined by AUC and the search strategy was discrete search. These experiments resulted in the optimized parameters as presented in Table 3.5(a).

The calculated AUC is .9990476. The ROC for the best Random Forest is shown in Figure 3.3 (b). The top 13 most important variables for the Random Forest are shown in Table 3.3(b).

Table 3.4. Variable Importance from (a) Gradient Boosting Machine Model and (b) Neural Network model.

(a)

Variable	Relative importance	Scaled importance	Percentage
V2	2048.575195	1.000000	0.508523
V18	983.166870	0.479927	0.244054
V3	514.169250	0.250989	0.127634
V4	199.151352	0.097215	0.049436
V19	93.082680	0.045438	0.023106
ability	65.761879	0.032101	0.016324
V5	48.220634	0.023539	0.011970
skills	18.311430	0.008939	0.004545
experience	8.022392	0.003916	0.001991
lead	7.940779	0.003876	0.001971
V1	7.236656	0.003533	0.001796
analysis	5.136837	0.002508	0.001275
V7	4.770038	0.002328	0.001184

(b)

Variable	Relative importance	Scaled importance	Percentage
solving	1.000000	1.00000	0.046386
V5	0.800713	0.80071	0.037142
software	0.724904	0.72490	0.033626
analytical	0.659675	0.65967	0.030600
V6	0.643668	0.64366	0.029857
responsible	0.503691	0.50369	0.023364
V1.CC	0.448885	0.44888	0.020822
support	0.388902	0.38890	0.018040
analysis	0.378964	0.37896	0.017579
V1.UH	0.376777	0.37677	0.017477
making	0.341317	0.34131	0.015832
V1.CD	0.321547	0.32154	0.014915
previous	0.316562	0.31656	0.014684

Gradient Boosting Machine. It is no surprise that some hyperparameters in random forest are also present in GBM. Hence, we tune the same hyperparameters except for the *number of variables to try.* The other hyperparameters that we tuned are the learning rate, learning rate annealing which scales the learning rate after each tree (by default no scaling is done), and column sample rate. We used the same values as in Random Forest and for the learning rate we tried (.001, .01, .1) and for the learning rate annealing we tried the following values (.99,.999, 1). Additionally, we experimented on the column sample rate (0.7, 1).

The best combination for the values of the hyperparameters yield the maximum AUC is shown in Table 3.5(b). The AUC for the best GBM model is .9988095. The ROC for the best GBM model is shown Figure 3.3(c). GBM also provides variable importance to help us decide which among the features are crucial in the classification (see Table 3.4(a)).

Table 3.5. Best parameters found for the (a) Random Forest model, (b) Gradient Boosting Machine model, and (c) Summary of the tested values of the parameters of NN and the optimal values found.

(a)

Hyperparameter	Optimum
Max depth	10
Minimum number of observations per leaf	20
Number of bins for continuous variables	5
Number of bins for categorical variables	3
Row sample rate for each tree	1
Number of variables to try	13

(b)

Hyperparameter	Optimum
Max depth	11
Minimum number of observations per leaf	5
Number of bins for continuous variables	5
Number of bins for categorical variables	3
Row sample rate for each tree	0.7
Learning rate	0.001
Learning rate annealing	0.99
Column sample rate	0.7

(c)

Type	Parameter	Experimented values	Optimal
Architecture	activation	Rectifier, Maxout, Tanh, Rectifier with drop out, maxout with drop out,and tanh with dropout	Maxout
Architecture	Hidden layer and nodes per hidden layer	(8,8,8,8)*, (64,64), (32,32,32), (128)	(64, 64)
Learning	epochs	(10,50,100)	50
Learning	L1	(0.001,0.0001)	0.001
Learning	L2	(0.001,0.0001)	0.001
Learning	adaptive rate	YES, NO	YES

Learning	rate	(0.1, 0.01,0.001)	0.001
Learning	rate annealing	(1e-8,1e-7,1e-6)	1e-07
Learning	rho	(0.9,0.95,0.99,0.999)	0.99
Learning	epsilon	(1e-10,1e-8,1e-6,1e-4)	1e-10
Learning	momentum start	(0,0.5)	0.5
Learning	momentum stable	(0.99,0.5,0)	0
Learning	input dropout ratio	(0,0.1,0.2)	0

Note. * This means that there are 4 hidden layers with each hidden layer consisting of 8 nodes each.

Neural network. We experimented on the different parameters to find the best neural network model and the search strategy that we used is discrete random search. Table 3.5(c) displays the different values we tried during the experiment and the optimal values found after training. In summary, the best architecture found has 2 hidden layers with 64 nodes each with the Maxout activation function. For the training algorithm, the best learning rate is .001, L1 and L2 of .001, rho of .99, and epsilon of 1e-10. The best neural network model has an AUC of .9999107. We can also examine variable importance in our neural network model, which is provided in Table 3.5(b).

Unlike Random Forest, GBM and Penalized Logistic Regression, the Neural Network seemed to favor word-features over sentence-based features. The word-based features are the ones we obtained inductively from the training data.

Stacking. The four classification models developed so far already exhibited good performance (AUC's of at least 90%). Next, we investigated whether we could further improve the performance by combining the four classifiers. As was noted above, NN gave a slightly different set of important features compared to the other classifiers. Our hypothesis was that by combining these classifiers we may further reduce the variability in our predictions. We combined the classifiers using a technique called stacking which is an ensemble technique that permits the combination of different base learners. Although majority voting could also have been applied, stacking has been shown to exhibit superior performance over the former (Zenko et al., 2001). For the base learners we used the preceding optimized hyperparameters and trained each classifier using 10-fold cross validation. We then used GLM to stack the four models. The coefficient of the stacked ensemble is shown in Table 3.6.

Table 3.6. Coefficient of the stack ensemble using on GLM as the stacking algorithm.

Names	Coefficients
Intercept	-7.264738
GLM_model_R_1587959348049_1348	6.231609
DRF_model_R_1587959348049_1381	5.113009
GBM_model_R_1587959348049_1473	0.000000
DeepLearning_model_R_1587959348049_1728	2.103388

BERT-Logistic regression classifier. Using the same train-test split we ran another classification model using BERT. The classification process is depicted in Figure 3.4(a). The architecture has 12-encoder layers, 768 hidden layers, 12 attention heads and in total there are 110M parameters. The architecture is trained on cased English text and is provided by Hugging Face[1].

We also experimented on the optimal lambda parameter of the logistic regression classifier which was found to be 0.19. The mean cross validated AUC was .9860. The AUC for the best classifier is shown in Figure 3.5.

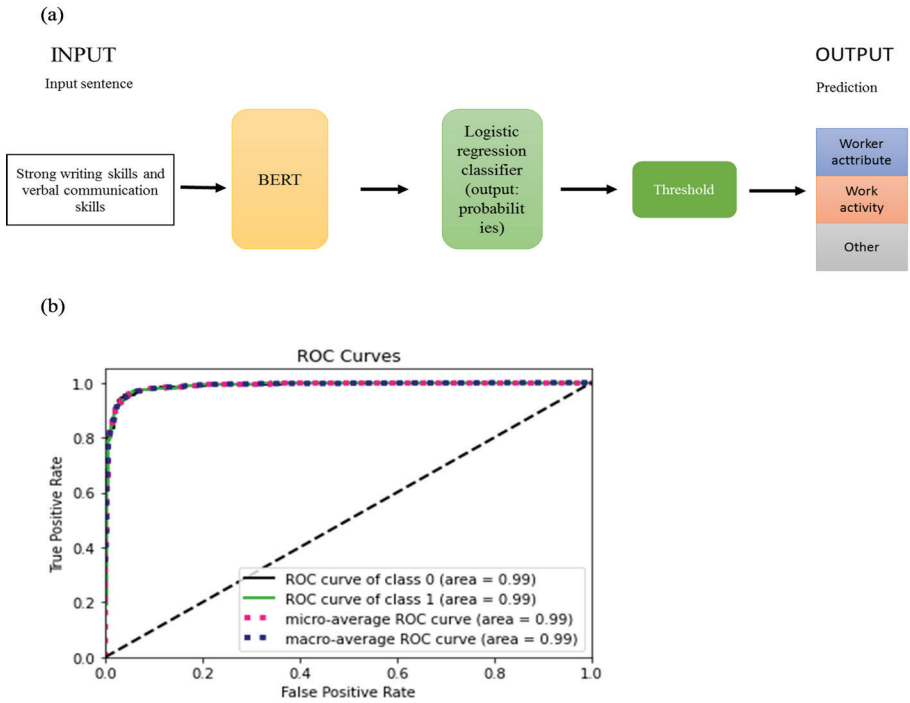


Figure 3.4. (a) The BERT Architecture and (b) its performance on each category

[1] https://huggingface.co/

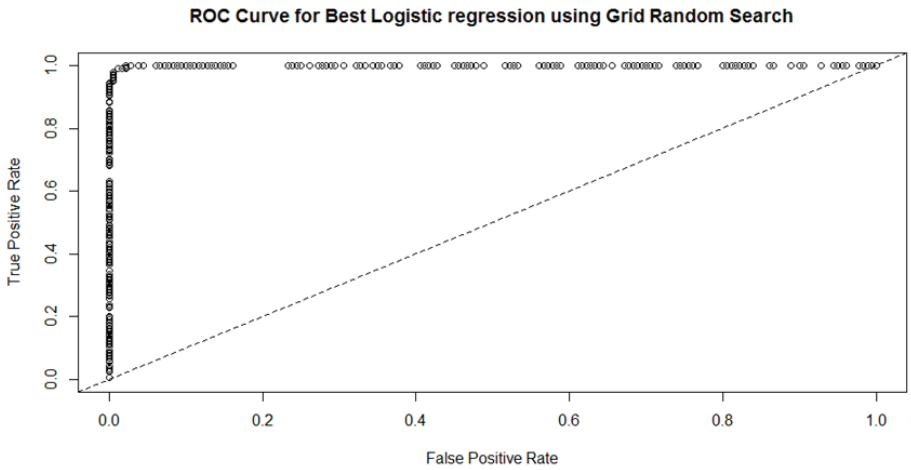


Figure 3.5. AUC of the Best Logistic regression based on the test set.

Comparison of performance. Here we compare the performance of the 6 models using the validation and test set to find the best model.

Based on the performance on the test set the best model is the Penalized Logistic Regression. The ROC is shown in Table 3.7. Although, the BERT-Logistic seemed to perform poorly compared to the other classifiers, we do not discount the use of this classifier since we can improve the performance by training the BERT architecture on additional vacancy data, or by fine tuning certain parameters. What we did here is use an off-the-shelf pretrained BERT model.

A comparison of the cross validated AUCs is shown in Figure 3.6. We tested if there was indeed a significant difference between the classifiers using Kruskall-Wallis test (a nonparametric test).

Table 3.7. Comparison of the models using AUC on validation and test sets.

Model	AUC on validation	AUC on test
Penalized Logistic Regression	0.9988988	0.999518
Random Forest	0.9964732	0.9991121
Gradient Boosting Machine	0.9988690	0.997032
Neural Network	0.9997024	0.9993403
Stacked Ensemble	0.9988690	0.9994419
BERT-Logistic	0.9860000	0.9531000

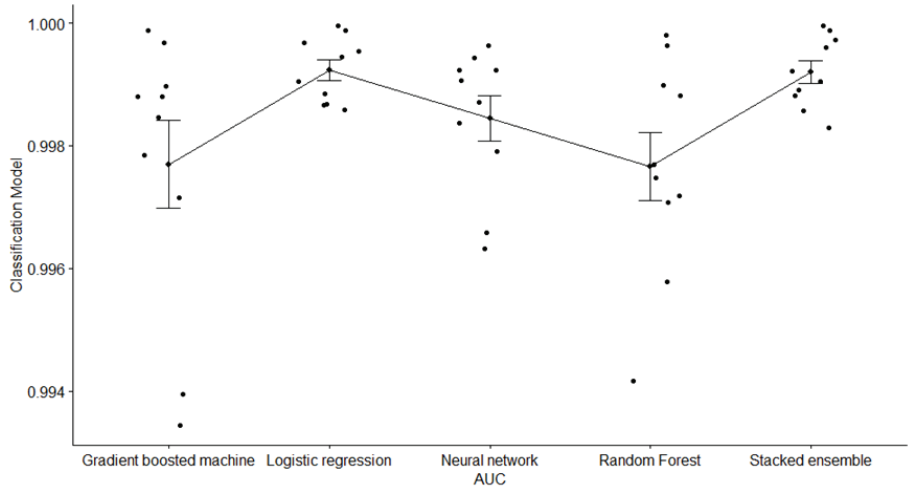


Figure 3.6. Comparison of the AUCs (with the AUC from each fold) of the five models.

The result of the test is summarized below:

Kruskal-Wallis rank sum test

data: AUC by classifier

Kruskal-Wallis chi-squared = 8.6519, df = 4, *p*-value = .07041

The *p*-value is greater than .05, hence we cannot conclude that there is a significant difference between the classifiers. Although mean accuracies are the same, in practice, we would prefer a classifier with low variance since it yields more stable predictions, in this case, we would either opt for penalized logistic regression (SD=.000529) or stacking (SD=0.000577). Furthermore, in job analysis applications it may be preferable to use models that lend themselves to easy interpretation, which is not the case for deep learning models that tends to focus more on performance.

The constructed classifiers in conjunction with the variables can classify sentences from vacancies into activities or attributes with good classification performance addressing RQ1. In the succeeding sections, we are going to show how these classifiers could be used to help address questions pertaining to job analysis. The first application is in analyzing skill demand and the second is in clustering jobs. A small app for the classifier can be found in https://vkobayashi.shinyapps.io/labelme/

In defining and building the classifiers, we found that it is possible to automate the extraction of job information from vacancies with a respectable performance. Also, both grammatical and syntactic features from the sentences are important for the classification task.

Having trained machine learning algorithms that classify sentences into worker attributes and work activities, in the next section we attempt to address

RQ2, that is, can the extracted work activities be used to construct task groups and meaningfully cluster jobs based on tasks? Subsequently, we attempt to tackle RQ3, that is, can the task groups be validated by comparing it to tasks as enumerated in the ESCO[2]? To address RQ3, we map the extracted work activities to the tasks (as identified by ESCO) of a specific job and measure the amount of overlap as a validation strategy.

Constructing Job Task Groups from Vacancies

Existing methods for studying work-oriented activities are time and motion study, Functional Job Analysis (FJA), task inventories and the Critical Incident Technique (Morgeson, 2017). FJA is the most popular method and has stood at the basis of the development of job taxonomies such as the O*NET (Campion et al., 1999; Jeanneret & Strong, 2003). FJA is conducted by job experts who manually collect job information from various sources, observe workers, and interview subject matter experts (SMEs). Hence, performing FJA can be time consuming and laborious. Our classification models may help address these challenges by automating some of the steps involved in task analysis. Moreover, this work is an initial effort to demonstrate the potential of the developed techniques to extract job information from vacancies which may be used as a starting point to other methods.

The analysis here is also guided by inference models in job analysis that were advanced by job analysis researchers. Specifically, we adopted the inference based model of Morgeson & Campion (2000), which states that there are three critical inferences in job analysis, namely, job descriptive inference, job specification inference, and operational inference. In the analyses that follow, we focused on job descriptive inference, which pertains to work activities and how these activities (i.e., tasks and duties) underlie job performance. Hence here we set out to find the types of activities that can be extracted from vacancies and by doing so will establish the utility of vacancies as valuable sources of job information for job analysis.

Another aspect that we want to explore is to extract activities that are timely and specific to the context of the current labor market. The contextualized information can be used for selection systems, employee training programs, and performance management.

Applying the classifiers

We ran the classifiers on over 650,696 sentences and obtained an additional 271,737 work activity sentences and 317,451 worker attribute sentences. Unlike existing approaches which analyze whole job descriptions, our classifiers have largely filtered out those portions of the text which were not relevant to work activities hence reducing errors and bias.

[2] https://esco.ec.europa.eu/en

Topic Modeling on Work Activities

The below analyses pertain to the extracted work activities. The goal was to summarize the work activities and to identify specific duties and tasks.

For this purpose, we applied a topic modeling technique called Latent Dirichlet Allocation (LDA) model (Blei et al., 2003) on the extracted work activities. Gibbs sampling was used to estimate the parameters of the LDA model (Porteous et al., 2008). The number of topics had been chosen based on four criteria. One criterion is based on topic distances as discussed in the paper of Cao et al. (2009) and the other is based on the idea that LDA is a matrix factorization mechanism and the quality of the factorization depends on choosing the right number of topics (see Arun et al., 2010 for more detail). Based on the results the number of topics was set to 200 (see Figure 3.7)

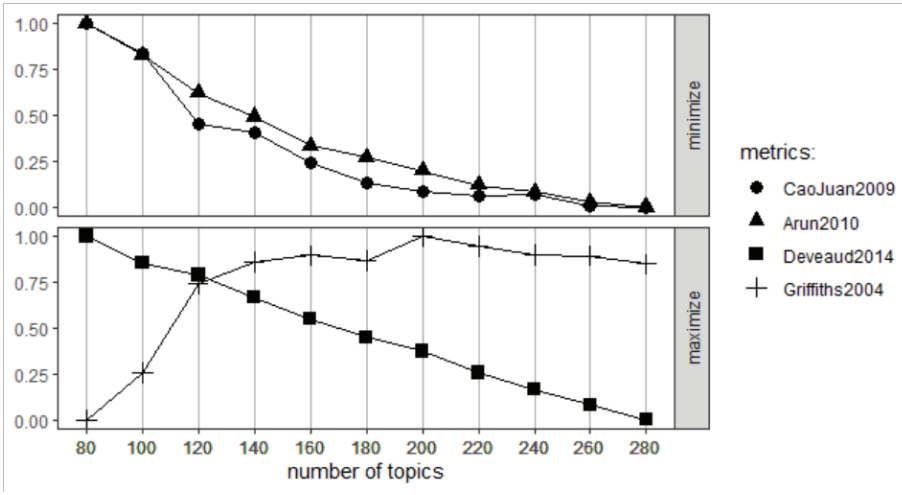


Figure 3.7. Determining the Number of Topics.

A complete set of extracted topics is provided as supplementary material. A sample of topics is presented in Table 3.8.

Examining the top 8 words, Topics 7 and 18 are related to collecting, analyzing, and reporting data and information. Topic 6 relates to the tasks involved in recruitment which are sourcing and interviewing candidates. Topic 15 refers to providing technical support. Topic 89 is related to developing and maintaining good relationships, Topics 20 and 38 seem to be about leading and managing teams, and Topic 112 is about coaching and mentoring. Topics 48 and 116 involve mundane tasks of attending and participating in meetings and performing follow-ups. It is quite surprising to find these activities mentioned in vacancies, because they may actually deter candidates from applying to the job in question. Topic 109 seems to be about tasks about equipment maintenance and Topic 180 is about digital/online

marketing campaigns. The rest of the Topics seem to be about technical skills specific to certain professions such as in the hospitality industry for Topic 160, accounting and finance related jobs for Topic 70 and 97 and health care for Topic 28. Topic 199 pertains to a specific task which is project development and evaluation. Hence some of the extracted topics correspond to actual tasks and duties. Interestingly, even without giving LDA prior information about which work activities to extract, it still was able to recover technical tasks.

Table 3.8. Sample of topics from applying LDA on work activities

Topic 6	**Topic 7**	**Topic 15**	**Topic 18**
candidate	report	technical	report
interview	analyze	provide	monthly
hire	information	support	weekly
recruit	prepare	engineer	daily
recruitment	datum	expertise	basis
job	collect	solution	quarterly
source	gather	documentation	update
manager	trend	assistance	review
Topic 20	**Topic 28**	**Topic 38**	**Topic 48**
leadership	patient	lead	meeting
provide	care	team	participate
development	medical	effort	attend
direction	health	development	review
team	clinical	develop	conduct
staff	hospital	generate	session
think	provide	manage	present
guidance	staff	generation	actively
Topic 70	**Topic 89**	**Topic 97**	**Topic 109**
tax	develop	financial	equipment
account	maintain	budget	maintenance

prepare	effective	forecast	repair
financial	good	analysis	perform
preparation	relationship	report	facility
statement	time	monthly	inspection
report	set	cost	installation
monthly	process	finance	safety
Topic 112	**Topic 116**	**Topic 160**	**Topic 180**
team	call	food	campaign
member	phone	ensure	market
mentor	email	hotel	digital
coach	via	standard	online
junior	answer	service	email
act	contact	guest	advertise
leader	follow	maintain	search
feedback	inquiry	follow	optimization

The distribution of terms associated to a topic give us an idea about the interpretation of that topic. The interpretation can also be enhanced by examining term correlations which can be visualized using a word-network (see Figure 3.8). Here, a word network is constructed wherein a link or edge between two terms signifies a correlation of at least 0.1. The strength of the correlation is highlighted by adjusting the width of the edge. Correlation is computed from the document-by-term matrix, which is a matrix containing the frequency of words in each document, each row corresponds to a document (in our case job vacancy) and each column corresponds to a term (i.e., terms/words found in the job vacancies). Then, to compute the correlation between two terms we extract the two columns corresponding to the two terms (i.e., two vectors) and calculate the spearman-rank correlation between the two vectors. The word-networks are in line with our interpretations and show that a topic could capture more than 2 work activities; the model puts them in one topic because they tend to co-occur. From the topics we can generate hypotheses about which work activities underlie job performance, which could then be validated by interviewing subject matter experts or by observing job incumbents.

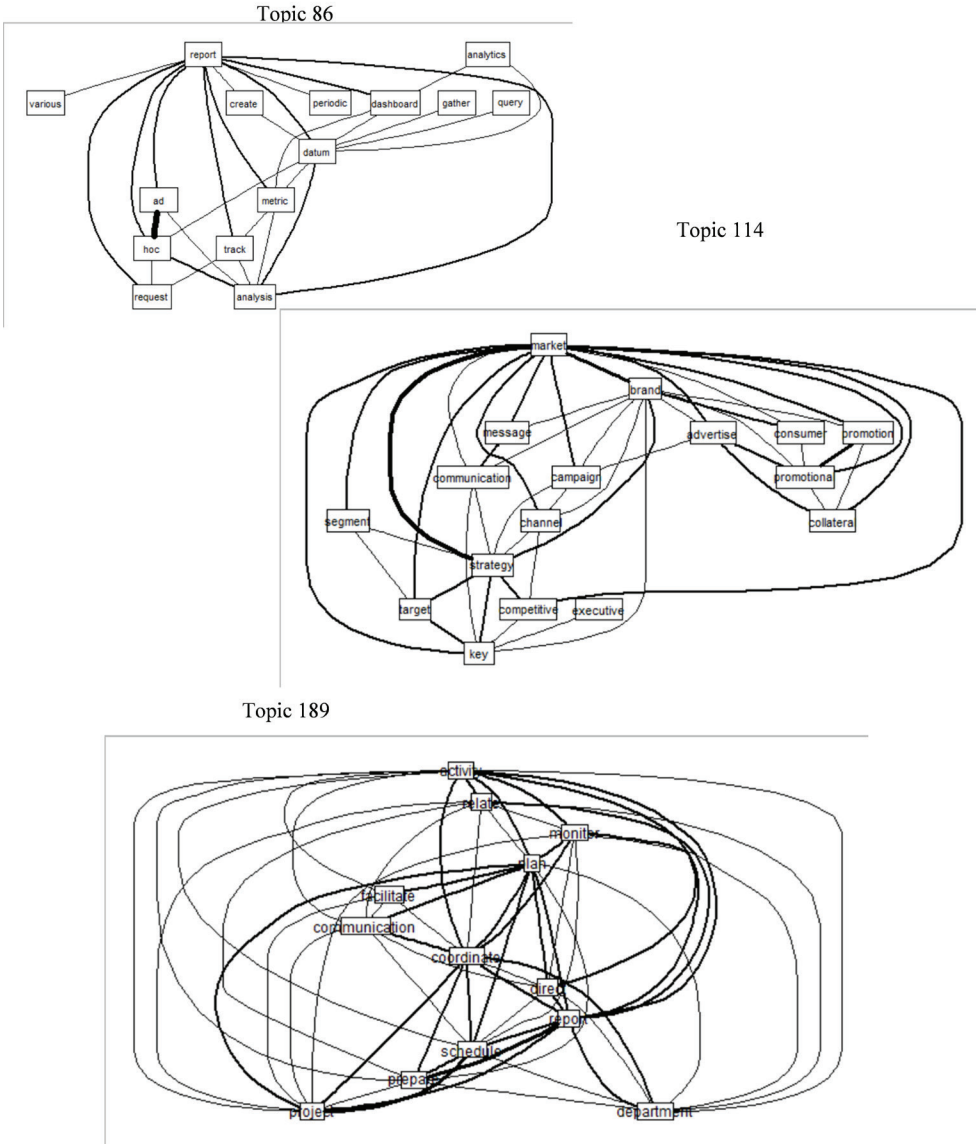


Figure 3.8. Correlation network for the terms in Topic 86, 105, and Topic 15

Further examining the relationships among the topics can be used to assess the convergent/discriminant validity of the topical content. Since LDA constructs uncorrelated topics, directly using correlation would be inappropriate, but there are other ways to measure relationships such as computing a similarity or distance measure between topics. A suitable distance measure for this application is the Jensen-Shannon divergence since it measures the distance between probability distributions (note that the topics are probability distributions on words). To focus our discussion, we use Topic 2 as an example. This topic can be interpreted as referring to the task of data analysis and reporting. Topic 2 is closer to Topics 7, 18, 97, and 117 (please refer to the supplementary material for the complete list of topics). Topics 7 (collecting, analyzing and reporting), 18 (periodic reporting), 97 (financial reporting and forecasting), and 117 (progress monitoring and reporting) all relate to the task of reporting and analyzing data and information. A noteworthy similarity exists between Topics 20, 29, 38, 102 and 193 which pertain to leading a team, providing support and guidance, leading team effort and development, developing leadership talent, and leading and contributing to project plans and activities, respectively. We can further explore these relationships by performing a more inference driven investigation by comparing the findings here to the results obtained by interviewing SMEs or job incumbents, which will further help in establishing construct validity. Aside from similar topics, there are also less similar ones (based on the calculated measure). For example, Topic 46 (making interactive customer experience) is least similar to Topics 70 (accounting and preparing financial statements) and 2 (data analysis and reporting). Possible interpretations include range restriction, that is, if job incumbents in a position do not vary on certain characteristics these characteristics may not be mentioned in the vacancies. But it could also mean that interacting with customers is not an essential task for job holders who perform Topic 2.

To examine the relationship among all topics simultaneously, we applied multidimensional scaling and noted the following observations: Topics 46, 19, 31, and 4 are close together because they all relate to making interactive customer experience. This also holds for Topics 70, 135, 145, 174, and 181, which are about complying with regulations. Topics 20, 43, 11, 102, and 171 are about how to lead a team, supporting team members, and training for leadership.

Validation using ESCO Classification

Next, we set out to validate the extracted activities to the ESCO classification (Nomden, 2012). ESCO categorizes occupations into different groups and hierarchies also called the occupational pillar. The classification is based on ISCO-08 (Ganzeboom, 2010; ILO, 2012) code and therefore each occupation in ESCO is mapped to this coding system. The classification has 10 major job groups

and each major group is further divided into sub-major, minor and unit groups (4 levels). For example, the Managers major group is divided into 4 subgroups, namely, Chief executives, senior officials, and legislators (11), Administrative and commercial managers (12), Production and specialized services managers (13), and Hospitality retail and other services managers (14). Then each of the subgroups are divided further into minor groups and finally to unit groups. Finally, ESCO provides an occupational profile which lists the tasks commonly performed in that occupation. For more information we refer the reader to this website: https://esco.ec.europa.eu/en/classification/occupation_main.

For this validation, we use the Software Developer as this occupation is a highly technical one and usually tasks for this role are enumerated in vacancies. The occupation has the following tasks as enumerated in ESCO:

(a) researching, analyzing, and evaluating requirements for software applications and operating systems;
(b) researching, designing and developing computer software systems;
(c) consulting with engineering staff to evaluate interfaces between hardware and software;
(d) developing and directing software testing and validation procedures;
(e) modifying existing software to correct errors, to adapt it to new hardware or to upgrade interfaces and improve performance;
(f) directing software programming and development of documentation;
(g) assessing, developing, upgrading and documenting maintenance procedures for operating systems, communications environments and applications software;
(h) consulting with customers concerning maintenance of software systems.

We manually mapped each of the topics from topic modeling to the preceding tasks and computed the simple matching coefficient or Rand similarity coefficient given by the formula

$$SMC = \frac{\text{number of matched tasks to topics}}{\text{number of tasks}}$$

The number of topics is 200 and the number of tasks is 8. A topic may be mapped to two or more subgroups. For example, Topic 13 which has the following top 20 terms (see Table 3.9) was mapped to a, c, d, g, and h tasks. All 8 tasks have been matched to at least one of the topics. The computed SMC is 1 implying that all the enumerated tasks have been recovered from the vacancies, hence providing support to the validity of the extracted work activities from vacancies.

Table 3.9. Top 20 terms for Topic 13

Topic 13
requirement
business
document
functional
specification
gather
process
technical
user
project
documentation
design
solution
define
understand
translate
analysis
detail
implementation
facilitate

Job Clustering

Job clustering is an important step for designing selection and promotion systems as well as in wage and salary administration (Mainert et al., 2019; Schlee & Karns, 2017). That is, similar jobs may share the use of the same selection system or may be put in the same salary range. Recent applications of job clustering include a job board recommendation system (Benabderrahmane et al., 2018) and the study of cross-occupational skill transferability (Snell et al., 2016).

The output from LDA allows us to determine the most likely topic for each document. We further made use of the extracted job information by defining "job similarity" based on topic mixtures, and then clustering the jobs. Here, the specific task is to find work activities for each job in terms of the topic distribution and to apply clustering to group similar jobs according to work activities. Consider Topics 20 and Topic 18. Most jobs under Topic 20 are managerial in nature jobs such as Director of Sales Engineering, Manager, Account Director, and Vice President. On the other hand, jobs under Topic 114 appear to pertain mostly to jobs involving record keeping and maintenance (e.g., Accreditation, Regulation and Licensing and Senior Associate, Internal Audit Job). Note that in LDA, each document can have more than one topic (each document is a mixture of topics), we can utilize all topic probabilities for each document and use that as input to cluster jobs.

To measure the similarity of each job we computed the Jensen Shannon divergence. Clusters ware then formed using a hierarchical clustering method called Ward's method (Berry & Castellanos, 2008; Mobley & Ramsay, 1973). Hierarchical clustering starts with individual job vacancy as its own cluster, then it sequentially merges vacancies until all vacancies are in one big cluster. The merging of clusters is represented in a dendrogram. The number of clusters to retain is determined either by specifying a threshold as to where to cut the dendrogram or setting the number of clusters to extract. In our case, we set the number of clusters to 100. For illustrative purposes, we show two job clusters in Figure 3.9: one cluster consists of managerial jobs in HR and the other cluster consists of support related IT and analyst jobs.

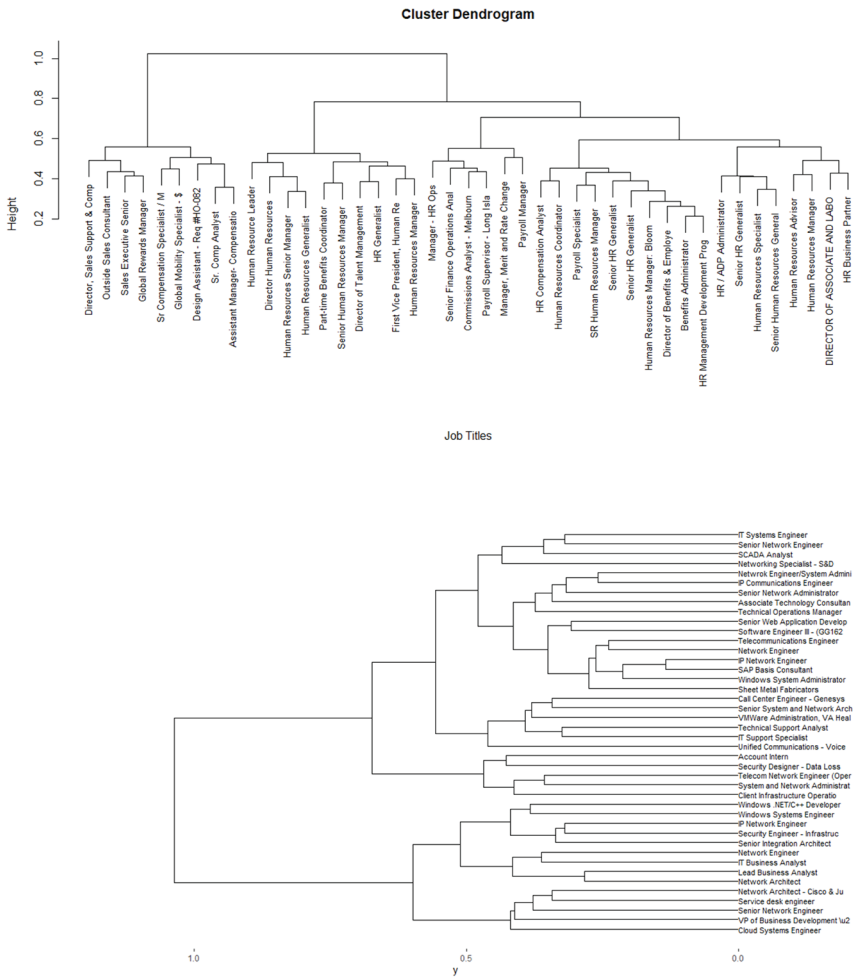


Figure 3.9. Two clusters of jobs from applying hierarchical clustering of jobs from topic distribution.

The natural next question is why do these jobs cluster together? What work activities are shared among these jobs in the two job clusters? For the first presented cluster (HR related), work activities common to these jobs are "Providing timely advice on HR policies and procedures", "Maintaining an updated recruiting pipeline", "oversee the development implementation of human resources policies programs and services", "Communicates with all employees regarding policies and procedures", and "Facilitate recruiting and new hire orientation activities". For the second presented cluster, the work activities common for this job cluster are related to support such as the use of Technical support, Service desk engineering as well as Associate technology consultant. For the specific work activities, job holders are expected to "Provide clients with full range of technical support services for their networks workstations remote computing communications and Internet needs", "Assist in technical implementation of online content for websites", and "… provide first and second-line computer support towards end users".

From this clustering of jobs our key finding is that is indeed possible to obtain job clusters using work activities extracted from vacancies. Clustering on the basis of work activities provides useful information for personnel selection and placement (Lopez et al., 1981). Through job clustering we can find which jobs are most similar in terms of work activities. Based on the clustering here, the "HR manager" vacancies tend to emphasize leadership skills and knowledge HR practices whereas the "IT and analysts support" vacancies put more weight on technical experience and customer support skills. The results here can be used as output to study how a certain task or duty is valued in the labor market, by relating it to the salary listed in the vacancies.

Summary and Conclusion

In the preceding analyses we addressed our research questions. For the first research question: What features are useful to identify work activities and worker attributes, and can these features be combined to build an automatic and accurate classifier that can distinguish between work activities and worker attributes contained in job vacancies? We managed to identify and build classifiers that extract two types of job information from vacancies, namely, work activity and worker attribute. The best models found were Penalized Logistic Regression and Stacking. In subsequent analyses, Stacking was used since it combines several methods, which often leads to better performance compared to just using one technique. Regarding the variables, the models revealed that the best features are those that consider the lexical content and structural form of the sentence. For the second research question: Can the extracted work activities be used to construct task groups and meaningfully cluster jobs based on tasks? We demonstrated that work activities extracted from vacancies can be further aggregated into groups that revealed task groups common to some jobs

(such as engage with customers for customer service-oriented jobs). The task groups were then used to represent each job, after which a clustering of jobs was performed which revealed job similarity based on the work activities. For the third research question: Can the task groups be validated by comparing it to tasks as enumerated in the European Skills, Competences, and Occupations (ESCO)? We successfully illustrated the validity of the extracted tasks for the Software development job by comparing it to the tasks enumerated by ESCO. This approach can be repeated for all jobs.

VIE allows for rapid, cheap, reliable, and valid extraction of job information which may complement other traditional job analysis data collection methods (McEntire et al., 2006). The extracted job information can reflect nuances in jobs that may be overlooked during data collection with SMEs because interviewing, observing, and surveying can be laborious and consume a lot of time. Moreover, through traditional means of collecting job analysis data, it is difficult to gather job information that spans geographical boundaries. Due to differences in ways jobs can be performed (from one employee to another), traditional ways of data collection may thus suffer from poor reliability and bias (Dierdorff & Morgeson, 2009; Morgeson & Campion, 1997, 2000; Morgeson et al., 2004; Sanchez & Levine, 2000). Instead of going through thousands of vacancies, through the application of classification models developed here, we managed to condense work activities to a few hundred themes which was determined empirically by the topic modeling method. The number of themes can be increased or decreased depending on the needs and goal of the analysis. In support of the validity of our approach, the extracted tasks and duties demonstrated considerable overlap with existing ESCO job profiles, which were obtained by painstakingly interviewing job holders and job experts. The simplicity of the resulting model did not limit its usefulness, and in fact worked well in extracting the intended information.

There are several ways to validate the extracted job information. First is to compare them with alternative sources of information such as interviewing SMEs (such as job incumbents) or as suggested validation through job inferences (Morgeson & Campion, 2000). As a start, one can enlist the help of job analysts and SMEs to evaluate the extracted information from vacancy texts. For example, SMEs may be consulted from time to time to ensure that the extracted information is still up-to-date or accurate. This is done by noting the inter-rater agreement between the experts and the model. The agreement is usually quantified using measures such as Cohen's kappa or intra-class correlation coefficient.

Another way to validate job information is through replication, data triangulation, and through an indirect inferential routing (Binning & Barrett, 1989b). For example, to validate work activities (or work data) such as job tasks, one can

administer questionnaires to job incumbents asking them about their work activities or compare the information to data obtained from conducting a survey, e.g., data from O*NET (Campion et al., 1999). This validation procedure can be replicated on other types of work activities extracted from vacancies to assess if the method consistently generates valid work activities.

In general, to further strengthen the validity, results (or part of) can be compared to independent data sources or other 'standards' (such as the survey or expert data). Though it is easy to view vacancy information extraction as a mechanistic means of extracting information, the input of domain experts, albeit minimal, is still critical from the start of model building all the way to validity evaluation. Finally, validity assessment procedures outlined by Binning and Barrett (1989b) and other inference based procedures can also be applied.

Limitations and Future Research

As demonstrated here, VIE is a promising approach to automate the extraction of job information from vacancies. These vacancies are written by hiring managers, recruiters, and job experts, and as such we can be confident that the information extracted have at least some levels of validity. However, as with any other sources of data, one must be aware of potential sources of bias and inconsistencies that may creep into the analysis. For example, it has been noted that vacancies posted by recruitment agencies often lacks detail, and that the information provided is sometimes vague as compared to vacancies posted by organizations themselves. This can be mitigated by analyzing large numbers of vacancies because they can supplement one another (i.e., what is missing in one vacancy might be present in another vacancy of the same job). Another is that online job data can also be considered as secondary data. The primary purpose of vacancies is to attract candidates and not to give a full description of the job which means that the included information is more likely about positive aspects of the job and may not mention for instance the boring or repetitive tasks. For example, for nursing jobs, some frequently occurring tasks are washing patients, changing patients, cleaning beds, checking temperature. Possible future research is to conduct an experiment that would test whether the mention of these tasks would attract applicants. However, as was shown here, some vacancies do mention repetitive and mundane tasks as long as they are essential for job performance. Third, some jobs do not appear in online vacancies (Sodhi & Son, 2007), potentially leaving out relevant information and jobs. Hence, it is still crucial to triangulate the results from VIE to other sources of job information. Fourth, job vacancies may provide an asymmetric source of information about jobs in that it only considers jobs from the point of view of employers. Hence, information extracted from vacancies may not be directly useful to study job seeker preferences.

Finally, the information extracted by the techniques developed here can be

further used as input to other analysis such as monitoring trends of work activities required by organizations across time, occupations, companies, and geographical regions given that these types of information are generally provided in the vacancies. Also, one can build a taxonomy of jobs based entirely on work activities to examine which jobs share tasks, which will help identify job redundancy as well as study how individuals can be retrained to transfer from one job to another. Finally, task groups recovered here can be used as a basis to verify or develop theories on job performance, since the task groups may be used to define factors that would determine job performance (J. P. Campbell et al., 1993) with its emphasis on behavior and its influence on observable results and outcomes.

PART 4

TEXT MINING APPLICATION IN CAREER RESEARCH

Predicting the next job and when to change jobs

In this Chapter we turn our attention to using resumes, another abundant source of information about candidates. Resumes are good sources of information for career research since it contains information about candidates job transitions. We ask:

RQ4 How to build a model that can predict when someone is going to change job?

RQ5 How to build a model that can predict the mostly likely next job of an individuals?

RQ6 How to use the models from RQ_4-RQ_5 to find similarities among jobs and does industry differences play a role in mobility, that is, some industries exhibit higher number of job transitions of employees?

Nowadays, the go-to online platforms for searching jobs are professional networking and employment websites (Vicknair et al., 2010). Job seekers regularly visit these websites to look for job openings. On the other hand, recruiters and employment agencies use these websites to source candidates, in fact, some develop their own websites for the purpose of maintaining databases of job candidates (Parry & Wilson, 2009; Thompson et al., 2008). Resumes uploaded by job seekers form a rich source of data to study career trajectories. These data extend beyond organizational boundaries and facilitate inter-organizational analysis at an unprecedented scale. The emergence of such massive data, combined with the availability of new analytical techniques therewith offer fresh opportunities to study career paths for individuals transitioning from one job to another, both within and across organizations.

Due to their tendency to focus exclusively on either the psychological or the socio-economic view, researchers have tended to ignore other more data-driven predictors of the actual job transitions, for example, how previous jobs shape and create opportunities for subsequent jobs. This is an unfortunate omission because clearly individual career trajectories are at least in part determined by the jobs that people have held in the past. Moreover, previous studies oftentimes relied on (quantitative) employee data limited to specific organizations or industries (Joseph et al., 2012), which precludes the investigation of inter-organization, inter-industry, and the external factors affecting job changes.

Researchers have long called for a more transdisciplinary approach to the study of job changes (Kalleberg & Mouw, 2018; T. W. H. Ng et al., 2007; Rosenfeld, 1992). Integrating economic perspectives with individual differences may explain predilection to change job but is inadequate to predict *when* job change happens or to explain *why* a certain job change is enacted (Sullivan & Baruch, 2009). This study answers this call for transdisciplinary approach as well as addressing the "when"

and "why" of job change at the individual level by leveraging an analytics approach. By analyzing massive amounts of career trajectories (extracted from resumes) using novel analytical techniques and employing theories to shortlist potentially important antecedents, this study set out to simultaneously examine mobility at the individual level and to establish which variables strongly predict such job change.

In terms of which variables are important, longitudinal variables seem to explain much of the variability in job mobility (e.g., age) (Arthur, Hall, & Lawrence, 1989; Ng & Feldman, 2010; Sicherman & Galor, 1990). This is because career is viewed as a "moving perspective" and the passage of time is associated with the development of both social and human capital (Sicherman & Galor, 1990). Hence, this study is focused on using temporal factors that characterize the individuals to predict job change related outcomes. For example, research have noted that age, *time on the labor market*, and *individual preferences* (e.g., preferences on long term employment vs. short assignment) determine which of the available transitions are enacted (Segers et al., 2008). This study, therefore, relies on these and other temporal characteristics (e.g., number of job changes, inter-job times, previous jobs, and tenure in previous jobs) resulting from the analysis of the data. Compared to psychological and other socio-economic variables, temporal characteristics are relatively straightforward to operationalize from resumes.

Adapting machine learning approach enables the extraction of patterns (that are hitherto unknown) from data in a manner that is replicable. This will remedy the lack of consensus of how to measure the concepts associated with job change. The extracted patterns could be used to build concrete models that are of direct relevance to practitioners (i.e., recruiters and HR managers) example is when to approach individuals for a new job opportunity or identify job seekers wishing to change jobs. The results could also be used as starting point for a more in-depth analysis and further clarification of career constructs, hence contributing to the advancement of theory.

Objectives

The overall objective of this study was to build models to predict two related outcomes, namely, the timing of job transitions and the most likely next job, and to elucidate which variables are antecedents of such outcomes. Specifically, we set out to (i) predict whether a given candidate will change jobs within a specified time (binary outcome), (ii) predict when the candidate will change jobs (continuous outcome) and (iii) determine the most likely next job(s). The models were further examined to determine the effects of *employment history, employment maturation, educational attainment, gender, nationality* and *location* on the two outcomes. Maturation was operationalized by computing the age, length of work-life, and number of jobs held by individual.

The first specific objective is addressed by applying classification models,

more specifically Random Forest (Breiman, 1996) and Gradient Boosting (A. J. Ferreira & Figueiredo, 2012; Schapire, 2003). Both classification models yield information about variable importance which was then used to determine which variables are influential in predicting job change. Further tests were conducted to establish the link between the antecedents and job change. The results of the two classification models were compared to a hierarchical Bayesian approach for modeling transitions (Fader & Hardie, 2005; Glady et al., 2009). The hierarchical Bayesian model explicitly models the number of transitions and the time between transitions. This model was tested in this study because it was a temporal model and uses fewer parameters. Therefore, it is relatively straightforward to derive time related outcomes, such as how many future job transitions may be expected and how long someone is likely to stay in a job. The second and third specific objectives were simultaneously addressed by co-training two Long Short-Term Memory (LSTM) networks: one that predicts when someone is going to change jobs (within or across organizations) and another that predicts what the most likely next job is.

As offshoots of these models the following are investigated:

(1) Is it possible to define similarities among jobs by investigating the career trajectories exhibited by individuals? From this perspective, two jobs are related when individual frequently transit between them and dissimilar when candidates rarely transit between them. Also, taken as a whole, career paths can be used to guide individuals who want to pursue specific career trajectories (Joseph et al., 2012).

(2) Do industry differences play a role in mobility in that job transitions are more frequent in some industries than in others?

A related concept to job change is job mobility which is defined as the patterns of intra- and inter-organization transitions. Job mobility theories provide some ideas about job change; hence, they have been used here to determine which variables to include in the models. In this paper, job mobility is interpreted as physical mobility, as there is another type of interpretation of job mobility that is psychological mobility which does not necessarily translate to actual job change.

Career theory and job mobility

Theories have been proposed to explain how careers unfold. One concept posits that individuals vary in aspects pertaining to their individual circumstances and subsequently change their careers to respond to both internal and environmental changes. From this perspective, individuals are no longer considered as being driven solely by relatively stable values and attitudes but can change and adapt through constant evaluation of their choices and options. The evaluation is based on their values, interests and relationships which may vary over their lifetimes and on constraints and opportunities in the environment. This idea is called the *kaleidoscope career model*, because it likens the changes in one's career to pattern changes in a

kaleidoscope (Mainiero & Sullivan, 2005). By rotating the tube, different patterns emerge in the same way individuals adjust their priorities to match their characters and their current contexts resulting in different career patterns. This may well be a better explanation of actual job movement than both the *protean* and *boundaryless* concepts because it considers the ever-changing structure of the labor market and the undulations that occur within individuals as they progress through their careers.

Instead of departing from career theory, other researchers have investigated specific factors that may cause an individual to change jobs or that may facilitate job transition, such as the roles of recruiter and candidate preferences (Malinowski, Keim, Weitzel, et al., 2006), social ties and contact (Wegener, 1991), and opportunities for mobility (T. W. H. Ng et al., 2007) . The same way that time is inherent in the study of careers, so it is in explaining job mobility. Incorporating time in the analysis of job mobility has been the motivation for using competing risk models to analyze transitions (Hachen, 1992).

Researchers focusing on physical job mobility have offered three theories on job mobility. The first is the *reward-resource model* which claims that individuals change job to maximize job reward. The change of job is a function of the individual's resources such as education and experience. The second model, called the *limited opportunity model*, places an emphasis on macro-societal factors that restrict job mobility. For example, social restrictions on women and non-Whites constrain their mobility on certain jobs. It also applies to cultural norms which dictate how employees should view career success, presumably resulting in employees only following specific career paths. The third and last model is the *vacancy competition model* which directs attention to organizational and industrial determinants of mobility patterns. This model highlights the type of labor market one is exposed to. Some organizations and industries have closed employment relations hence lower rates of involuntary moves, and higher rates of within-employer and upward moves. The three models imply different levels of analysis: the reward-resource model operates at the individual level, the limited opportunity at the societal level and the vacancy competition at the organizational level. Hence, the study of job mobility should not only focus on the type of mobility but must set out to identify relevant antecedents down to the level of the individual.

This study builds on elements from the three models to identify factors for mobility that can be extracted from resumes. Consequently, the analysis is at the individual level.

Candidate sourcing problem

Studies on job transitions are not limited to the traditional career and job mobility fields. Due to the rise of analytics and the relative ease of collecting employees' data, an increasing number of researchers are studying the phenomenon of job changes. These studies tend to focus more the practical implications than

on building theory, hence the need to create a middle ground between theory and practice. This is one of the goals of this paper. Next we provide a review of related work about a related problem in candidate sourcing.

Literature abounds on the candidate sourcing problem (Rafter, Bradley, & Smyth, 2000; Smyth, Bradley, & Rafter, 2002), which is a type of job recommendation problem in that it seeks to gauge the fit between candidates and vacancies. Many approaches for job recommendation have been proposed in literature (Al-Otaibi & Ykhlef, 2012; Hong, Zheng, Wang, & Shi, 2013; Lu, El Helou, & Gillet, 2013), some focus on improving recommendations by means of data mining and analytics (Abel, Benczúr, Kohlsdorf, Larson, & Pálovics, 2016; Pacuk, Sankowski, Węgrzycki, Witkowski, & Wygocki, 2016), while others focus on system implementation (Abel, 2015). Researchers also differ on the data types they leverage in addressing the sourcing problem. Since in this study we attempted to offer yet another approach to improve job recommendation using a dataset containing parsed CV data, we will only consider job recommendation literature discussing methods which can be applied to such a dataset.

Content-based job recommendation systems

Guo, Jerbi, & O'Mahony (2014) tested several content-based approaches on a data set provided by CareerBuilder. Comparing the cosine similarity of Bag of Words (BoW) representations with tf-idf weighting between vacancies a job seeker already applied to and all other vacancies in the corpus yielded the best results. Heap, Krzywicki, Wobcke, Bain, & Compton (2014) used the co-occurrence of key words in the job title and job seeker's previous job titles, and in the vacancy description and job seeker's previous job description. Especially the co-occurrence of words in the descriptions resulted in a high recall, implying that indeed past positions may be pertinent to predicting future positions.

Liu, Rong, Ouyang, & Xiong (2017) consider the problem of recommending employers to undergraduates by first computing the similarity between students and graduates (i.e., alumni) for whom the employer is known, and recommending the employer of the graduate student with the highest similarity.

One problem with comparing job seekers and jobs is that the characteristics of the job seeker might not match the characteristics of the job. To overcome this problem, Malherbe, Diaby, Cataldi, Viennet, & Aufaure (2014) propose an approach in which the similarity between job seekers and jobs is computed by weighting the cosine similarity between feature vectors of the job seeker and the job. The weights, which are trained using a Support Vector Machine (SVM) with a vanilla linear kernel, represent how relevant the cosine distance between a user feature and job feature is. The feature vector is obtained by computing the tf-idf vector from the textual features of the job seeker and the job provider. Using this method to compute similarity outperformed other similarity-based methods including basic cosine

similarity (no weighting), field weighting (weighting the individual feature vectors instead of the cosine distance between vectors), and average field to field similarity (each cosine distance between a user feature and job feature has equal importance).

Career move prediction

Several methods have also been proposed to predict job seekers' next career move. Li et al. (2017) applied a Long Short-Term Memory (LSTM) network to model job sequences, where the context of the job seeker is encoded and added to the LSTM network. They used the seq-to-seq LSTM flavor. The results suggest that fitting an LSTM model outperforms algorithms which do not incorporate information about the career paths and those that use a first-order Markov property.

Although content-based job recommendation systems and career move predictions provide interesting methods to determine which candidate might be most interesting for a given vacancy, they ignore the effect of time in their recommendation. That is, even if a candidate is likely to switch to a certain job or is found to be a good match for a vacancy, this will not be useful to a recruiter if the expected tenure in his/her current job is only a few years. To account for the effect of time on the job Wang, Zhang, Posse, & Bhasin (2013) proposes a survival model in a Bayesian framework. Recommendations are made based on the integrated hazard rate on some interval of interest, which can be interpreted as the probability the job seeker will switch to the proposed job within a specific duration, given having occupied his/her current job up until time t. In particular their push-based scenario, in which recommendations are "pushed" to passive job seekers. Results of this push-based scenario show the downside of the impression dataset: there are usually just a few positive responses and ignoring the probability of the job being relevant to the job seeker can lead to bad recommendations.

Data, variables, and models

Data Description

Models were fit on data extracted from candidates' CVs (or resumes) which were provided by USG People, a recruitment and human resources company based in the Netherlands. The CVs were uploaded by candidates to the USG people managed Dutch job board G.us[3]. There are approximately 305,912 unique candidates in the dataset.

Since the raw CVs were created in text format, a CV parser was applied to extract the relevant data from the text. The parsed data were stored in different tables. The tables that are of interest to this study are:

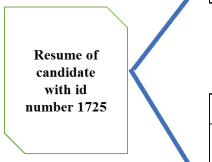
- Candidate table. Each row corresponds to a specific candidate and provides general information about the candidate including gender, current residence, current employment, type of driving license, year of birth and nationality. Each candidate was assigned a unique id.

[3] https://www.gus.nl/

- Employment history table. Contains all previous jobs of candidates as mentioned in their CVs. One row for each job previously held by a specific candidate. Apart from the job title, jobs are further classified into the following hierarchy: function class -> function group -> function. The function is the lowest level in the hierarchy and contains 1330 classes. Also, the sector classification of each job is provided. Other attributes include the name of the employer and the id of the candidate who held the job.

A sample CV and the corresponding rows (not all columns are shown) in the two tables are provided in re 2. The two tables were then merged into one big table by matching the candidate id fields in both tables.

The parsing and subsequent merging resulted in a table with 2,010,112 rows. To limit the variability of jobs and make the analysis manageable, *function* is used instead of job titles as our substitute for job. Using *function* is better than dealing with specific job titles because oftentimes the same job may be described using different titles by the candidates (who were asked for this using open-ended questions). This leads to redundancy and an explosion on the number of job titles which may add unnecessary complexity to the analysis. Therefore, for the purposes of this study, job refers to function.



Entry in the Candidate table

candidate_id	location_info	gender	drivers_license	year_of_birth	nationality_short
1725	leiden	1	BE	1960	NL

Entry in the Employment history table

candidate_id	job_title	start_date	function	function_class	function_group
1725	Operations Manager Productiemanager	1977-01-01	hoofd bedrijfsbureau planning en controle	Beleid en bestuur	beleid en bestuur (overig)
1725	Interim Manager	1993-01-01	manager	Beleid en bestuur	beleid en bestuur (overig)
1725	Adviseur-Consulent Diversiteit	2010-01-01	adviseur civiele techniek	Engineering	ingenieurs, constructeurs en technici gww
1725	Stafmedewerker Diversiteit	2012-04-01	beleidsmedewerker (overig)	Beleid en bestuur	beleidsmedewerkers

Figure 4.1. Extraction and Conversion of Raw Text CV Data into Tables.

Data were divided into a training, validation, and test set. The split was done randomly; with 209,164 candidates being allocated to the training and validation sets and 89,663 candidates to the test set. All models were trained using the training set. Table 4.1 provides some pertinent summary statistics for the variables describing job transitions.

Table 4.1. Summary Statistics for the Variables Describing Job Transitions.

	Min	Max	Mean	Median	SD
Number of transitions	1	64	5.33	5	3.20
Time between first job and current job (in months)	0	963	157.64	123	129.35
Sum of the logarithm of inter-job times	0	79.27	12.77	11.85	8.74

Variables

The variables considered in this study were chosen based on existing studies or models of career and job mobility.

The three models of job mobility as expounded by Hachen (1990) provided the basis for the inclusion of some of the variables. According to *the reward resource model*, job seekers move to better jobs to maximize their job rewards. Job change is viewed as function of an individual's resources such as education and previous job experience. Hence, this study included ***education level*** and ***number of previous jobs*** as antecedents of job change. Education level expectedly reflects the individual's education attainment; the higher the level the greater the associated resource (e.g., human capital). The number of previous jobs points to the individual's accumulated job experiences which may encapsulate knowledge, skills, and abilities (KSAs) acquired from previous employment (e.g., career capital). Aside from capturing accumulated knowledge and experience, the number of job changes may also indicate seniority in the labor market.

The *limited opportunity* model on the other hand, examines macro-societal factors in determining mobility options for individuals. One aspect of the model states that mobility patterns differ because of social restrictions imposed certain types of jobs based on one's gender or race. Hence, gender and nationality variables were also included in this study. The choice for **gender** is also empirically supported by the findings in other studies (Segers et al., 2008; Sullivan & Mainiero, 2007)

The third model of job mobility is the *vacancy competition model* which highlights the organizational and industrial determinants of mobility patterns. For example, jobs in some organizations or sectors which emphasize closed employment relations characterized by internal labor markets should have lower rates of job exiting and between-employer job shifts. Considering this, the feature ***sector of previous jobs*** was also included in the study.

Apart from the three models of job mobility, examining the career literature also provided clues as to which variables are predictors of job mobility. As was mentioned earlier, the study of Segers et al. (2008) found gender differences in mobility patterns. Whereas the limited resource model presupposes that macro-societal restrictions limit the job mobility of women, Segers et al. (2008) further argued that women tend to have higher psychological mobility because they are better able to see multiple career paths than men. Another variable underscored in their study is the role of age, that is, they observed that increased age is associated with lower physical job mobility. Aside from age it is also important to take into account generational differences among individuals (Lyons et al., 2015). According to the kaleidoscope career model (KCM) people born between 1965-1983 (Generation X) have higher needs for authenticity (choices that permit people to be true to themselves) and balance (equilibrium between work and nonwork) than people born between 1946-1964 (Baby Boomers) (S. Sullivan et al., 2009). Hence, in this study both ***age*** and the ***year of birth*** of candidates entered the analysis. The ***year of birth*** is discretized into 5-year intervals. Age and year of birth are expected to be correlated and may be presumed as redundant, however as pointed out by Guyon & Elisseeff (2003) the addition of redundant variables (though must not be perfectly correlated) could reduce noise and consequently yield better model performance. In this case the purpose of adding year of birth is to convey from which generation a candidate belongs in.

Integrative frameworks that combine ideas from both protean and boundaryless models also offered guidance in the choice of variables considered in this study. According to one framework, individuals exhibit different levels of protean and boundaryless orientations and may be categorized into 16 profiles pursuant to whether they are high and low on 2 protean dimensions (i.e., self-directed career management and values driven attitudes) and 2 boundaryless dimensions (i.e., psychological and physical mobility) (Briscoe & Hall, 2006). A study by Segers et al. (2018) demonstrated that contextual factors and industry influenced the presence of these career profiles. For example, the protean career architect profiles (i.e., high in all four dimensions) are overrepresented in the Scandinavian countries and the Netherlands, whereas Belgium had a higher proportion of individuals with the curious/wanderer profiles (i.e., individuals who are physically and psychologically mobile but not values driven or self-directed). In the same manner, the profiles associated with high mobility are more likely to be found in the industrial sectors of health and social work, counseling, science and research, marketing, government, sales, and education. Whereas profiles that demonstrate low mobility are evident in industries including construction, manufacturing, transportation and logistics, and Internet/new technologies. To capture these differences across industries, the

variables reflecting the ***function classes of previous jobs*** were included.

It has been recognized that individuals change the patterns of their careers over the course of their lifetimes (Mainiero & Sullivan, 2005; D. Super et al., 1996). These changes may happen in response to internal changes such as those due to maturation or environmental changes such as being laid off. According to the life-span, life-space approach of career development as advanced by Super (1980), these internal changes include the changes in the roles that people play as they mature. A role is defined as a set of expectations that others have of a person occupying a position. Example of roles include the student, worker, and homemaker roles. In the context of occupation career, the worker role changes their specific behavioral definitions with increasing age. In the same manner, position, which defines the worker role, may also change when the individual changes occupations. Career decisions and how long someone is already in the labor market provide indications in which phase one is currently in and as to what type of career path one is intending to take (i.e., traditional or nontraditional). Making frequent job moves may indicate that the employee in question is in the exploration phase. This phase usually happens in the early stage of career building. On the other hand, when one is in the establishment (i.e., pursuing a plan to make a place for oneself in a new job) and maintenance (i.e., holding one's own in the new job) phases then no or low mobility is observed. Hence, incorporating a variable that considers how long an individual is in the labor market already (i.e., ***time in the labor market***) may capture elements of these anticipated phases and career behaviors that individuals may go through over the course of their career lifespans.

The preceding variables are augmented by including variables which may capture further individual differences in career enactment as well as how individuals change their career patterns over time. Drawing further inspiration from the preceding models, this paper also included ***time in the current job, time in the previous jobs*** and ***an aggregate measure of time between previously held jobs.***

Operationalizing the variables

To compute the variables, calibration and observation periods were set. Both the calibration and observation periods were determined by date thresholds. In this study the end of the calibration period was set to Dec 31, 2010, and the end of the observation period was set to Dec 31, 2012. This partition serves two purposes; first, for the classification models, model training was done within the calibration period and predictions were made between the end of the calibration and the end of the observation period. For example, predicting whether a candidate has changed job after the calibration period until the end of the observation period. For Bayesian and LSTM models, it is used as a validation period. The use of this partition will be

further elaborated during the discussion of the model.

Using the merged data set, the variables were calculated as follows:

Education level is the highest education attainment of the candidate. It is treated as a categorical variable with four categories, namely, Beroepsonderwijs (Vocational), HBO (Applied degree), Master, Universiteit (Research degree), and Voortgezet Onderwijs (Secondary education). The ***number of job changes*** is operationalized as the number of previously held jobs (excluding the current one) before the calibration period. The ***gender*** variable is the gender of the candidate. It is coded as 1 for female and 2 for male.

The ***sectors*** for previous jobs are the industry classification of jobs. For models that do not explicitly model time, only the last three jobs were considered, hence sectors of the previous 3 jobs prior to the most recent one. This choice is supported by some studies that showed that individuals' chances of moving to new positions are a function of their current states or positions, not of the entire sequence of their career. On the other hand, other studies subscribe to a path-dependent model in that early positions indirectly affect occupational attainment. This study recognizes that individuals' current careers are determined by previous positions or states, however, there is no need to look too far in the past since the effect is only indirect. Hence, only the three previous jobs were considered. Likewise, only the last three jobs were considered for the function class variables.

The ***age*** variable is computed by subtracting the year of birth to the end of the calibration and measured in months. This is an estimate and may be off by a few months. Instead of using the actual year of birth of the candidates, the years were discretized into equal sized bins of size 5 years (except for the first bin which has no lower limit) and each bin is recoded into integers with values from 0 to 8. The 0 value is assigned to candidates who were born before 1960 and the last value 8 is assigned to the candidates who were born from 1995 to 2000. As was mentioned earlier, this is done to capture from which generation a candidate belongs in (e.g., Generation X) and also to cut down the range of the values which sometimes improve model performance (Lustgarten et al., 2008).

For the following variables, since this study is limited by the information that can be extracted from CV, some trade-offs in accuracy had to be made. The **time in the labor market** is computed by subtracting the start date of the first job to the end of the calibration period and is measured in months. This variable should be interpreted as time from the start of the first job to the end of the calibration period. This does not coincide exactly with how long someone is in the labor market since a person may be out of the labor market for some period of time or is retired or has passed away before the end of the calibration period. There is no way to tell this from the information in the CV, hence the way it is measured is an upper bound

for the actual time in the labor market. The ***time in the current job*** is measured by determining the duration in months between the start of the current or latest job and the end of the calibration period. Some employment histories of candidates are right censored, that is, it is difficult to pinpoint what happened exactly after the start of the last job mentioned in the CV. If there was a next job, then it was clear that there had been a transition. Otherwise, either the candidate had stopped working or had transferred to another job but failed to disclose this in the CV.

Time between jobs or ***inter-job times*** are computed as the differences between the start date of two consecutive jobs. Inter-job time approximates tenure but may not represent actual tenure in some instances. For example, an individual may have taken a break from work or have been out of work for some time before the next job (e.g., time spent searching for job). This also does not consider if someone held multiple jobs at the same time. The **aggregate measure of inter-job times** is the logarithm of the sum of all computed inter-job times for each candidate. Additionally, for some models in this study, the three previous inter-job times were included as an input variable.

Table 4.2 below provides a summary of the independent variables used in this study. As was mentioned previously, the sectors, function groups and inter-job times of the three previous jobs are included in the classification models.

Table 4.2. Codebook of Variables

Variable	How to compute	Type	Basis
Time in the labor market	Length of time between the start of the first job and now (in months)	Numerical (Continuous)	Temporal (Super, 1980)
Time in the current job	Length of time between the start of the latest job and now (in months)	Numerical (Continuous)	Temporal (Super, 1980)
Labor market time before the start of the current job	Time between the start of first job and the start of the current job (in months)	categorical	Temporal (Super, 1980)
Inter-job times	Length of time between the start date of two consecutive jobs as written in the CV (in months)	Numerical (Continuous)	Temporal (Super, 1980)
Aggregate inter-job time	Logarithm of the sum of inter-job times	Numerical (Continuous)	Temporal (Super, 1980)
Number of jobs so far	Count of jobs held so far (including part time jobs, volunteer work, traineeships, etc. As long as they are mentioned in the work history in the CV) up to the end of the calibration period	Numerical (Discrete)	Reward resource model (Hachen, 1990)
Age	Time difference between birth year and now (in years).	Numerical (discrete)	(Lyons et al., 2015; Segers et al., 2008)
Year of birth	Year of birth	Numerical (discrete)	(Lyons et al., 2015; Segers et al., 2008)
Sector	Sector classification of jobs	Categorical	Vacancy competition model (Hachen, 1990) and industry differences (T. W. H. Ng et al., 2007; Wegener, 1991)
Function Group	Function group classification of jobs	Categorical	(Hachen, 1992)
Gender	Gender of candidate (Male or Female)	Categorical (2 categories)	Limited opportunity model (Hachen, 1990; Sousa-Poza & Henneberger, 2004)
Location	Location of jobs	Categorical (193 categories)	Contextual variable (Sørensen & Tuma, 1978)
City Cluster	City cluster of the location of jobs	Categorical (4 clusters)	Contextual variable (Nimczik, 2017)
Nationality	Nationality of the candidate	Categorical (83 categories)	Limited opportunity model (Hachen, 1990)
Education	Educational attainment of the candidate	Categorical (4 categories)	Reward resource model (Hachen, 1990)

Predicting transition using Classification models

Classification models were constructed to address the first objective which was to predict whether a candidate will make a transition within a specific period—in this case within a 2-year period. The constructed models are further investigated to unravel the variables that are crucial for the prediction task. Three classification algorithms were used, namely, Random Forest, Extreme Gradient Boosting, and Elastic-net Generalized linear models (Elastic-net GLM). The preceding algorithms were selected because they have been shown to be effective in other applications, they are rapid to train, and most importantly each provides information on the importance or contribution of each variable (Breiman, Friedman, Stone, & Olshen, 1984; Ferreira & Figueiredo, 2012; Liaw & Wiener, 2002; Pacuk, Sankowski, Węgrzycki, Witkowski, & Wygocki, 2016). Additionally, both are scalable when the observations are numerous (Breiman et al., 1984; T. Chen & Guestrin, 2016). Likewise, there exist efficient implementations of Elastic Generalized linear models (Friedman et al., 2016). Estimated coefficients for the input variables in Elastic GLM may be inspected to assess their influence on the outcome. For further information about these algorithms, the reader is referred to the following publications (Breiman et al., 1984; de Oliveira, 2019; Zou & Hastie, 2005).

The response variable is a binary variable indicating whether a candidate is going to change their jobs between the end of the calibration period and the end of the observation. This is just whether a candidate is going to change job within 2 years. This response variable is chosen to determine how mobile a candidate is within the next two years since the average tenure is about a year.

Training and evaluation of the Classification models

The training data set contains 192,205 candidates. The end of calibration period was set to 2010-12-31 and the end of the observation period was set to 2012-12-31. The outcome variable for each candidate is computed within this 2-year period, that is, if the candidate had changed jobs at least once within this 2-year period then he/she is said to have made a transition.

Models were evaluated using 10-fold cross validation stratified on the response variable to maintain the relative proportion of categories in the response variable during model training. For each run, one of the folds is used for validation and the rest is used for training. Parameters were tuned for each run. The final model in each run, i.e., with optimized parameters, is used to predict the values of the response variable in observation in the validation set.

Table 4.3 provides a summary of the parameters associated to each algorithm along with indications whether a specific parameter was optimized or was set to default values following the implementation. Parameters were tuned using grid expansion search.

Table 4.3. Classification models and their associated parameters.

Random Forest	Gradient Boosting	Elastic-net GLM
Number of trees: 500 Number of variables: ***tuned***	Number of boosting iterations: ***tuned*** Booster: tree Eta: .3 Max depth of trees:6 Objective: logistic regression for binary output	Alpha: 0.5 Response type: binomial Lambda: ***tuned***

Note. Parameters were either tuned or set to default values

The area under the ROC curve (AUROC) was chosen as the final evaluation metric. A permutation test was conducted to assess the significance of the computed AUROC for each model to prevent considering AUROCs that are below the sampling noise level and hence not considered significant. As was highlighted before, more than classification performance, this study is concerned with identifying variables that were strong predictors of whether someone is going to change job or not. Fortunately, the algorithms considered here provide straightforward ways to assess variable contribution. For the Random Forest and Gradient Boosting models, each run in the cross-validation part the top 10 variables were identified and the top 10 variables with the highest mean reciprocal rank across runs were reported. For the Elastic-net GLM, all variables with nonzero coefficients were identified.

The data was preprocessed by converting all categorical variables to one-hot encoding and numerical variables were rescaled to 0-1 range. The rescaling of numerical variables is needed to prevent variables with large range from unduly influencing the analysis.

Counting job transitions using Hierarchical Bayes

The goal here was to predict the number of job transitions for a specific time period by modeling explicitly the number of transitions within a time period and the length of time between two consecutive transitions (this is related to the first objective). Compared to the classification models, the resulting models here are generative in nature because they model the process of transitions. A generative model is sometimes preferred because it is possible to derive quantities (e.g., number of transitions for any specified time duration) which will otherwise require retraining for classification models. For example, predicting whether a candidate will change job within a 1-year period rather than within a 2-year period would require redefining the outcome variable so that it will consider only the next 1 year. Another advantage of this method is that it uses only a handful of variables, namely, the inter-job times,

the number of previous job changes, and the time since the start date of the first job.

Predicting the number of transitions

In order to predict the number of job transitions a candidate will make in any specified time duration, models from database marketing were borrowed and tailored to fit this purpose. To the authors' knowledge, this is the first time that these models were applied to the study of job transitions. The models were originally devised to forecast customer purchasing behavior, specifically, models in which customers are assumed to have non-contractual relationship with the firm/company (e.g., no subscription). These models are also applicable to study job transitions by viewing each transition as a "transaction." Some aspects of the models were redefined to make them applicable to predict future job transitions.

The most popular among these models is the Pareto/NBD model which has 5 assumptions. The assumptions are restated in terms of job transitions.

(i) Candidates go through two stages in their working lives. The first stage is called the Active Work Stage (AWS) which begins the moment a candidate starts his/her first job and continues until the end of the last job. In between, a candidate may switch to various jobs with intermittent periods of unemployment. After exiting the AWS, individuals are permanently out of work. Life events that may lead to exiting the AWS include accident, death, or retirement (D. Super et al., 1996 or career shocks (Akkermans, Seibert & Mol, 2018).

(ii) While in AWS, the number of work transitions made by a candidate follows a Poisson process with transition rate λ, therefore the probability of observing x transitions (including transition from the last job) in the time interval $(0, t]$ is given by

$$P(X(t) = x|\lambda) = \frac{(\lambda t)^x e^{-\lambda t}}{x!}$$

This means that the time between jobs is exponentially distributed with transition rate λ,

$$f\left(t_j - t_{j-1}\middle|\lambda\right) = \lambda e^{-\lambda(t_j - t_{j-1})}$$

(iii) A candidate's time in AWS denoted by τ is exponentially distributed with drop-out rate μ:

$$f(\tau|\mu) = \mu e^{-\mu\tau}$$

(iv) Transition rates vary across candidates and we assume they follow a gamma distribution with shape parameter r and scale parameter α:

$$g(\lambda|r,\alpha) = \frac{\alpha^r \lambda^{r-1} e^{-\lambda\alpha}}{\Gamma(r)}$$

(v) Dropout rates vary across individuals and we assume they follow a gamma distribution with shape parameter s and scale parameter β:

$$g(\mu|s,\beta) = \frac{\beta^r \mu^{s-1} e^{-\mu\beta}}{\Gamma(s)}$$

(vi) Dropout rate μ and transition rate λ vary independently across candidates.

Assumptions (ii) and (iv) imply a Negative Binomial distribution of the number of transitions while the candidate is in AWS, while assumptions (iii) and (v) yield the "Pareto distribution of the second kind" for the time in AWS, hence the name Pareto/NBD (Fader & Hardie, 2005).

Researchers have either modified or extended the Pareto/NBD model to account for different distributional assumptions or addition of covariates. For example, Platzer and Reutterer (2016) replaced the distribution of inter-transition times by a gamma distribution leading to the Pareto/GGG model. Fader and Hardie (2009) replaced the exponential distribution for the length of time in the AWS by a shifted geometric distribution. The authors called this model BG/NBD. In this model a probability is set that a candidate will no longer transition to a new job after the i-th transition (start of last job). Abe (2009) made possible the addition of covariates in the Pareto/NBD model.

Several variations (including the original) of the Pareto/Model namely, BG/NBD, Pareto/GGG, NBD, and Abe's Pareto/NBD without covariates were tested for predicting the number of transitions within the 2-year observation period. Models were compared using the least mean squared error.

Predicting when to change job and the next job

The second and third objectives which are to predict when a candidate will change job and the next landing job, respectively, are addressed by co-training two Recurrent Neural Network (RNN) models. The first RNN is used to predict when candidates are going to change job and the second RNN identifies jobs which the candidates will most likely transition to, specifically, the top-k jobs for the candidate. The job here refers to the function classification of a job rather than the actual job title. The function classification is chosen because considering individual job titles contain a lot of redundancies, for example, the same job maybe written in different ways. A candidate is denoted by c and c_j denotes the j-th job of candidate c, hence c_{j+1}

is the next job of candidate *c*. Without loss of generality, and following the current literature, we set *k* equal to 5 (Bylinskii et al., 2017). This would mean predicting the top 5 most likely next jobs for a candidate. The specific RNN architecture that was used here is the Long-short term memory (LSTM). LSTM is a type of recurrent neural network which is suited for sequences and more capable of remembering past events. For the first LSTM, the many-to-one flavor was used in which the outcome variable is the time until the next transition which is computed by subtracting the start date of the next job and the end of the calibration period. The input variables are the sector of jobs, nationality, gender, education level, city cluster, and year of birth of candidates. The LSTM architecture has 36 input nodes, 2 LSTM layers, and one output node. Figure 4.2 below shows the process of converting the original input variables to a format amenable to the application of LSTM. Note that not all variables are shown to save space.

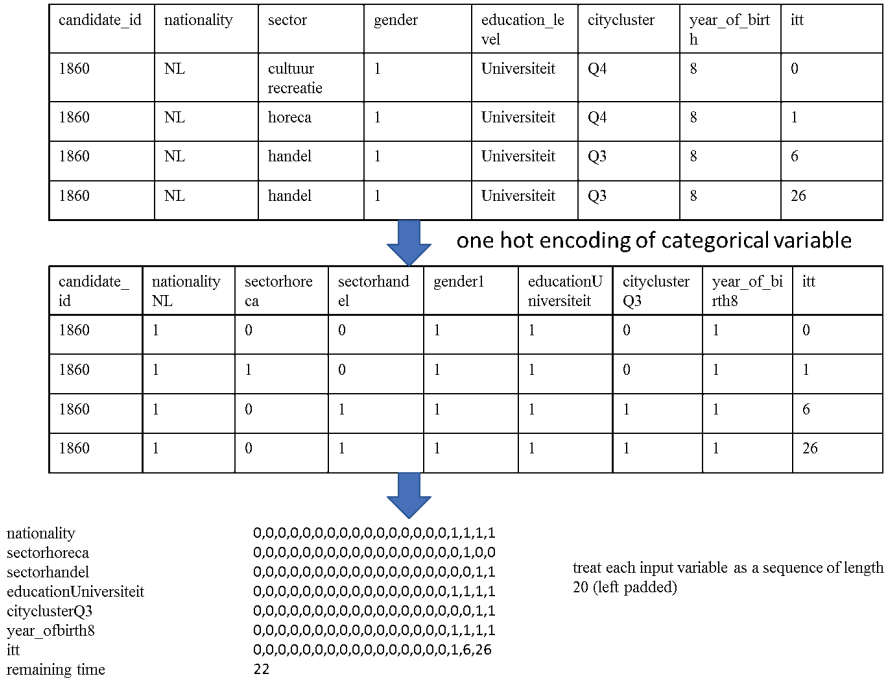


Figure 4.2. Data Input Transformation for LSTM 1

The architecture of LSTM 1 is shown in the Figure 4.3(a). Each sequence corresponds to a variable, for example, Sequence 1 refers to gender.

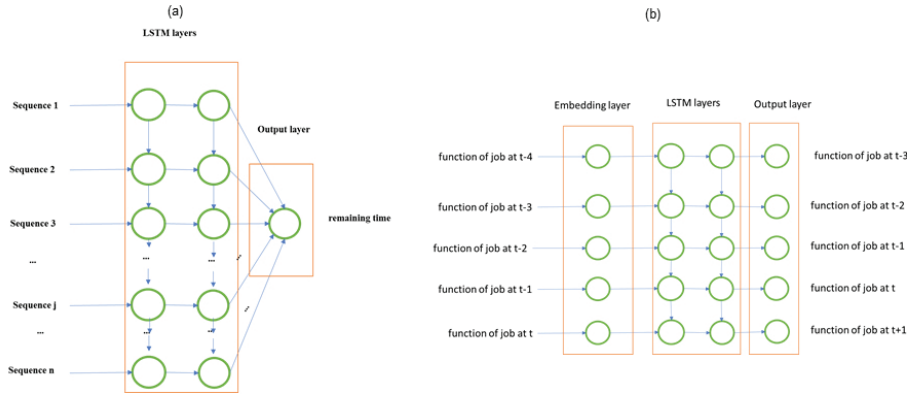


Figure 4.3. (a) Architecture of LSTM 1 Containing 2 LSTM Layers with 10 states each and (b) Architecture of LSTM 2 containing 2 Stacked LSTM Layers with 64 States each

First, categorical variables were transformed using one hot encoding, and each variable was treated as a sequence of length 20. The length of 20 is selected because 99.998% of observations made at most 20 transitions. Observations with less than 20 transitions were left-padded by 0 values. There were variables that were time invariant such as *gender* and *education level*, they were still treated as a sequence, but their values were then constant over time. The outcome variable is denoted by remaining time, which is the time until a candidate changes job to a new job from his/her current job. In summary, the architecture for LSTM 1 has 2 hidden LSTM layers, each with 10 states. The input dimension is 36 by 20 (number of variables × length of each sequence) and there is one output. The delta optimizer was used with the following parameters: decay rate=0.9, eps=.00001, window size=0.000001, clip_gradient=1, and the rescale gradient is equal to 1/batch.size where batch size=32. Weights were initialized using Xavier's algorithm (Kumar, 2017) using Gaussian distribution with standard deviation equal to .32 and with a scale up to 3.

For predicting the next job, we trained a one-to-one LSTM, hereafter referred as LSTM 2. The architecture has 2 stacked LSTM layers with 64 states in each layer. An embedding layer was added to transform the jobs from text to a vector of size 30. Moreover, for regularization a dropout technique was applied with dropout rate equal to 0.2. The output function is *softmax* which was specifically chosen because it returns probabilities for all classes and will be useful later to determine the top-5 jobs. Figure 4.3(b) shows the diagrammatic representation of the LSTM 2 architecture. There were 1329 unique functions and hence the size of the dictionary.

The performance of each LSTM model, that is, the LSTM model for predicting when someone is going to change job (LSTM 1) and the LSTM model for predicting the next job (LSTM 2) was computed. For LSTM 1, performance was gauged using mean squared error and for LSTM 2 we employed the perplexity

measure. The perplexity measure is used since the goal is to predict the top 5 most likely next jobs. The output of LSTM 2 is a probability distribution over the set of possible job and top 5 jobs with the highest probabilities subsequently selected. The mean squared error is computed by means of the following formula:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(predicted_i - observed_i)^2$$

where n is the total number of observation and $predicted_i$ is the predicted value for the response variable in observation i and $observed_i$ is the observed value of the response variable in observation i.

Perplexity, on the other hand, is a measure of prediction error suitable for predicting probability distribution. It is computed using the following formula:

"Perplexity = " e^(-∑_x^{p(x) ln p(x)})

For example, suppose there are only three jobs (3 categories) and the prediction probabilities are (.20,.50,.30), then the perplexity is 2.8001. If the prediction probabilities are (1,0,0), i.e., category 1 is predicted with probability one, then perplexity is 1, which is the lowest value for perplexity. Higher values of perplexity indicate greater errors.

Combining Hierarchical Bayes Model and LSTM 2 for Job Recommendation

The Hierarchical Bayes models with the best performance were then combined to LSTM 2 in a cascading manner: for each candidate, first it was predicted if a transition will happen within two years, then the recurrent neural network was applied to predict the top-5 most likely next function groups. Of course, if the model predicts no transition, then we did not need to apply the second model.

A 10-fold cross-validation was used to assess the performance of the models on independent data. Previously, we mentioned the validation dataset which will only be used for testing. The validation is set aside for the purpose of simulating new candidates in the system and served as the independent data. None of the candidates in the validation set were used during parameter estimation.

Software and codes

All calculations, model training and evaluation were done using the R software. The repository for the codes will be made available upon request.

Results

Predicting a transition

The performance of the three models is shown in Table 4.4. Among the three, Gradient boosting gives the highest cross validated AUC.

Table 4.4. Performance of the classification models.

	AUCROC	Standard deviation	Significance (Using permutation test)
Random Forest	0.79	0.018	P<1e-05
Gradient Boosting	0.81	0.021	p<1e-05
Elastic-Net GLM	0.72	0.002	P<1e-05

Note. Significance of each AUCROC was assessed using permutation test.

The analysis of whether a candidate is going to switch jobs also allows the identification of factors that have the greatest influence on switching jobs. Both Random Forest and Gradient Boosting models give rankings of variables based on importance. Figure 4.4 shows the importance of each variable for the Random Forest and Gradient Boosting models. For the Random Forest, variable importance was assessed in terms of the mean decrease in accuracy and mean decrease in Gini coefficient. The mean decrease in accuracy for a given variable is the corresponding decrease in accuracy of the model due to the exclusion of that variable. The mean decrease in Gini coefficient on the other hand is a measure of how each variable contributes to the homogeneity of the nodes and leaves in the resulting Random Forest.

Time-dependent variables are the most important predictors (i.e., Tcal, timespent, etc.) along with gender, education, current sector and function classifications of jobs. The inclusion of some sectors and functions further support the claim that industry differences play a role in job mobility and may offer evidence on labor market segmentation (Sørensen & Tuma, 1978).

The effects of some selected variables for the Elastic-net GLM model are presented in Table 4.5. They are grouped according to their type.

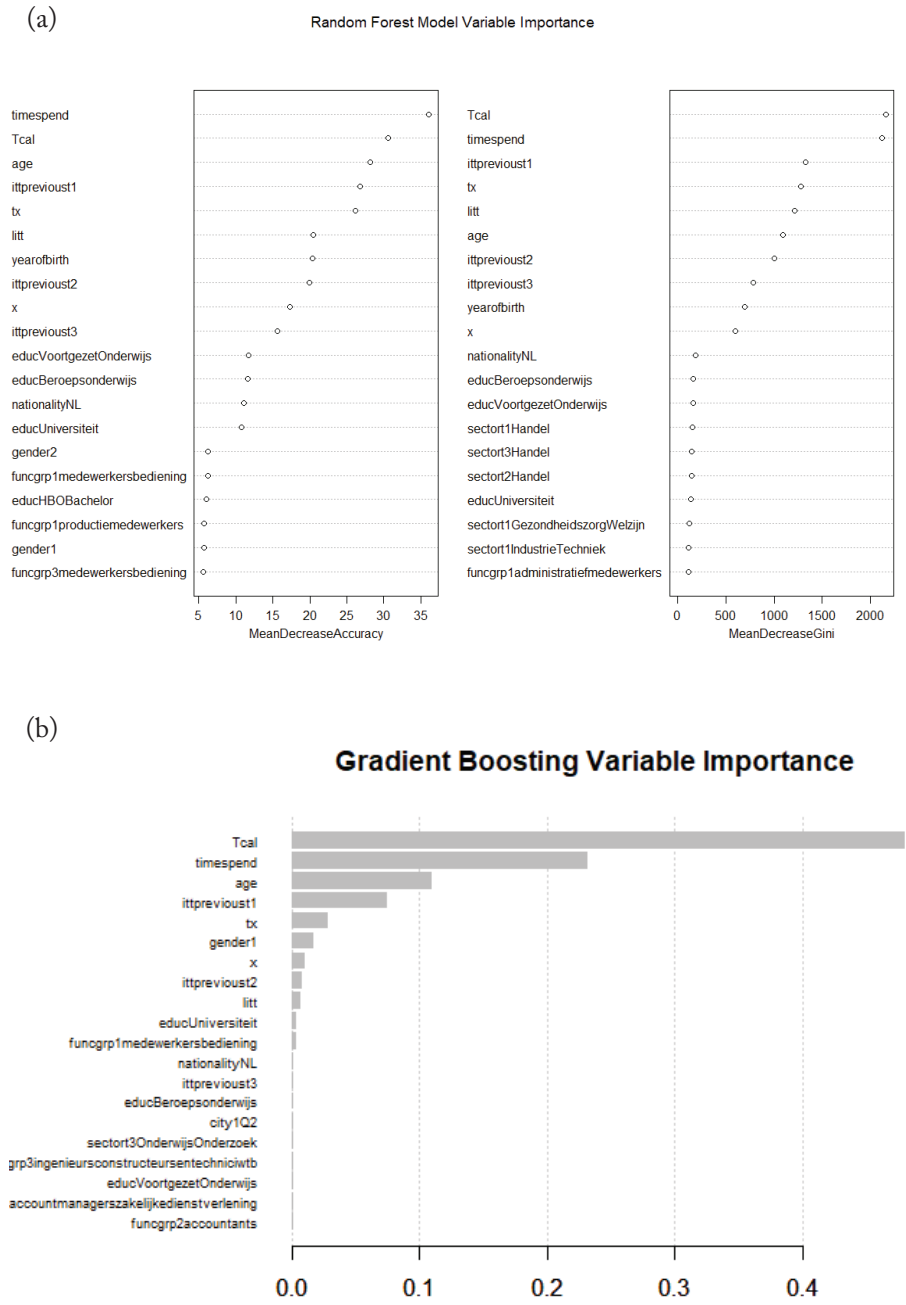


Figure 4.4. (a) Variable importance from Random Forest and (b) variable importance from Gradient Boosting.

Table 4.5. Variable Marginal Effects from Elastic-net GLM.

Variables positively affecting transition	Variables negatively affecting transition	Variable type
x (number of previous job transitions)	itt previous 1 (interjob time), itt previous 1, itt previous 1, age, timespend (time spent already in current job), Tcal (time between start date of first job and the end of calibration period)	
educ Universiteit Arbeidsbemiddeling, Horeca, Onderwijs / Onderzoek, Overheid Non profit Handel and Horeca	Educ VoortgezetOnderwijs Bouw FacilitairSchoonmaak	Education Sector of current job
Docentenmiddelbaaronderwijs, medewerkersbediening, verkopers	algemeendirecteurs, financieeladministratiefmedewerkers	Sector of previous job Function group current job
keukenhulpen, medewerkersbediening, medewerkersbediening Koksenanderevoedselbereiders, verkopers	productiemedewerkers, financieeladministratiefmedewerkers, programmeurs , administratiefmedewerkers chauffeurspersonenvervoer, managersarbeid,	Function group of previous job Function group of job before the previous job

It is clear from Table 4.5 that the number of previous job changes is a strong indicator of the tendency to transition, that is, the higher the number of previous transitions of candidate the more likely more likely he/she is going to make transition within 2 years. The findings here also contradict the commonly held assumption that the longer someone is in the job the more likely he/she is going to make a transition, this is because timespend is negatively related to the outcome variable. However, it is also important to look at the concurrent effects of variables, for example timespent and itt (inter-job times) variables negatively affect the tendency to make a transition, that is, someone who has stayed long in the previous job will likely stay long in the current job.

In the case of sector of jobs, employees in Handel (Commerce or Trading), Horeca (Hospitality), Onderwijs / Onderzoek (Education and Research), and Non-profit sectors are highly mobile. Whereas employees in the Construction sector is less mobile. In terms of function, verkopers (sellers), Docentenmiddelbaaronderwijs (secondary school teacher), keukenhulpen (kitchen helper), and Koks en andere voedselbereiders (chef and other food preparers) are the most mobile, not surprisingly these functions belong to Trade or Commerce and Education sectors (Hachen, 1992; Segers et al., 2008). On the other hand, managers, administrators and programmers

become less mobile once they reached this job position.

Predicting the number of job transitions

Five models were tested, namely, Pareto/NBD, Pareto/GGG, BG/NBD, NBD, and Abe's Pareto/NBD (without covariates), the mean squared error for each model is shown in Table 4.6. Also shown are the mean squared error of a base model where it was assumed that all candidates made at least one transition in the holdout period (i.e., between the end of the calibration period and the end of the observation period). The model which yields the lowest mean squared error are Pareto/NBD, BG/NBD, and NBD which yield an almost 100% improvement from the base model.

Table 4.6. Performance Comparison of Various Hierarchical Bayes models using Mean Squared Error.

Model	Mean squared error
Base model (everyone will make a transition)	0.789
Base model (no transitions)	0.484
Pareto/NBD	0.344
Pareto/GGG	0.521
BG/NBD	0.344
NBD	0.344
Abe's Pareto/NBD (without covariates)	0.535

Note. The models were also compared to a base model that predicts that everyone will transition in the hold out period.

Similarly, to the classification models above, it is also possible to use these models to predict whether a candidate will make a transition within a two-year period. The conversion is the following: if the number of transitions is more than 0 then the prediction is "transition"; otherwise, the prediction is "no transition." If the only thing of interest here is in predicting who will make a transition in the holdout period, then using mean squared error would be misleading because a large proportion of candidates in the training data did not make a transition in the hold out period. A baseline model of no transition yielded a low mean squared error even when it was assumed that no one will make a transition (29%). At the other extreme if a model that predicts everyone made a transition there is a risk of losing some candidates in the system because receiving unwanted recommendations is annoying for candidates who do not intend to switch to another job. Hence, alternative evaluation measures were chosen, which revealed the "true" performance of models. Since giving untimely recommendations is undesirable and the purpose is to detect candidates who will transition to new jobs, measures suitable for this purpose were selected. The two

evaluation measures suited for this are *sensitivity* and *specificity*. Sensitivity is the proportion of candidates who transitioned that were predicted successfully by a model. On the other hand, specificity is the proportion of candidates who did not change job as predicted by such a model. Ideally, we want to maximize both sensitivity and specificity. To compare the performance of the hierarchical Bayes models to the classification models, the AUCROC is also computed. Table 4.7 shows the computed sensitivity, specificity, and AUCROC for each model. Sensitivity and specificity are presented because in recruitment it is more profitable to detect a transitioning candidate (as will be illustrated later) than to detect a non-transitioning candidate. Of course, the only drawback is that the non-transitioning candidate might opt out from the system.

Table 4.7. Performance of hierarchical Bayes models on a binary outcome variable, that is, if a candidate will transition within a two-year period or not.

Model	Sensitivity	Specificity	AUCROC
Base model 1 (everyone will make a transition)	1.00	0.00	.5
Base model 2 (no one will make a transition)	0.00	1.00	.5
Pareto/NBD	0.731	0.507	.6192
Pareto/GGG	0.461	0.730	.5958
BG/NBD	0.731	0.507	.6192
NBD	0.731	0.507	.6192
Abe's Pareto/NBD (without covariates)	0.407	0.756	.5815

The best hierarchical Bayes models found so far are still the three models, namely, BG/NBD, NBD, and Pareto/NBD. Note that although recall is not 100%, we were still able to recover a sizeable number of candidates who made a transition and at the same we reduced the number of false positives by more than 50%. One explanation why the three models have the same and yielded the best performance is because all three assumed that the number of job transitions enacted by a candidate during the active work stage typically follows a Negative Binomial Distribution which implies that during the first few years (usually first 10 years) candidates enact numerous transitions albeit randomly and then after some time they tend to settle on a particular job (i.e., job transitions become less frequent). Hence, the Negative Binomial Distribution assumption holds true in practice at least for this specific CV data.

To make the impact of this more concrete in practice consider the following scenario. Suppose that for each successful hire a revenue of 3,000 euros is generated (Bika, 2019) and the cost is around .01 euro (*How Much Does Email Marketing Cost in 2019?*, 2019) for contacting candidate who will transition and 1 euro for candidate who does not intend to transition (this is just a conservative estimate, it

can be higher because of the cost associated with losing a candidate due to sending untimely recommendations). Given that an appropriate recommendation (we will discuss how in the next section), then the base of model of assuming everyone in our database will make a transition will generate a net loss 163.25 whereas the best model (the three models) will generate a revenue of 502.73. Now imagine this for thousands of vacancies and over a million candidates.

Predicting when to change job and the next job

The second and third objectives are to predict when to change job and what the next job will be for each candidate who has a high chance of transitioning to a new job.

First, the training performances of LSTM 1 for 100 epochs are presented in Figure 4.5(a) below using the mean squared error. The mean squared error on the test set is 54.07.

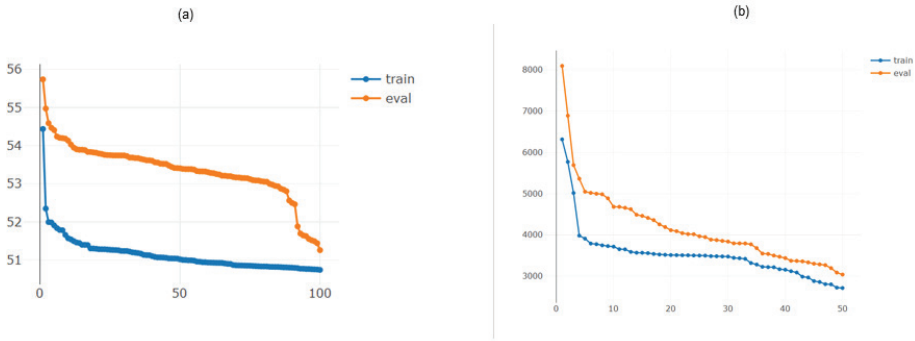


Figure 4.5. (a) Training and evaluation mean squared errors for 100 epochs and (b) Training and Validation Perplexity for 100 epochs
Note: One epoch is one pass of an entire dataset through the LSTM

Next the results of the analysis on predicting the next job are presented. The perplexity of the prediction on several rounds of RNN training is shown in Figure 4.5(b).

The performance of LSTM 2 was compared to a simple discrete time Markov chain (Kulkarni, 2016). For the comparison, the accuracy of the top 5 recommended jobs was used as a performance metric. A Markov Chain makes the simplifying assumption that the future depends only on the present and not the past (Markov property). The transition matrix was estimated via maximum likelihood method. Table 4.8 shows the result for LSTM 2, Markov chain and the base model which uses the latest job to predict the next job.

Table 4.8. Performance comparison of a Base model, Markov chain, and LSTM 2

Model	Top-5 accuracy
Base model (current job as prediction for the next job)	.267
Markov chain	.267
LSTM 2	.512

It is evident the LSTM 2 outperformed both the Markov and the base models. It is interesting to note that Markov chain and the base exhibited the same performance. This is not surprising since Markov chain like the base model infers the next job from the current job only.

Combining Hierarchical Bayes and LSTM 2 for Job Recommendation

The hierarchical Bayes was combined to the LSTM model so job suggestions are not only relevant but also timely. The two models were combined in a cascading manner. First, we determined whether someone would change jobs within two years and second, we aimed to offer a personalized recommendation. Accuracy was used to evaluate model performance. The prediction is correct when it correctly predicts that someone is going to change job in the holdout period and the next job is in one of the k-jobs we recommended. Of course, when a candidate does not intend to switch to a new job then there is no need to send him/her job vacancies. The computed accuracy on the test set is 70.10%.

Criterion validity using the Test data

The criterion validity of the models was demonstrated by using the test dataset. The test data here was never used during model training and validation and the candidates can be considered as "new" candidates. This independent data was set aside before model training and validation. The performances of the models are shown in Table 4.9.

Table 4.9. Performance on Test Set

(a)

Model	Recall	Specificity	AUCROC
Pareto/GGG	0.731	0.510	.6164
BG/NBD	0.731	0.510	.6164
NBD	0.731	0.510	.6164

(b)

Model	Top-5 accuracy
LSTM 2	0.625

The results are very similar to the ones we obtained previously. Hence, demonstrating the criterion validity of the models.

Discussion and Summary

The models explored in this paper solved four interrelated problems. First, predicting whether a candidate will change job within a two-year period using classification models. Second, predicting how many job changes a candidate will undergo (voluntary or involuntary) within a two year period (or any time duration for that matter) using Hierarchical Bayes. Third and fourth, when a candidate will change job and the most likely next job. Solving the first also gave insight as to which variables are significant for job mobility.

The models considered here yielded better performance compared to the naïve or non-ML approaches of just assuming that everyone will make a transition or that no one will make a transition. Aside from the predictive power, the classification insights about how candidates transition between jobs and about job relationships were elucidated. Variables associated with individual differences of career enactment are the most influential variables. Just by inspecting how long a candidate is in the labor market, his/her age, his/her previous interjob times, how long a candidate is in the current job, and the number of previous job changes, it is possible to predict whether he/she will make a transition within a 2-year period. It was also revealed that contrary to studies that assumed that the longer a candidate has stayed in the current job the likelihood of changing job increases (James et al., 2018; L. Li et al., 2017; Xu et al., 2015), results from this study showed that length of time in the current job is inversely propositional to the likelihood of changing job given that previous interjob times were also long. Moreover, education level was found to affect job mobility, specifically university level candidates appear to be more mobile than those who only finished secondary education.

Apart from individual level characteristics and patterns of mobility, industry differences and specific jobs also determine job mobility. As mentioned previously, once people reach managerial or administrative positions, they become less mobile. This is also true for highly technical jobs such as programmers and finance specialists. On the other hand, candidates who were sellers, or chefs and other food preparers were highly mobile.

The actual transitions of individuals from one job to another may be further investigated. For example, by inspecting transitions between jobs belonging to the same function, the mobility of candidates between these jobs can likely be better understood. Figure 4.6 shows the plot of most likely transition between jobs in function class "Informatie- en communicatietechnologie". As can be seen there seems to be relatively low mobility for database administrators and heads of automation confirming the findings in Segers et al., (2018) and the hypothesis of the vacancy

competition model that employees in some industry profiles have low mobility which is specifically true in the ICT industry nowadays.

Perhaps the reason for the low mobility for these two ICT jobs is that these jobs required specialized technical skills that are difficult to find replacements with, that employees in this industry usually accept a non-compete contract or that that the grass is simply not greener elsewhere. One implication is that these jobs are susceptible to worker shortages if demand for workers for these jobs continues to grow. Another observation is that there are a relatively high number of transitions to systeem en applicatiebeheerders (system and application manager), which is confirmed by the fact that this position is managerial in nature and requires seniority. Hence, with enough experience someone working as network administrator, system developer, or ICT consultant and specialist may in due time work as system and application manager.

It is also possible to inspect transitions between jobs from two function classes. Here we consider the function classes "Informatie- en communicatietechnologie" and "Onderwijs, opleiding en training". Figure 4.7 shows that most transitions happen within the same function class, but it is interesting to point out that there are also transitions from one function class to another. Specifically, most inter-function class transitions happen from "Onderwijs, opleiding en training" to "Informatie- en communicatietechnologie". Examples are from univ docenten and docenten hoger (university teachers) to supportmedewerkers and systeem en applicatiebeheerders respectively. On the other hand some candidates from "Informatie- en communicatietechnologie" were able to move to "Onderwijs, opleiding en training" such as programmeurs (programmers) becoming directeuren onderwijsinstelling (directors of an educational institution) and user interface designers switching to docenten basisonderwijs. The example here alludes to a potential remedy to ICT teacher shortage, that is, tap programmers and user interface designers to teach ICT subjects.

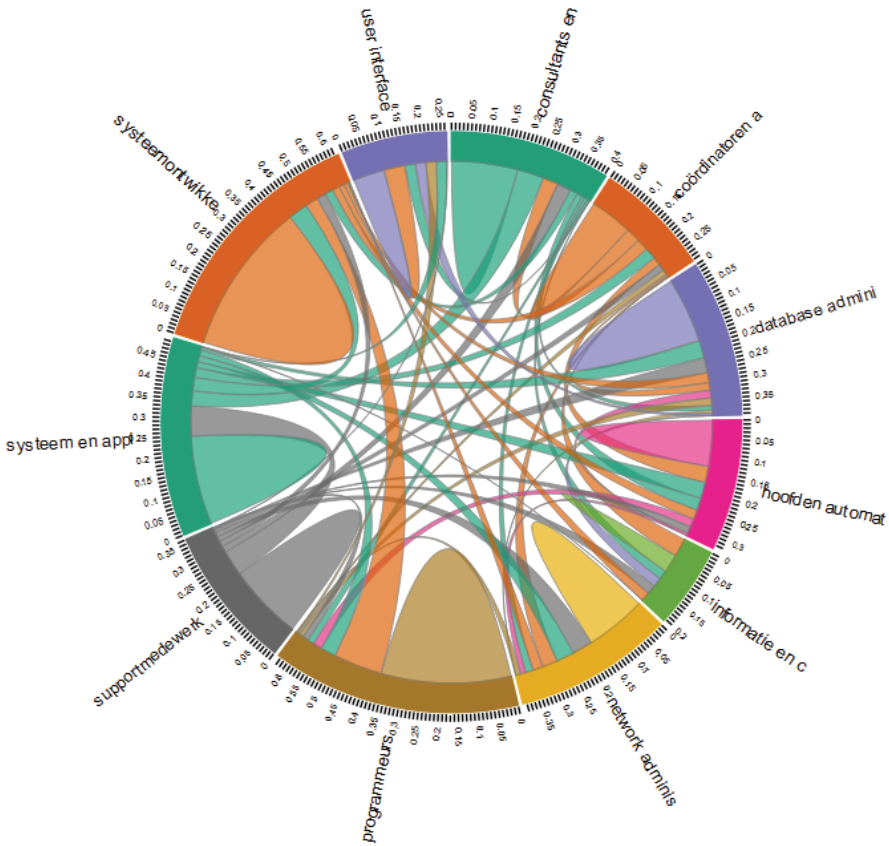


Figure 4.6. Transitions Among Jobs in the ICT Function Class.

Finally, the embedding from LSTM 2 also provides an overview of job relationships as shown in Figure 4.8. The figure shows the first and second embedding dimensions. It is possible to interpret the proximity of function as a degree of similarity. Hence at the bottom we see medewerkers kinderopvang, hulpen, and medisch assistenten positioned close to each other signifying similarity. This similarity may imply that these jobs have some overlap in terms of competencies required and responsibilities. The same can be said for administratie en klantenservice, managers schoonmaak en beheer, managers productie and commercial medewerkers en telemarketers. Note that these embeddings were obtained solely from the reported job transitions of candidates.

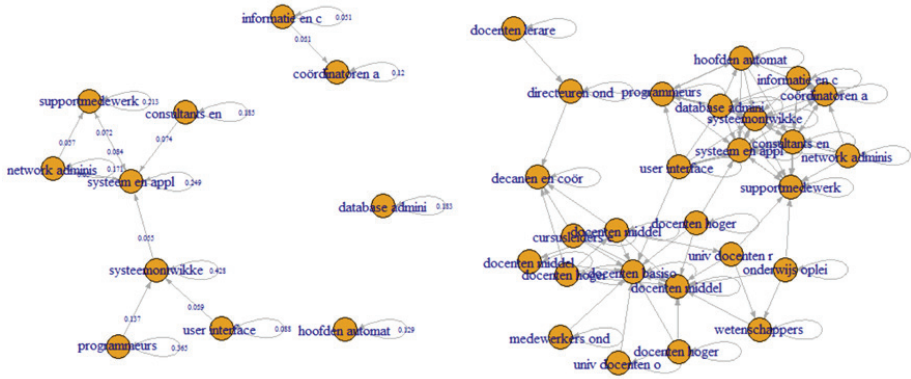


Figure 4.7. Transition Among Jobs from ICT to Education and Training and vice versa.

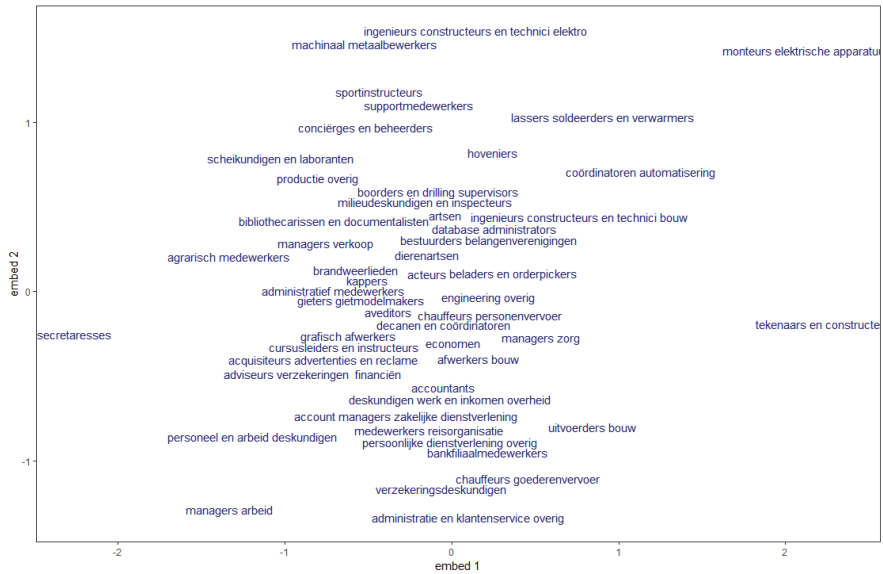


Figure 4.8. Function Class Proximities from the Embedding using the First Two Dimensions.

Contribution

The contribution of this study to the career literature is threefold. First, this study enhances our ability to predict job transitions. Second, the study introduces a novel means of job change analysis, and third, we propose a new view on job similarity. This study is one of the first that leverages the potential of resumes obtained from a large recruitment and HR company to analyze job changes on a massive scale (Xu et al., 2015). Our work contributes to the body of knowledge regarding job mobility as it affirms that patterns of job changes are indeed influenced by industry differences and maturation as construed by the various job mobility models. It is demonstrated that it is possible to infer job similarity and concomitant job clusters that correspond closely to existing job taxonomies, by analyzing employment histories using sequence-based text mining techniques. In contrast to studies that apply analytical techniques to study job transitions which stop at predictive performance, the emphasis here is put on insights that can be derived to test hypotheses and to understand job changes therewith better by contributing to theory development.

Besides contributions to the extant job mobility and career research literature this study also contributes to the job analysis literature because information about transitions provides information about job relationships, specifically because candidates tend to make transitions between similar jobs. This will open new avenues for potential transitions since finding suitable candidates for new or hard to fill jobs will be made possible by leveraging similarities and differences between jobs. This approach thus facilitates the identification of those jobs that constitute viable next career steps. The nature of experience and knowledge payoff from one job, possibly augmented with some training, will help candidates identify realistic career development opportunities.

On a more practical side, this study sets out to contribute to the improvement of job recommendations. By recommending vacancies to individuals who are about to change jobs, the scenario of candidates quitting from the system because of irrelevant and annoying recommendations (Gui et al., 2016), for example an associate professor in Organizational Behavior being recommended an associate professorship in quantum physics, can be avoided. Insights from this study will save time and effort and increase the likelihood of identifying qualified applicants. HR managers may also benefit from this study because its findings may help them determine the timing of promotion and plan the workforce better. Lastly, knowledge about timing and propensity of job changes would help organizations plan their workforce to avoid shortage of talents and perhaps mitigate potential employee turnovers.

Conclusion and Future work

In this paper, various methods for recommending jobs to candidates were compared using data that can be extracted from CVs, specifically, employment histories reported by job seekers. Four interrelated problems were solved: (1) predicting whether a candidate will change job, (2) predicting the number of job changes, (3) predicting when to change job and (4) predicting top-5 job options for the next job of candidates. These problems are also of primary interest to recruiters and HR specialists who are interested in hiring candidates.

This paper is also the first to apply hierarchical Bayes model to study job transitions. Studies often use survival models such as competing risk models to study this phenomenon (Hachen, 1988; Joseph et al., 2015). Moreover, we co-trained two LSTM models which to the best of our knowledge have not been done before. Some studies use the seq-to-seq LSTM models which claim to predict the entire career path of individuals (L. Li et al., 2017), this may be useful in some applications such as in exploring potential career progression, however, these models do not give the timing of job changes and are not dynamic and flexible enough to account for changing circumstances in the labor market and individual mobility patterns. The timing is of special interest as well to recruiters and HR specialists who want to determine when the best time is to offer a job opportunity

Further research can move in several directions. First, although the previous literature presents several approaches towards job recommendation using semantic matching, sequential pattern mining, and survival analysis, they lacked results on the similarities and differences between the generated recommendations of these approaches. Hence, further research should consider how these different approaches compare, and could be combined in a hybrid recommender system. Second, it would be interesting to extend the models and results of this study by incorporating other information present in the resumes such as KSAs of the candidate as this would give richer context and provide deeper level of differentiation for job change. Third, is to extend this work to candidates in other countries (if CV data is available). Although the employees in the data come from different nationalities, it would be interesting to investigate how the transitions may differ depending on the type of labor markets. Fourth, another aspect which may be interesting to consider is whether the job changes are inter or intra organization and whether the same factors would hold or whether another mix would likely be needed. Fifth and finally is to investigate the validity of this model during and after major global events have happened such as the COVID-19 pandemic and recent wars.

Moreover, the contribution of this paper extends beyond the confines of analyzing job changes as the results also have implications on job mobility and job

analysis because they offer evidence for alternative approaches and perspectives about jobs and how careers unfold. Typically, jobs are studied by collecting information about them such as work activities and work attributes (Brannick et al., 2007b; Sanchez & Levine, 2012b). The next step is to summarize the information collected and then to develop job profiles and ultimately a job taxonomy. Our proposal instead is to analyze job changes of individuals because these job changes capture the screening done by candidates when they assess their compatibility to the job which is possibly influenced by their experience or knowledge gained from previous jobs (Ferreira, 2016). It also reflects candidate preferences because they decided to accept the job. In traditional job analysis, unfortunately, job holder differences are seldom incorporated in the analysis. Inclusion of job holder differences pertaining to how they enact their career is informative because jobs should not be viewed as entirely distinct from the job holders since they influence and shape jobs (Berg et al., 2013). It was shown that whereas maturation and length of work life dictate whether to change job or not, previous job experience and industry differences on the other hand specify which transition is enacted.

In contrast with studies that either focus on methodology or theory (not both), this study seeks a middle ground by not only demonstrating the utility of machine learning in practical application such as in improving job recommendation but also trying to incorporate theory. The output from the models may be used for further investigation and may even become a source of hypothesis that will lead to further investigations. Finally, the models developed here can be integrated in automated IT systems for recruitment and talent management. The ability to learn from data and to deal with individual level information would certainly improve hiring chances and guide the planning of career paths for individuals.

Text mining in career studies: Generating insights from unstructured textual data

In this Chapter, we explain the process of text mining as it may be applied to the study of careers, starting with text preprocessing and ending with the application of analytical techniques. In the example, vacancy data are used to identify those knowledge, skills and abilities that (co)determine the salary associated with particular occupations. Here we address the following research questions:

RQ7 How to build a model that can predict salary from job descriptions in vacancies?

RQ8 What are the differences in pay structures among jobs and what drives these differences?

Text data pertaining to peoples' careers have proliferated in the past few decades. Due to the digitization of job search, recruitment, and the development of HR systems, it is relatively easy to access and obtain large datasets containing information about jobs or other work-related information at the micro (individual), meso (institutional), and macro (regional, national and global) levels, or some combination thereof. Examples of text data that may be used to study careers include (auto)biographies, resumés, posts in professional social networking sites, online job boards, public surveys, interview transcripts, personal diary entries, and even academic publications. Of particular interest are job vacancies, as aside from education and job experience, they also contain information about individuals' roles, responsibilities, knowledge, skills, and abilities, which comes with the promise of adding specificity and context to the career domain, which has come to be dominated by reductionist and generalist approaches to operationalizing key constructs. Online forums and social media also provide data relevant to the study of careers since employees use these platforms to voice their ongoing opinions and sentiments about their past and present employers.

As a way to characterize big text data we can use the framework of the four 'V's of Big Data: Volume, Velocity, Variety, and Veracity (De Mauro, Greco, & Grimaldi, 2015). The sheer *Volume* of the available text data on careers is unprecedented, and far beyond the traditional qualitative and quantitative datasets in careers research. It is oftentimes not possible to store these data locally on a single computer (e.g., a desktop) and to use traditional analytical software and methods for their analysis. Furthermore, the rate at which data about work and careers is generated (*Velocity*) is also growing. One should simply think about the number of public status or CV updates on popular professional/social networking sites such as LinkedIn or Facebook, or the number of vacancy announcements posted to the

Internet on a daily basis. Data also comes in many different guises (*Variety*), and are hardly ever produced with the primary aim of facilitating the conduct of research. Therefore, substantial effort must be invested to process different data types and forms in order to make the data suitable for analysis. A final challenge lies in the question of data and data-source integrity (*Veracity*), which also needs to be carefully considered when one wants to generate valid insights from textual data.

The abundance of 'big' text data containing information about careers also offers new avenues for careers research and paves the way for the development of bespoke text analysis methods and applications. Gone are the days when text analysis was limited to the mere counting of words, as contemporary sophisticated methods allow researchers to extract and organize textual content into topics, themes, and semantics, opening the door to theory generation and theory-testing (Kobayashi et al., 2018b). The current chapter was written to illuminate some of these possible avenues and to provide initial guidance to careers researchers who might be interested in working with 'big' text data.

Most of the aforementioned types of text data are relatively easy to collect, as they can be simply scraped from the Internet. However, in some cases researchers may need to consult with website owners, and ask for direct access (for instance through an Application Programming Interface or API). It comes without saying that there are privacy and possibly copyright issues associated with the use of such data. Privacy and copyright considerations should be negotiated prior to data collection and analysis (van Wel & Royakkers, 2004).

In an effort to familiarize careers researchers with text mining practices, this chapter provides an overview of the text mining process, from data pre-processing, the application of text mining operations, validation, to post-hoc analysis of the resulting models. Various text mining operations, such as dimensionality reduction, text clustering, topic modelling, text classification, and neural word embeddings are discussed. Finally, to demonstrate the capability of novel text mining methods in the area of careers research, an example use case is provided.

Text Mining

The primary aim of text mining (TM) is to generate (exploratory) insights from and/or test hypotheses using free unstructured big text (Kao & Poteet, 2007). Hence, text mining is viewed as an objective-driven, systematic process that is generally performed through the following three steps: (1) text data collection and preprocessing, (2) application of text mining techniques, and (3) postprocessing (Zhang, Chen, & Liu, 2015). Figure 5.1 shows an overview of the different steps in the TM process. The organization of this chapter follows this figure.

Text preprocessing may be further subdivided into *text data cleaning* and *text*

data transformation (e.g., converting unstructured text into intermediate forms which are used as input to the actual TM operations). TM operations refer to the application of algorithms with the goal of extracting hidden patterns and characteristics from text. Finally, postprocessing involves interpreting and validating knowledge obtained from TM operations. The focus of this chapter is on the different text mining operations. The readers are referred to other open source papers (Kobayashi et al., 2018b, 2018a) for a more comprehensive treatment of the other steps. Hereafter, all mentions of "data" refer to text data. Moreover, "document" and "text" are used interchangeably and corpus refers to the collection of text.

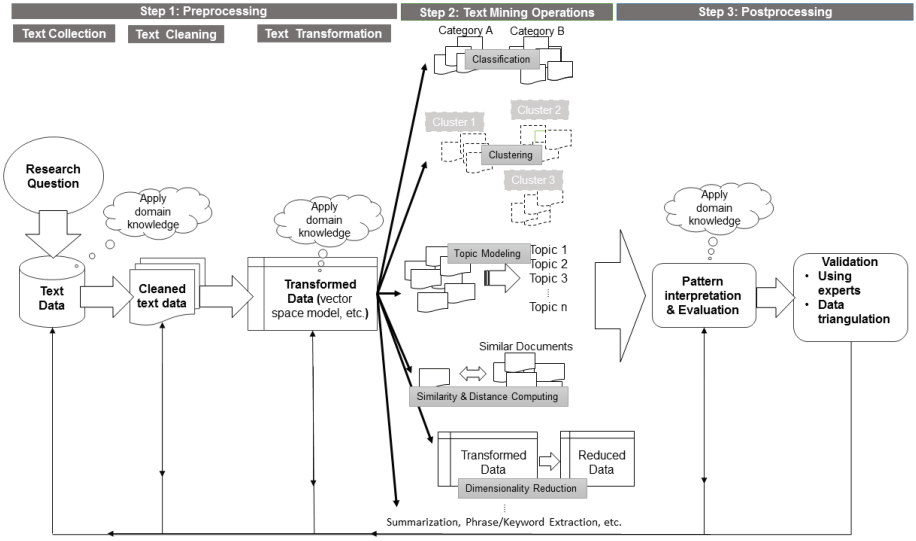


Figure 5.1. Flowchart of the Text Mining process
Note: Reprinted from Kobayashi et al. (2018b). Retrieved from https://doi.org/10.1177/1094428117722619. Licensed under a Creative Commons Attribution-NonCommercial 4.0 License.

Text Transformation: From Unstructured to Structured Data

Text mining techniques require that the input data be in a specific format. Before applying TM techniques, unstructured text data is first transformed so that TM techniques can be applied (Weiss et al., 2015)(Weiss et al., 2005). Unlike common approaches to analyzing quantitative data, in which data cleaning, data transformation, and data analysis are sequential and separate stages, the specific TM operation determines the text transformation because oftentimes a TM operation accepts only a particular text data format. In fact, when subsequent results are unsatisfactory, one can try different combinations of representations (Lewis, 1992; Scott & Matwin, 1999), different transformation methods, or TM operations. Usually

different combinations of data transformation and analytical techniques are tested and evaluated. In essence, text transformation is a representation strategy in which free text is converted into structured format (or formally mathematical objects). Most analytical techniques accept a matrix structure, where the columns are the variables (more commonly referred to as *features*) and the rows are the documents (for instance resumés, vacancies, or biographies). A straightforward approach is to construct this matrix by simply using the words (or *terms* as they are more commonly known) as variables. The resulting matrix is called a "document-by-term matrix" in which the entries of the matrix are raw frequencies of terms occurring in the documents. In many applications, this is an obvious choice since words are the basic linguistic units that express meaning. Thus, in this transformation, each document is transformed into a "vector", the size of which is equal to the size of the vocabulary (i.e., the set of unique words in the corpus), with each element representing the number of times a particular term occurs in that document (Scott & Matwin, 1999). By using raw frequencies, commonly occurring words in documents are given more weight.

Term frequency, in itself, may not be useful if the task is to make groupings or categories of documents (Kobayashi et al., 2018a). Consider the word "study" in a corpus consisting of abstracts of scientific articles. If the objective is to categorize the articles into topics or research themes then this term is not particularly informative in this particular context since many abstracts will contain this word. A way to prevent the inclusion of terms that possess little discriminatory power is to assign weights to each term that reflect its specificity to particular documents in a corpus (Lan et al., 2009). The most commonly used weighting procedure is the Inverse Document Frequency" *(IDF)* (Salton & Buckley, 1988). It is computed using the following formula:

$$\mathrm{IDF}(\text{term i}) = \log\frac{N}{n}$$

where *N* is the corpus size and n is the number of documents containing term *i.*

A term with an IDF of zero is useless in the discrimination process because that term is present in every document (i.e., *N=n)*. In fact, *IDF* can also be used as a criterion to filter out common terms, that is, terms that have a low *IDF* have little discriminatory power and hence can be disregarded. When the raw term frequency (*TF*) and *IDF* are multiplied together it yields the popular *TF-IDF,* which in principle simultaneously accounts for both the word's frequency and specificity (Frakes & Baeza-Yates, 1992) with higher values being more desirable.

One disadvantage of representing texts as a document-by-term matrix, is that it ignores word order information. This can be especially problematic when the goal is to extract semantics in text (Harish, Guru, & Manjunath, 2010). However, it

turns out that despite ignoring word order information, this representation seems to work well for many text classification applications, such as in email spam detection, document authorship identification, and topic classification of news articles (Song, Liu, & Yang, 2005; Zhang, Yoshida, & Tang, 2008). Another disadvantage of the document-by-term matrix representation is the resulting high dimensionality, that is, size of the vocabulary. One can use different data dimensionality reduction methods to reduce the number of variables (e.g., variable selection and variable projection techniques) or employ specific techniques suited for data with high dimensionality. These techniques will be discussed in the Text Mining Operations section.

There are other ways to represent text such as using the n-gram approach which uses n consecutive words or letters (including spaces) as features. Another is one-hot encoding which represents each word as a vector consisting of 1 in the position of the word and 0 in the other positions. The vector has the same size as the vocabulary. For example, suppose that the vocabulary only has these words {"the", "covid-19", "pandemic", "killed", "thousands", "of", "people"} then the vector representation for "pandemic" is (0,0,1,0,0,0,0). This type of representation stands at the basis of neural word embedding (more on this later).

Once text is transformed, techniques such as classification or cluster analyses can be applied. Concatenating noncontiguous words can also tackle substantive questions about the text. For instance, in resumés of job applicants the proximity of the words "experience" and "year" together with a number are used to deduce applicants' length of work experience.

Text Mining Operations

In the following sections, those TM operations which we find most applicable to careers research are discussed. This is followed by a discussion of methods that may be used to assess the credibility and validity of TM outcomes. Most of the techniques covered here accept the document-by-term matrix as input, otherwise we shall explicitly mention the format of the input data.

Dimensionality Reduction. Document-by-term matrices tend to have many variables and subsequent analyses may suffer from what is called the *curse of dimensionality* (Aggarwal & Zhai, 2012; Alpaydin, 2014). It is usually desirable to reduce the size of these matrices by applying dimensionality reduction techniques. Benefits of reducing dimensionality include more tractable analysis, greater interpretability of results (e.g., it is easier to interpret variable relationships when there are few of them), and more efficient representation. Compared to working with the initial document-by-term matrices, dimensionality reduction may also reveal latent dimensions (e.g., higher level concepts) (Yarkoni, 2010) and may result in improved performance (Bingham & Mannila, 2001).

Two general approaches are commonly used to reduce dimensionality. One

is to construct new latent variables and the second is to eliminate seemingly irrelevant variables. New variables are modelled as a (non)linear combination of the original variables and may be interpreted as latent constructs. For example, the words "flexible", "willingness", and "abroad" may be merged to express the concept of a "willingness to travel" in a corpus of job vacancies. An added advantage of dimensionality reduction techniques is that they may be used to eliminate variable multicollinearity.

Singular Value Decomposition (SVD) is a classic tool which underlies techniques such as Latent Semantic Analysis (Landauer et al., 1998b)(Landauer, Foltz, & Laham, 1998) and Principal Components Analysis (PCA) (Jolliffe, 2005) (Jolliffe, 2005). Reducing the number of dimensions is accomplished by retaining only the first few largest singular values. Usually, this implies choosing latent dimensions and recovering the right dimensionality of the data because at times, true dimensionality is obscured by random noise.

Latent Semantic Analysis (LSA) is commonly used when synonymy (i.e., different words that have the same meaning) and polysemy (i.e., one word used in different yet related senses) are present in the data. PCA is effective for data reduction as it preserves the variance in the data. Parallel analysis (Ford, MacCallum, & Tait, 1986; Hayton, Allen, & Scarpello, 2004; Montanelli Jr & Humphreys, 1976) is the recommended strategy to choose how many dimensions to retain in PCA. A disadvantage of both LSA and PCA is that it may be difficult to interpret the derived dimensions. Another technique is Random Projection (RP) where data points are projected to a lower dimension while maintaining the distances among points (Vempala, 2005). Compared to PCA, RP is computationally less demanding and its results are comparable to those of PCA (Bingham & Mannila, 2001).

An alternative approach to reduce dimensionality is to eliminate variables by using variable selection methods (Guyon & Elisseeff, 2003). In contrast to projection methods, variable selection methods do not create new variables but rather select from the existing variables by eliminating those that are uninformative or redundant (e.g., words that occur in too many documents such as "the", "to", "and", "of"). Three types of methods are available: filters, wrappers, and embedded methods. Filters assign scores to variables and apply a cut-off score in order to select relevant variables. Popular filters are TF-IDF thresholding, Information Gain, and the Chi-squared statistic (Forman, 2003; Yang & Pedersen, 1997). Wrappers select the best subset of variables depending on the particular analytical method that is to be applied. Searching for the best subset of variables using embedded methods is accomplished by minimizing an objective function that simultaneously takes into account model performance and complexity. Model performance can be measured, for example, by prediction error (in the case of classification), and complexity is operationalized in terms of the number of variables in the model. In line with Ockham's razor, the preferred subset is the one that achieves the best balance between the number of variables (fewer is better) and prediction error (lower is better). In practice, the model

prediction error is computed using a separate test set. That is, the model is developed on training data and validated on a separate sample of testing data.

Text Clustering. Many tasks in TM involve grouping documents such that documents belonging to the same group are similar and documents from different groups are dissimilar (Jain, Murty, & Flynn, 1999; Steinbach, Karypis, & Kumar, 2000). The process of grouping is called *clustering*. The main uses of text clustering are to organize documents to facilitate efficient search and retrieval and to impose an automatic categorization of documents. For example, text clustering has been used to create topical groupings in a collection of legal documents (Conrad et al., 2005) and automatic grouping of search query results (Osinski & Weiss, 2005). In many clustering procedures the researcher needs to define a measure of distance between texts. Commonly used measures that operate on vector representations are the Euclidean and Hamming distances.

Most clustering algorithms are categorized as either hierarchical or partitional (Steinbach et al., 2000). Hierarchical clustering algorithms are classified into either agglomerative or divisive. In agglomerative clustering, initially there are as many clusters as there are documents and then gradually clusters are merged until all objects belong to a single cluster. Conversely, the divisive approach entails first assigning all documents to a single big cluster and recursively splitting clusters until each document is in its own cluster. The merging (or splitting) of clusters is oftentimes depicted using a tree or dendrogram.

For partitional clustering the user must specify the number of clusters beforehand and clusters are formed by optimizing an objective function that is usually based on the distances of the objects to the centers of the clusters to which they have been assigned. The popular k-means algorithm is an example of partitional clustering (Derpanis, 2006). One key challenge in clustering is the determination of how many clusters to form. Since clustering is an exploratory and inductive technique, a common strategy is to try out different numbers of clusters (k) and use cluster evaluation metrics (e.g., Dunn index, Silhouette coefficient, or an external evaluation criterion) to decide upon the most suitable number of clusters (Jain et al., 1999).

Topic Modeling. Topic modelling can be applied to automatically extract topics from documents, where extracted topics represent latent constructs or themes. For example, in a corpus of exit interviews, one could set out to extract the various reasons that people have mentioned for quitting their current job. In machine learning and natural language processing, topic models are probabilistic models that are used to discover topics by examining the pattern of term frequencies (Blei, Ng, & Jordan, 2003). Its mathematical formulation has two premises: a topic is characterized by a distribution of terms and each document contains a mixture of different topics. The most likely topic of a document is therefore determined by its terms. For example,

when an exit interview contains words such as "pay", "compensation", "salary", and "incentive", one of its candidate topics is "rewards or compensation".

Perhaps the most popular topic models are the Latent Dirichlet Allocation (LDA) (Blei et al., 2003) model and the Correlated Topic Model (CTM) (Blei & Lafferty, 2007). LDA and CTM both operate on the document-by-term matrix (Porteous et al., 2008). CTM will yield almost the same topics as LDA. The main difference between the two is that in LDA, topics are assumed to be uncorrelated (i.e., orthogonal), whereas in CTM topics can be correlated. In comparing LSA with LDA, the latter has been found to be particularly suitable for documents containing multiple topics (Lee et al., 2010). Supervised Topic Modeling is an extension of LDA (Mcauliffe & Blei, 2008). The modeling assumptions remain the same as in LDA except that it is possible to incorporate an outcome variable. For example, one could use Supervised LDA to extract those career shocks from interview transcripts that are related to a quantitative measure of career sustainability.

Classification. Classification is the assignment of objects to predefined classes or categories, which unlike clustering are known a priori. Logistic regression is perhaps the best-known classification method. The goal is to construct a model that can predict the category of a given document. Example applications of text classification are spam or ham classification of emails (Youn & McLeod, 2007) (Youn & McLeod, 2007), authorship identification (Houvardas & Stamatatos, 2006)(Houvardas & Stamatatos, 2006), thematic categorization (Phan, Nguyen, & Horiguchi, 2008), and identification of sentiments in product reviews (Dave et al., 2003a; M. Hu & Liu, 2004b; Pang & Lee, 2008; Popescu & Etzioni, 2007)(Dave, Lawrence, & Pennock, 2003; Hu & Liu, 2004; Pang & Lee, 2008; Popescu & Etzioni, 2007). In the career domain, one could consider using written performance appraisals to predict dichotomously coded promotion decisions for a particular time period. For a fuller discussion and tutorial on text classification we refer the reader to (Kobayashi et al., 2018a).

Neural Word Embeddings. Word embedding involves the mapping of a vocabulary of words to vectors of real numbers, this with the purpose of looking at the similarity and dissimilarity of the word vectors. Similarity and/or dissimilarity between vector representations can help gain intuition in the relation between words, for example "What are the top 5 most similar words to career satisfaction." Furthermore, vector representations allow us to perform certain arithmetic operations with an equal linguistic meaning, for example "Job" – "Pay" might result in a vector representation closest to "Voluntary work" (note that this is but an example).

A central paradigm for deriving these word vector representations is the distributional assumption on which it is based. First proposed by Harris in 1970, the distributional assumption states that words in similar contexts have similar meaning.

For example, we could derive the meaning of the knowledge, skill, and abilities (KSAs) from certain occupations by looking at the context in which the KSAs occur. Thus, occupations related to "Computer Science", will then be describing a context containing KSAs such as; "data structures and algorithms," "optimization," "discrete mathematics," and so on. However, in order to fully capture the semantics of, for example, "Computer Science", domain-specific documents fully capturing the terminology describing the phenomenon "Computer Science" has to be eminent.

There are a variety of different methods to derive the word representations. Levy, Goldberg, and Dagan (2015)(2015) compared matrix-based algorithms, such as positive pointwise mutual information (PPMI) and singular value decomposition (SVD) to neural methods such as skip-gram with negative sampling (SGNS) and GloVe. Results found mostly local or insignificant performance differences between the methods, with no global advantage to any single approach over the others (Levy et al., 2015)(Levy et al., 2015). For a more comprehensive overview and practical recommendations towards the implementation of the right method we refer the reader to Levy et al. (2015)(2015).

Validation and postprocessing

The postprocessing step may involve domain experts to assist in determining how the output of the models can be used to improve existing processes, theory, and/or frameworks. Two major issues are usually addressed here. The first is to find out whether the extracted patterns are real and not just random occurrences due to the sheer size of the data (e.g., by applying Bonferroni's principle). The second is, as with all empirical research, whether data and results are valid. Establishing validity (e.g., content, construct, internal, and external validity), and (therewith) the credibility of the output of TM models is particularly important for TM to gain legitimacy in careers research. It is important to note here that it is not the TM procedures that need to be validated but the output (in the same manner that we do not validate Factor Analysis), for example, the predictions of a TM based classifier.

Prior to being applied to support decision making and knowledge generation, the validity of TM based findings will need to be established. When TM is used to identify and operationalize key careers constructs, using different forms of data triangulation will help generate content and construct validity evidence. For example, in our job analysis example of TM application, which follows below, we enlisted the help of job analysts and subject matter experts in evaluating the output of the TM of vacancy texts. In other cases, TM outcomes could be compared to survey data, as was the case in the study on the role of personality in language use (Yarkoni, 2010). More generally, TM based models will require a comparative evaluation in which (part of) the TM output is correlated with independent or external data sources or other 'standards' (such as the aforementioned survey or expert data). Though it is easy to view TM as a mechanistic means of extracting information from data, the input of

domain/subject matter experts is critically important.

A straightforward practice for content validation is to have independent experts validate TM output. For example, in text classification, subject matter experts (SMEs) may be consulted from time to time to assess whether the resulting classifications of text are correct or not. A high agreement between the experts and the model provides an indication of the content-related validity of the model. The agreement is usually quantified using measures such as the Cohen's kappa or intra-class correlation coefficient.

Another way to validate TM output is through replication, data triangulation, and/or through an indirect inferential routing (Binning & Barrett, 1989b)(Binning & Barrett, 1989). The standard can be established by obtaining external data using accepted measures or instruments that may provide theory based operationalizations that should or should not be correlated to the model. Such correlations give an indication of construct validity. For example, to validate experience requirements extracted from job vacancies, one can administer questionnaires to job incumbents asking them about their experience. Validity is then ascertained through the correlation between both operationalizations. This can be replicated on various types of text to assess if the TM model consistently generates valid experience requirements for a particular occupation. In theory, one could even compute full multi-trait multi-method correlation matrices (D. T. Campbell & Fiske, 1959b) (Campbell & Fiske, 1959) to compare the measurements obtained from TM with established instruments, although in practice it may be difficult to obtain the fully crossed dataset that it requires.

As with the statistical analyses that are (more) commonly applied in career research, text mining procedures in and of themselves cannot support causal inference (i.e., internal validity) unless the study design is such that, next to association, temporal precedence and isolation are also established. Given that many text mining applications rely on "data exhaust" (i.e., data that were not purposively collected for investigating the research questions at hand) and between-subjects designs, this is often difficult if not impossible to achieve. Despite these constraints, inferences pertaining to internal validity may be strengthened by collecting multiple waves of data and seeking to establish that changes in (a set of) TM based independent variable(s) are predictably related to changes in some hypothesized dependent variable(s) over time. Furthermore, although the ideal of randomized allocation of subjects to (pseudo-)experimental conditions may not always be viable, the sheer size of the data may likely yield sufficient statistical power to include a large number of theory-based (Bernerth & Aguinis, 2016)(Berneth & Aguinis, 2016) control variables and/or to leverage a propensity score matching approach. Finally, as to external validity researchers need to be cognizant of the fact that although a given textual dataset may be qualified as being 'big' it may still represent a nonrandomly selected sample from a (much) larger population to which one wishes to generalize.

The fact that Amazon ceased its resumé screening program, could be argued to have resulted from the fact that the (what turned out to be a gender biased) algorithm, was trained on an unrepresentative sample of data that was dominated by successful males. In our own work on vacancy mining we have also struggled with defining what our population is comprised of in the first place. Should the unit of analysis be vacancy, job, job candidate, or task? Clearly, whatever is decided upon has an important bearing on inferences pertaining to external validity. In sum, it is critical that the evaluation metrics developed and accepted within the data science tradition, must be augmented with the aforementioned validity types so as to facilitate adequate and accurate knowledge generation and decision making.

Text mining Example: Salary prediction from job vacancies

The purpose of this section is to illustrate how text mining may be applied in practice. The example uses job vacancies as a data source. The objective is to create a model that can be used to predict the salary associated with a particular occupation from job descriptions in job vacancies. The results could be used to shed light on the pay dynamics across jobs and may answer the question of what makes a particular job pay more as compared with other jobs. Although there are many factors that affect pay (e.g., discrimination, geographic immobility, supply and demand of labor, etc.), here we examined how text mining could identify those antecedents that derive from the nature of the job. Also, based on the results we briefly discussed the issue of how skill requirements influence pay.

Salary prediction from job vacancies

Every day, thousands of vacancies are posted to job boards and employment websites across the world. These vacancies are rapidly becoming a valuable source of job information. Most vacancies contain both worker-oriented (e.g., abilities, skills, knowledge, etc.) and job-oriented (e.g., work activities) information (cf. Peterson et al., 2001)(2001). Job analysts and labor organizations are looking for ways on how to use this abundant source of data to answer questions about jobs including evolution of skill demand, emergence of new jobs, and job task analysis. From the career research perspective, TM of vacancies can be to support career counseling and educational choices of students (Messum et al., 2017) and for quantifying the readiness of employees with respect to new work paradigms (Fareri et al., 2020).

A crucial step in using these vacancies is to be able to analyze their textual contents. For example, jobs may be grouped according to skill requirements or activities performed in the job. Also, job categories may reflect differences in salary. Classifying vacancies would give us a better understanding of the job demand in each industry and may form the basis of further analysis such as the investigation of which skills are better compensated in the labor market.

Although there are already existing taxonomies of jobs and their associated

pay systems, salary differences are still present even within same job category and likely to change over time. Some researchers ascribe these differences to gender and/or salary negotiation skill (Voigt & Ruppert, 2019)(Säve-Söderbergh, 2019; Voigt & Ruppert, 2019). While other researchers point out other factors including artificial barriers (e.g., union and government restraints) and investment in human capital. In our small example, we will try to elucidate such salary differences just by analyzing job vacancies. Guided by the expansive literature on wage differentials, we attempt to identify some underlying factors influencing salary differences solely from analyzing vacancies. This approach applies a TM technique, specifically supervised LDA, to automatically extract topics from these vacancies and use these topics to create a predictive model of salary. The approach here is exploratory in nature but can be used as a basis for subsequent hypothesis testing. The following steps, which are explained in greater detail below, were followed: (1) Preprocess and transform the textual content of job vacancies, (2) extract topical patterns and build the model for salary prediction, and lastly, evaluate the performance of the resulting model. This evaluation, will be done by running the model on test data (i.e., data not used in building the model). Once we have determined that the model yields a reliable (consistent) prediction of salary, we can then develop strategies to establish evidence for content, construct, internal, and external validity, along the ways that were outlined earlier, although clearly some outcome and design would need to be defined for the establishment of internal validity. Although this process is unwieldy, and perhaps even beyond the scope of a single empirical study, ultimately, the insights we derive from examining how predictions are obtained and how they relate to other constructs may improve our understanding regarding pay differences in jobs and may form the basis for succeeding analyses.

Data

The data consisted of 50,000 job vacancies posted to various job boards in the United Kingdom. The automatic prediction of salary from job vacancy descriptions is expected to shed light on how salary differs across job groups. Furthermore, by analyzing the content of vacancies it should be possible to provide career options on jobs that pay better and a more detailed quantification of the skill–salary relationship, that is, how each skill is valued in the labor market.

Preprocessing

The first step in the process is text preprocessing. First, we extracted relevant content from text. Each vacancy is in HTML format, hence, HTML and other formatting tags (e.g., tabs, new line, and long white spaces) were removed. Moreover, numbers, punctuation marks and stop words (Dolamic & Savoy, 2010; Fox, 1989) for the English language were deleted. Upper case letters were converted to lower case and finally all terms containing only two characters were removed. Extra whitespace

and whitespaces at the beginning and end of the description were trimmed out. The extra whitespaces were the result of removing characters. An example job description and the resulting text after applying the preceding text cleaning procedures is shown in Table 5.1.

After text preprocessing, each text is then transformed into a vector resulting in a document-by-term matrix (DTM) representing the entire corpus. The columns of the DTM are the distinct terms occurring in the corpus and the entries are raw term frequencies. Since we want to run a supervised topic model, we need an outcome variable, in this case the outcome variable is annual salary, mentioned in the vacancies. For this, we developed a parser that automatically extracts the salaries. There are few nuances that we needed to deal with in order to extract the salaries, one is some salaries are provided on an hourly and monthly rate. Another is some vacancies provide a range rather than a single figure for the salary. To harmonize the different salary rates, we converted all hourly and monthly salaries to annual salaries. For the hourly salary, we multiplied the hourly rate by 1538 which is the average hours per year that a full-time employee in the UK will spend working. In the same manner, we multiplied reported monthly salary by 12.

Table 5.1. An example job vacancy text prior to and after text preprocessing

Original text	**Resulting cleaned text**
Apply now Castles Solicitors are looking for a part / full time Legal Secretary / Admin Assistant to join their team in Hurstpierpoint, working across multiple departments (flexible working hours available ideally 20+ hours a week) Essential skills: Microsoft and excel competentAbility to communicate with clients via email and telephoneOrganisation / good time managementAbility to work independently and as part of a teamGCSE Maths and English C or above Desirable but not essential as full training will be given: Legal background / qualificationsPrevious office experienceExperience with case management systems Role: Communicating with clients and creating appointments / general diary managementOpening and closing fileGeneral correspondence with clients and third partiesDrafting legal documentsTyping from dictationGeneral admin roles	apply now castle solicitor look part full time legal secretary admin assistant join team hurstpierpoint work across multiple department flexible work hour available ideally hour week essential skill microsoft excel competent ability communicate client via email telephone organisation good time management ability work independently part team gcse math english desirable essential full train will give legal background qualification previous office experience experience case management system role communicate client create appointment general diary management open close file general correspondence client 3 party draft legal document type dictation general admin role

For salary ranges, we computed mean salary or the middle of the range as an estimate of the salary and multiplied by an appropriate number to convert them to annual salaries. We put all extracted and calculated salaries in a column vector. Finally, we merged the column containing the salary of each job with the DTM.

Supervised LDA

For the supervised LDA part, we extracted one hundred topics. The choice of 100 topics is default in many other studies (Airoldi & Bischof, 2016; Wallach et al., 2009). However, the choice for the number of topics to retain can be evidenced by examining the prediction accuracy of models with varying numbers of topics. Since the purpose here is to illustrate rather than to optimize accuracy, the choice for 100 topics suffices in this case. The outcome of our supervised LDA here is a predictive model that is used for predicting salary (similar to a linear regression model). Aside from predicting salary, we can also examine the topics that were created. For purposes of illustration, we show nine topics in Table 5.2. The rest of the topics can be obtained upon request from the first author. In principal components analysis (PCA) in quantitative research, factor loadings are examined to interpret PC dimensions, in topic modeling, a topic can be interpreted by inspecting the top words, that is words that have the highest probabilities of belonging to the topic. The content of the vacancies is summarized by the topics such as shown in Table 5.2. Some topics are indicative of the job requirements written in vacancies. Finally, Table 5.3 shows the salary ranges and sample job titles associated to Topics 3, 9, 32, 33, and 72.

Table 5.2. Sample topics constructed using supervised topic model

Topic 8	**Topic 19**	**Topic 20**
food	member	software
chef	ability	developer
restaurant	relevant	technology
kitchen	communicate	test
hotel	effectively	web
hospitality	communication	java
guest	check	net
cater	write	sql
head	responsibility	script
fresh	clear	end
T**opic 25**	**Topic 27**	**Topic 29**
digital	maintenance	safety
medium	machine	health
campaign	installation	energy
social	system	requirement
content	carry	responsibility
brand	equipment	legislation
channel	tool	environmental
agency	gas	assessment
online	repair	current
strategy	test	qualification
Topic 40	**Topic 61**	**Topic 86**
datum	care	engineer
analyst	people	manufacture
analysis	worker	maintenance
report	social	production
insight	child	mechanical
analyse	young	electrical
model	health	equipment
analytic	home	technical
analytics	community	plant
use	life	electronic

Table 5.3. Job titles, salary ranges associated to topics 3,9 32, and 76

Topics	Terms	Average salary	Top job titles
3	organisation partner strategy develop key stakeholder strategic plan influence relationship	50,000–115,000	"Head of Communications and Engagement," "Head of Internal Communications & Engagement," "Head of Partnerships and Performance," "Stakeholder Engagement Strategy Manager Commercial Excellence," "Chief Product Officer"
9	risk bank programme function operational investment asset knowledge financial senior	90,000–132,000	"Operational Risk Manager," "Senior Investment Risk Manager," "Financial Intelligence Specialist," "Treasury Analyst, Liquidity Management, ALM, Banking," "Brexit Risk Manager – Insurance Delegated Authority Background," "Chief Operation Officer, FX trading"
32	clean area facility duty use keep general part equipment site	16,800–26,000	"School Cleaner," "Housekeeping Assistant/Laundry Assistant," "Cleaning Operatives/Cleaners Night – Gatwick Airport," "Refuse/Recycling Loader," "Aircraft Cleaning Operatives-Heathrow Airport," "Night Cleaning Manager"
33	detail ability attention communication deadline pressure write organisational verbal able	25000-50000	"Litigation Support Specialist – Northampton," "Trade and Transaction Reporting Officer," "Recruitment Resourcer – Admin," "Sales Administrator," "Marketing CRM Assistant," "Desktop Publishing Specialist," "IT Project Support Analyst," "Administrator," "Temporary Sales Administrator," "Sales Order Processor," "Immigration Solicitor," "Costs Draftsman"
76	call centre free park yorkshire advisor contact inbound outbound position	22,500–78,000	"Call Centre Advisor," "Your 1st Call Centre Customer Service Advisor Job," "Receptionist," "Assistant Buyer/Purchasing Assistant," "Retail/Shop Supervisor," "Digital Marketing Executive," "Customer Services Adviser – Investment and Financial Services," "Telesales Executive," "Collections Advisor"

Predictive performance is measured using the mean squared errors which is computed by the following formula:

$$mean\ squared\ error = \frac{1}{n}\sum_{i=1}^{n}(observed_i - predicted_i)^2$$

where n is the size of the corpus. The performance was evaluated using 10-fold cross validation.

Results

The 10-fold cross validation yielded an average mean squared error of 0.16 which denotes good performance. For the postprocessing, and to establish construct validity, the model is applied to predict the salary in the vacancies in the test data and it yielded a mean squared error of 0.23. We also used the output of the model to examine which topics are associated with high and low salaries. Figure 5.2 shows the highest five topics and lowest five topics as they relate to salary. The figure shows that jobs about strategy planning, banking jobs and jobs located in big cities (e.g., London and Birmingham) tend to pay more than cleaning and call center jobs. Figure 5.3 shows the ordering of skills and knowledge as they relate to salaries. As can be seen, knowledge on finance, software development and data analysis command high salaries in the labor market. Moreover, organizing, leadership, and problem-solving skills are also highly valued. Other employee characteristics that also appear to incur higher pay are being motivated and ambitious. It is surprising to see that communication skills do not figure in the list of high paying skills. One explanation is this skill is commonly required and that it is usually stated in low paying jobs such as in call centers or administrative support. We investigated further by analyzing which job categories tend to pay higher, Figure 5.4(a) shows a scatter plot of jobs according to average salary and availability. It is obvious that some high paying jobs are not necessarily in high demand, this can be explained by the fact that these jobs require higher ability and more experience (more investment in human capital). Another explanation is that the productivity of workers in these job categories contribute more on the revenue of the companies (banking jobs, strategy and consultancy jobs). Two job categories seem to offer very good opportunity/outlook because of their high demand and pay, these are IT and accountancy qualified jobs. Professional jobs lie somewhere above the median salary and jobs in retail, admin, social care, customer service or related to leisure are the least paying jobs, with the exception of apprenticeship jobs which usually are unpaid jobs. We compared our results with the data from National Statistics Office of UK (Smith, 2019) (Figure 5.4(b)) and we find similarity in terms of the ranking of job categories with respect to salary. The similarity provides evidence of the content validity of the information extracted from job vacancies.

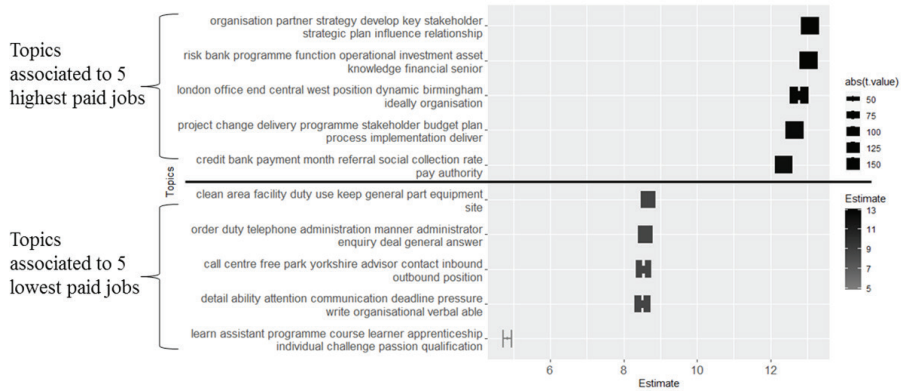


Figure 5.2. Top five topics associated to higher pay (first five) and topics associated to least pay (last five)

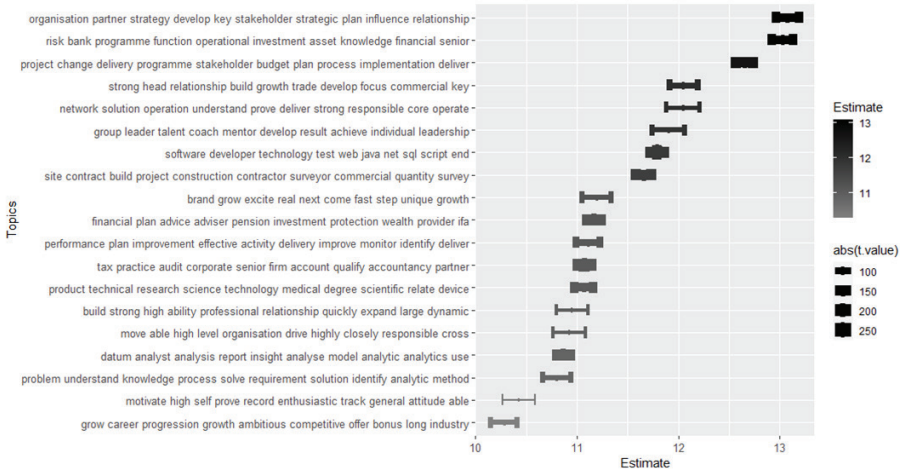
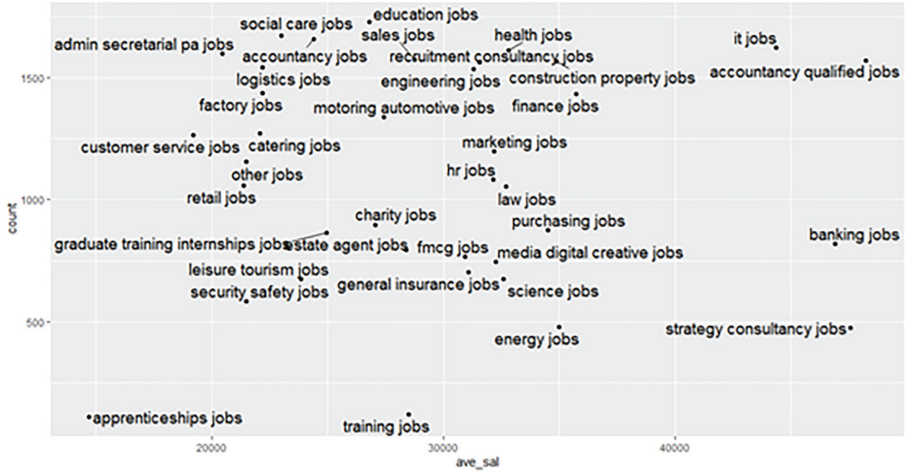


Figure 5.3. Knowledge and skills associated with higher pay. The x-axis represents the coefficients of the predictive model; the higher coefficients the higher the contribution to salary. The error bars represent the standard errors of the coefficients.

(a)



(b)

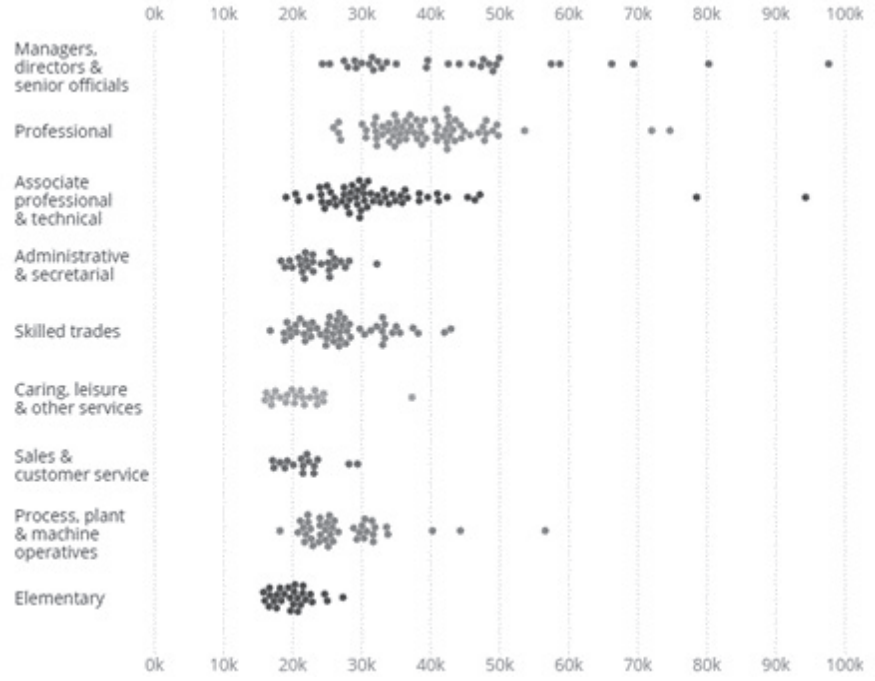


Figure 5.4. (a) Job categories according to salary and availability and (b) Plot of salaries according to job categories
Note: Reprinted from Smith (2019). Public sector information licensed under the Open Government Licence v3.0.

This example showed the feasibility of developing a prediction model for the automatic prediction of salary from text. The resulting classifier is scalable as it can predict salaries on thousands of vacancies which would be laborious and time consuming when done manually. Moreover, the model also sheds light on what determines salaries in jobs which can be difficult to determine especially in the case of new jobs.

Other potential applications

In the preceding section we applied a text mining technique (e.g., supervised topic modeling) on a problem not only of potential interest to career researchers but also to labor economists, and education researchers. Here we explore other potential avenues for future research at the interface between text mining and careers research by suggestion some ideas how existing problems in career research can be reconceptualized as a text mining problem. Our recommendations are purely speculative at the moment and are aimed to excite and provide inspiration to career researchers to incorporate text in their analysis as well as to encourage them to collaborate with text mining experts.

Studies that attempt to analyze how individual attributes relate to career outcomes may benefit from text mining. Specifically, TM may provide highly scalable and unobtrusive means of assessing those psychological constructs that careers researchers often hypothesize to predict career outcomes. Not only would this approach prevent burdening respondents with lengthy surveys, it also has the potential to generate data for much larger samples than we are typically accustomed to. One may build a simple score-based system that measures how each word is related to a certain a career concept or dimension using frequency analysis or word-embedding approaches (Shen, Brdiczka, & Liu, 2013; Yarkoni, 2010). Indeed, results from TM can be compared to Likert-scale questionnaires and when the results are properly validated, they may even be used in situations where it is not feasible to gather participants to complete survey measures, for example in observational studies or in the use of secondary text (web blogs, speeches, essays, etc.).

Another potential application is in helping students shape career interests and paths that may lead to satisfaction/well-being. For example, in conceptualizing personal goals, students may be encouraged to write about their career goals in narrative form. By running topic modeling, we may be able to surmise pivotal goal components that relate to career satisfaction. Also, we may build classification models to automatically sort goals into choice or performance goals. The classification is trained by preparing a training data where subject matter experts (SMEs) annotate the goal narratives by manually classifying the mentioned goals into the two

categories. Using the annotated narratives, a classification model can be trained and subsequently applied to other goal narratives.

Text mining can also be used in the analysis of mentoring relationship such as in coaching evaluation. Analyzing the written evaluation of or testimonials about career development coaches may be used to analyze which criteria are important for training and development. By analyzing coaches' evaluation using latent semantic analysis (LSA) we may be able to untangle how coaches differ in their evaluation based on length of experience and expertise (Theeboom et al., 2017). This can be done by relating the results of LSA to training outcomes.

Perhaps, one big advantage of text mining in today's research landscape is its ability to analyze big text data that transcend organization and geographical boundaries. Nowadays, online text data about almost anything are readily available and can be collected via an application programing interface (API) or web crawling. Also, these text data can be routinely analyzed even in the absence of a solid theoretical model. A newer approach is to run text analysis on a specific corpus, evaluate whether the patterns extracted are of practical/theoretical significance, before digging deeper in the hope that the extracted information can be used as a basis for building conceptual models (i.e., as is done in grounded theory approach). This approach is particularly useful to study new phenomena in careers research. For example, in the study of new forms of organizing problems, Karanović, Berends, and Engel (2020) applied structural topic modeling to analyze how workers in the platform economy engage with novel forms of organizing across regulatory structures. By analyzing posts in fora set up by the workers themselves, the authors found that workers respond differently to organizing solutions imposed by these platforms. Hence, this may reveal a new dimension of worker–employer relationship in platforms where this relationship is fuzzy. Note that using traditional methods in careers research to investigate this phenomenon may be challenging because it involves non-traditional workers who are dispersed across the globe.

Finally, text mining is not limited to the above-mentioned topics or applications, it can be used to study well-establish career topics such as job turnover (Frederiksen, 2017), job burnout (Lizano & Barak, 2015; Reid, Short, & McKenny, 2017), work/non-work conflict (Dettmers, 2017), and expatriation (Wechtler, Koveshnikov, & Tienari, 2018). As long as there are free texts one can get hold of, the limits of investigation lie on the researchers' creativity and imagination. In order to proceed with the analysis, our recommendation is to first investigate whether existing conceptual models are available to guide the text mining process. A model can be used to judge the validity of the extracted patterns. However, the models should not limit the inquiry as potentially interesting patterns may emerge. New interesting

patterns may be used to improve or even to refute the model. The researcher may subject the patterns to both data triangulation and validation (e.g., use of subject matter experts, convergent validity and predictive validity) as this will enhance the credibility of the newly discovered concept. Table 5.4 summarizes how existing and new career concepts can be analyzed through text mining.

Conclusion

To explicitly address the research questions raised previously, namely:

RQ7 How to build a model that can predict salary from job descriptions in vacancies?

We created a supervised topic model to predict the salary from vacancies using only the text in the vacancy.

RQ8 What are the differences in pay structures among jobs and what drives these differences?

The differences in salaries are accounted by several factors including the knowledge and skills required for the job as well as the location and industry type of the job. Examples knowledge on finance, software development and data analysis command high salaries in the labor market. For the soft skills, organizing, leadership, and problem-solving skills are also highly valued. Other employee characteristics that also appear to incur higher pay are being motivated and ambitious. Although, commonly required communication skills do not figure in the list of high paying skills. One explanation is this skill is commonly required and that it is usually stated in low paying jobs such as in call centers or administrative support. Moreover, some high paying jobs are not necessarily in high demand, this can be explained by the fact that these jobs require higher ability and more experience (more investment in human capital). Another explanation is that the productivity of workers in these job categories contribute more on the revenue of the companies (banking jobs, strategy and consultancy jobs). Professional jobs lie somewhere above the median salary and jobs in retail, admin, social care, customer service or related to leisure are the least paying jobs, with the exception of apprenticeship jobs which usually are unpaid jobs.

Table 5.4. Potential applications of text mining in career research

Career topics	Text source	Text mining methods	Validation	Sample studies
Study of psychological constructs as they relate to career outcomes	Responses in open-ended questions	Frequency analysis, word embedding	Matched to validated Likert-style questionnaires	Shen et al. (2013); Yarkoni, (2010)
Training and development	Coaching/ trainer evaluation	Latent semantic analysis	Validate through subject matter experts and data triangulation. Also through predictive validity	Theeboom et al. (2017)
New employment relationships	Posts in online forum, web blogs, and twitter data	Structured topic modeling	Subject matter experts	Karanović et al. (2020)
Job turnover intention	Articles and exit interview transcripts	Text classification and frequency analysis	Matched to turnover decisions; convergent validity by investigating its relationship to organizational commitment, job satisfaction, and organizational justice	Barrick & Zimmerman, (2005); Frederiksen, (2017); (Lee, Kim, & Mun, 2019)
Work/non-work conflict	Diary; interview transcript; self-evaluation	Longitudinal topic modeling (e.g., dynamic topic models)	Convergent validity as well as predictive validity by relating it to family, physical and psychological health outcomes	Dettmers (2017); Judge, Van Vianen, & De Pater (2004)
Expatriate assignments	Expatriates narratives from semi-structured interviews	Topic modeling and text classification	Subject matter experts (e.g., expatriates coaches)	Salomaa & Makela (2017); Wechtler et al. (2018)

The proliferation of text data about peoples' career and the development of text analytical techniques hold great promise in accelerating and augmenting career research. However, up to this time, most career studies seldom utilize unstructured text as data, if ever used, text analysis is limited to counting word frequencies. Text mining enables career researchers to augment their text analytical toolkit with powerful analytical techniques from machine learning and statistics. As illustrated in this chapter, text mining is a process that broadly consists of three iterative phases: text preprocessing, the application of TM operations, and finally postprocessing. We hope to have provided readers with: (i) a general understanding of how text is transformed to a form amenable to the application of analytical techniques, (ii) the ability to identify appropriate TM operations to address a research problem, and (iii) tactics to validate the results. Finally, TM derived insights may be further investigated through hypothesis testing in order to inform theory. This perhaps is the real crux why career researchers may be interested in using text mining in their investigations. Although, this chapter is largely written to instruct, it is our hope that through this chapter more career researchers will augment their analytical toolkit with TM, and that this in turn will lead to a better understanding of how careers develop.

Conclusion

In the preceding chapters, I have developed a framework for text mining in the organizational research context and text classification tutorial. I also developed models to address the research questions enumerated in the Introduction. In this chapter, I first revisit the framework and each of the research questions and then summarize the main findings and implications of my research in the next section. Then, in the succeeding sections, I discuss the limitations of this work and possible future directions.

Main Findings

Text mining and text classification for organizational researchers

I began with laying down the principles of text mining in the context of organization research and presented a methodological framework of the text mining process and elucidated each of its steps. Associated methodologies are also discussed to provide a sense of the applicability of text mining methodologies within the field of organizational research. Besides discussing steps and techniques, I offered practical recommendations sections on how to start text mining and which tools to use, and I illustrated how text mining can be applied in the field of job analysis specifically to vacancy analysis.

When incorporating text mining in organizational research, domain knowledge or theory can help supplement the more inductive approach often followed in text mining, and I demonstrated the role and importance of such knowledge and theory to a number of text mining steps. Yet, for the expansion of text mining to areas where research goals are not only to classify or to cluster but also to explain, using existing knowledge or theory and incorporating this into the analysis from the start is vital (see also George, Haas, & Pentland, 2014).

In Chapter 1, we enumerated the ways that text mining can help address research questions in organizational research such in identifying research streams in HRM, personality prediction from language use, and clustering documents using cognitive situational models. Our own investigation also demonstrated that text mining can be an indispensable tool for coaching effectiveness, leadership and job analysis.

Delving on a specific class of text mining techniques called text classification in Chapter 2, I set out to increase the uptake of text classification among organizational researchers. Furthermore, I elaborate the text classification process and establish validity for results derived from text classification.

I provided an overview of text classification and a tutorial on how to

conduct actual text classification on the problem of job task information extraction from nursing vacancies. We discussed and demonstrated the different steps in text classification and highlighted issues surrounding the choices of features, classification algorithms, and evaluation metrics. We also outlined ways to evaluate and validate the resulting classification models and prediction from these models. Text classification is an empirical enterprise where experimentation with choices of representation, dimensionality reduction and classification techniques are standard practices. By building several classifiers and comparing them, the final classifier is chosen based on repeated evaluation and validation. Thus, text classification is not a linear process; one must revisit each step iteratively to examine how choices in each step affect succeeding steps. Moreover, classifiers evolve in the presence of new data.

Text classification models are often descriptive as opposed to explanatory in nature, in the sense that they capture the pattern of features and inductively relate these to class membership (Bird et al., 2009). This contrasts with explanatory models whose aims are to explain why the pattern in features leads to the prediction of a class. Nevertheless, the descriptive work can be of use for further theory building too as the knowledge of patterns can be used as a basis for the development of explanatory models. For example, in the part about feature selection we found out that the word *sicherstellung* (to guarantee or to make sure) is useful in detecting sentences containing nursing tasks. Based on this we can define the concept of "task verb", that is, a verb that is indicative of a task in the context of job vacancy. We could then compile a list of verbs that are "task verbs" and postulate that task verbs pair with noun or verb phrases to form task sentences. Further trials could then be designed to validate this concept and establish the relationship between features and patterns. In this way, we are not only detecting patterns, but we also attempt to infer their properties and their relationship to class membership.

Whether a descriptive model suffices or whether an explanatory model is needed depends on the objectives of a specific study. If the objective is accurate and reliable categorization (e.g., when one is interested in using the categorized text as input to other systems) then a descriptive model will suffice, although the outcomes still need to be validated. On the other hand, if the objective is to explain how patterns lead to categorization or how structure and form lead to meaning, then an explanatory model is required.

In this thesis, we tried to present text classification in such a manner that organizational researchers can understand the underlying process. However, in practice, organizational researchers will often work with technical experts to make choices on the algorithms and assist in tweaking and tuning the parameters of the resulting model. The role of organizational researchers then is to provide the

research questions, help select the relevant features, and provide insights considering the classification output. These insights might lead to further investigation and ultimately to theory development and testing.

Finally, we conclude that text classification offers great potential to make the conduct of text-based organizational research fast, reliable, and effective. The utility of text classification is most evident when there is a need to analyze massive text data, and in some cases text classification can recover patterns that are difficult for humans to detect. Otherwise, manual qualitative text analysis procedures may suffice. As noted, the increased use of text classification in organizational research will likely not only contribute to organizational research, but also to the advancement of text classification research, because real problems and existing theory can further simulate the development of new techniques.

The two chapters which have been published in a leading methods journal have managed to garner many citations and paper downloads over the past two years. The traction that these two chapters have generated exemplifies the strong uptake of organization researchers that now use text analysis.

A Fresh Perspective on Job Analysis: Extracting Job Information from Job vacancies

Abundant sources of job information are Job vacancies posted in various online employment platforms. However, most job vacancies are unstructured and need further preprocessing and analysis before relevant information can be used. Before we provide each of the research questions and our associated findings.

RQ1 What features are useful to identify work activities and worker attributes, and can these features be combined to build an automatic and accurate classifier that can distinguish between work activities and worker attributes contained in job vacancies?

The application of several classifiers showed that job information can be extracted automatically and efficiently from vacancies. Additionally, I note the following findings:

1) Knowledge from the job analysis field and expert judgement can be used to preselect features useful for classification.
2) Although keyword extraction may be useful to some extent, incorporating sentence and word level characteristics, syntactic information of words and grammatical pattern would greatly enhance the classification performance.
3) Among the classifiers tested, Logistic regression and stacked ensemble seemed to perform exceptionally well. However, the performance

of other classifiers are not far behind which demonstrates the effectiveness of the chosen features.

RQ2 Can the extracted work activities be used to construct task groups and meaningfully cluster jobs based on tasks?
Instead of manually going through thousands of vacancies, through the application of classification models developed here we managed to condense worker attributes to a few hundred topics.

RQ3 Can the task groups be validated by comparing it to tasks as enumerated in the European Skills, Competences, and Occupations (ESCO)?

The extracted worker attributes highly overlapped with existing a compiled hierarchy of skill pillars of ESCO which was obtained by painstakingly interviewing job holders and job experts. This signifies convergent validity. The simplicity of the resulting model did not limit its usefulness, in fact it worked well extracting the intended information.

Predicting the next job and when to change jobs

Apart from job vacancies, resumes also provide rich sources of information about candidates. Most of these resumes are compiled by recruitment agencies to have a readily available pool of candidates to fill in some vacancies. However, matching candidates to vacancies extends beyond matching through skills and competencies, as it is also important for recruiters to know whether someone is willing to change jobs. Our results yielded the following answers to our research questions:

RQ4 How to build a model that can predict when someone is going to change job?

For predicting when someone will change jobs, we used the employment histories reported by job seekers in their CVs. We then applied hierarchical Bayes model to study job transitions. The models considered here yielded better performance compared to the naïve or non-ML approaches of just assuming that everyone will make a transition, or that no one will make a transition. Aside from the predictive power, the classification insights about how candidates transition between jobs and about job relationships were elucidated. Variables associated with individual differences of career enactment are the most influential variables. Just by inspecting how long a candidate is in the labor market, his/her age, his/her previous inter-job times, how long a candidate is in the current job, and the number of previous job

changes, it is possible to predict whether he/she will make a transition within a 2-year period. It was also revealed that contrary to studies that assumed that the longer a candidate has stayed in the current job the likelihood of changing job increases (James et al., 2018; L. Li et al., 2017; Xu et al., 2015), results from this study showed that length of time in the current job is inversely propositional to the likelihood of changing job, given that previous inter-job times were also long. Moreover, education level was found to affect job mobility, specifically university level candidates appear to be more mobile than those who only finished secondary education.

RQ5 How to build a model that can predict the mostly likely next job of an individuals?

For predicting the most likely next job of individuals, we used the same data on employment histories reported by job seekers. We built a model that can predict the top-5 job options for the next job of candidates. We co-trained two LSTM models which to the best of our knowledge has not been done before. Some studies use the seq-to-seq LSTM models which claim to predict the entire career path of individuals (L. Li et al., 2017). This may be useful in some applications such as in exploring potential career progression, however, these models do not give the timing of job changes and are not dynamic and flexible enough to account for changing circumstances in the labor market and individual mobility patterns.

RQ6 How to use the models from RQ_4-RQ_6 to find similarities among jobs and does industry differences play a role in mobility, that is, some industries exhibit higher number of job transitions of employees.

Variables associated with individual differences of career enactment are the most influential variables. Just by inspecting how long a candidate is in the labor market, his/her age, his/her previous inter-job times, how long a candidate is in the current job, and the number of previous job changes, it is possible to predict whether he/she will make a transition within a 2-year period. It was also revealed that contrary to studies that assumed that the longer a candidate has stayed in the current job the likelihood of changing job increases (James et al., 2018; L. Li et al., 2017; Xu et al., 2015), results from this study showed that length of time in the current job is inversely propositional to the likelihood of changing job, given that previous inter-job times were also long. Moreover, education level was found to affect job mobility, specifically university level candidates appear to be more mobile than those who only finished secondary education.

Apart from individual level characteristics and patterns of mobility, industry differences and specific jobs also determine job mobility. As mentioned previously, once people reach managerial or administrative positions, they become less mobile. This is also true for highly technical jobs such as programmers and finance specialists. On the other hand, candidates who were sellers, or chefs and other food preparers were highly mobile.

Our proposal is to analyze job changes of individuals, because these job changes capture the screening done by candidates when they assess their compatibility to the job which is possibly influenced by their experience or knowledge gained from previous jobs (Ferreira, 2016). It also reflects candidate preferences because they decided to accept the job. Inclusion of job holder differences pertaining to how they enact their career is informative, because jobs should not be viewed as entirely distinct from the job holders since they influence and shape jobs (Berg et al., 2013).

Text mining in career studies: Generating insights from unstructured textual data

Many problems in career research can be addressed using text data. Also, most studies tend to focus on using numerical data (longitudinal, survival analysis), and text data are rarely used. Hence, I set out to popularize the use of text mining to career research, and demonstrated the creation of a model that can predict salary from vacancies. Understanding salary structures of the labor market have long been studied in the literature, and here I showed that as an alternative we can investigate how salary differs across jobs. The answers to each of the research questions are the following:

RQ7 How to build a model that can predict salary from job descriptions in vacancies?

A model based on supervised topic modeling was developed to address RQ1. The results of evaluation showed a decent predictive performance. Several challenges were encountered during model building. One is the unbalanced characteristic of the data due to some jobs being underrepresented. Another is the different ways that compensation is reported in vacancies, some report on an hourly basis and other report salary per annum. Some jobs do not even pay (internship or secondment). I managed to address the challenges by apply techniques to handle unbalanced data (oversampling and SMOTE) and standardizing the salary by converting everything to salary per year.

RQ8 What are the differences in pay structures among jobs and what drives these differences?

Perhaps, the most interesting part is the insights provided by the model on the relationship of the job characteristics and pay. The model revealed that pay is highly influenced by the role, industry type, location, and skill required by the job. For example, managerial and financial jobs tend to pay higher, and hospitality and customer support jobs pay relatively low. IT and other highly technical skills command higher pay in the labor market, but communication skills, although a common requirement of employment, is at the lower end of the pay scale. One explanation is vacancies that explicitly mention this skill are associated to lower pay.

Future work

Establishing validity

One recurring theme in this dissertation is establishing or evaluating the reliability and validity of a given study using text mining. A key question is whether we should adopt our evaluative criteria from the qualitative research tradition (cf. Yu et al., 2011), the quantitative research tradition, or perhaps even both. Of course, insofar as a text mining study can withstand scrutiny from both methodological perspectives, this only serves to increase its credibility. Yet, the relevance of specific quality measures is likely to be contingent upon the epistemological orientation and specific objectives of the researcher. That is, for more exploratory or descriptive studies, such as those relying on topic modelling and clustering (see Table 1.1), it is not mandatory to impose strategies designed for establishing the validity of inferences."[B]y definition 'inference' is an act of expanding the conclusion from a smaller subset to a broader set (e.g., from the sample statistics to the population parameter), but most qualitative studies do not aim to make 'valid inferences'." (C. H. Yu et al., 2011, p. 736). Krippendorf (2012) echoed this for CATA stating that "Deductive and inductive inferences are not central to content analysis" (p.36). Nevertheless, text mining output can also provide a starting point for studies aiming to take an inferential route.

Text mining also has limitations and constraints. *Text* mining requires specific expertise and resources. Not all organizations/researchers have the computing resources to develop massive text mining applications or the necessary expertise to execute these appropriately. The expertise and computing resources constraint could be addressed by outsourcing the task to companies and people who specialize in text mining. Another limitation is the question of how representative the information is found in text data. The quality of the data will matter for the outcomes, as with

any type of data. The limitation of text data as an incomplete source of information could be mitigated by supplementing the analysis with additional types of data. For instance, in our job vacancy analysis we could triangulate our findings against the Occupational Information Network (Jeanneret & Strong, 2003), or other data sources that provide rich job information.

Of course, there are also validity concerns in relying on online vacancies as a data source. First, there are noticeable differences in the quality of the information across sources. For example, vacancies posted by recruitment agencies are often lower in quality (e.g., level of detail, clarity of information) compared to vacancies posted by organizations. Second, online data, like all secondary data, is often produced for very different purposes than the research purpose it may subsequently be repurposed for, in this case job analysis. For example, online vacancies are aimed at recruiting employees, which means that the included information might be biased through advertising only certain, mainly positive, aspects of the job and/or not mentioning very mundane tasks. Tasks unique to the traditional task inventory, for example, included more mundane and less positive, but very frequently occurring tasks in the nursing profession (e.g., washing patients, changing patients, cleaning beds, checking temperature). This can be a subject of an investigation on sentiment analysis. Third, not all jobs are advertised online (Sodhi & Son, 2007), potentially leaving out relevant information and jobs. One way to delineate the discrepancy is to explore other sources of job information. Fourth, job vacancies may provide an asymmetric source of information about jobs in that it only considers jobs from the point of view of employers. Hence, information extracted from vacancies may not be directly useful to study job seeker preferences. Our recommendation to further validate the relationships is to compare the results we obtained with alternative sources of information, such as interviewing SMEs or job incumbents, and computing measures traditionally used in inter-rater reliability.

Ethical considerations

The different legal and ethical considerations that come with using particular forms of text data form a final limitation. Some text data are proprietary or contain privacy sensitive information that may be difficult to anonymize. The difficulty of obtaining permission to use text data can be addressed in part by implementing safeguards to protect the confidentiality of the data and to perform the analysis securely. Wider ethical concerns (Van Wel & Royakkers, 2004) on the use of 'Big' data, urgently need further and wider development and discussion.

In relation to the developed models and the corresponding results, while there are obvious benefits, their use could also lead to unintended outcomes on the part of the employees. For example, the employer who holds such a system could

use the features that are correlated with employees quitting and manipulate these features so that employees do quit without directly firing the employees. As with any other innovation, the use of our models comes with associated risks and threats. Hence, guidelines and policies must be set that define the limits of the use of such models where the rights and privacy of employees are protected. Another potential issue are the algorithms may produce erroneous prediction. Models are never one hundred percent accurate, in fact over time their performances may degrade. Hence, decision makers must always triangulate the results with other data sources and not to rely solely on the predictions, since it may produce results that may be contrary to what is happening in practice. Regarding the models becoming outdated as a possible source of bias, the best practice in deployment is to perform model monitoring and updating to ensure that the predictions can still be trusted. For example, the model can be retrained using new data. The techniques presented here can update the model without the need for frequent retraining. Researchers may also use the techniques from machine learning such as online learning for model updating.

Recommendation on possible methodological extension

We hope our discussion of text mining helps foster dialogue and collaboration between organizational researchers and data scientists, particularly text miners. Though most discussions here have centered on how text mining can help organizational research, text mining as a field also has something to gain from organizational research. The richness of problems that organizational research is trying to analyze can stimulate the creation of novel text mining methodologies, thereby contributing to its advancement. In sum, the deluge of text data, the need to combine qualitative approaches with their quantitative counterparts, and the resulting progress for the two fields (organizational research and text mining) brought by the interplay of theory and methods make the inclusion of text mining methods ever more relevant to organizational research.

Text mining is a wide research field and there are many other techniques that were not covered here examples are other other sequence models such as conditional random fields. Moreover, new developments in deep learning techniques for text understanding (for more on this we refer the reader to Maas et al., 2011; Mikolov, Chen, Corrado, & Dean, 2013; Zhang & LeCun, 2015) may contribute to the creation of accurate models. However, existing pretrained deep learning models were trained on natural language and not on job transitions. In order to use the full potential of deep learning for this application, more data on job transitions should be collected. Here, we offered a step on this direction by training an LSTM model to predict the next job .

Finally, the information extracted by the techniques developed here can be further used as input to other types of analysis, such as analyzing trends of worker attributes required by organizations across time, occupations, companies, and geographical regions given that these types of information are generally provided in the vacancies. Also, one can build a network of work activities to examine relationship among tasks.

Further research can move into several directions. First, although the previous literature presents several approaches towards job recommendation using semantic matching, sequential pattern mining, and survival analysis, they lacked results on the similarities and differences between the generated recommendations of these approaches. Hence, further research should consider how these different approaches compare, and could be combined in a hybrid recommender system.

In contrast with studies that either focus on methodology or theory (not both), this study seeks a middle ground by not only demonstrating the utility of text and sequence mining in practical application, such as in improving job recommendation but also trying to incorporate theory. The output from the models may be used for further investigation and may even become a source of hypothesis that will lead to further investigations.

References

Abdessalem, W. K. B., & Amdouni, S. (2011). E-recruiting support system based on text mining methods. *International Journal of Knowledge and Learning*, 7(3), 220–232. https://doi.org/10.1504/IJKL.2011.044542

Abe, M. (2009). "Counting your customers" one by one: A hierarchical Bayes extension to the Pareto/NBD model. *Marketing Science,* 28(3), 541–553.

Addom, B. K., Kim, Y., & Stanton, J. M. (2011). eScience professional positions in the job market: A content analysis of job advertisements. *Proceedings of the 2011 IConference*, 630–631. https://doi.org/10.1145/1940761.1940846

Aggarwal, C. C., & Zhai, C. (2012a). A Survey of Text Classification Algorithms. In C. C. Aggarwal & C. Zhai (Eds.), *Mining Text Data* (pp. 163–222). Springer US. https://doi.org/10.1007/978-1-4614-3223-4_6

Aggarwal, C. C., & Zhai, C. (2012b). A survey of text classification algorithms. In *Mining text data* (pp. 163–222). Springer. http://link.springer.com/chapter/10.1007/978-1-4614-3223-4_6

Aizawa, A. (2003). An information-theoretic perspective of tf–idf measures. *Information Processing & Management*, 39(1), 45–65. https://doi.org/10.1016/S0306-4573(02)00021-3

Albitz, R. S. (2002). Electronic resource librarians in academic libraries: A position announcement analysis, 1996-2001. *Portal: Libraries and the Academy*, 2(4), 589–600.

Algarni, A., & Tairan, N. (2014). Feature Selection and Term Weighting. *Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT) - Volume 01*, 336–339. https://doi.org/10.1109/WI-IAT.2014.53

Alpaydin, E. (2014). *Introduction to machine learning*. MIT press.

Amato, F., Boselli, R., Cesarini, M., Mercorio, F., Mezzanzanica, M., Moscato, V., Persia, F., & Picariello, A. (2015). Challenge: Processing web texts for classifying job offers. P*roceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, 460–463. https://doi.org/10.1109/ICOSC.2015.7050852

AP, Meltwater settle copyright dispute. (2013, July 29). http://www.ap.org/Content/AP-In-The-News/2013/AP-Meltwater-settle-copyright-dispute

Arthur Jr, W., Bennett Jr, W., Edens, P. S., & Bell, S. T. (2003). Effectiveness of training in organizations: A meta-analysis of design and evaluation features. *Journal of Applied Psychology,* 88(2), 234–245. https://doi.org/10.1037/0021-9010.88.2.234

Arthur, M. B., Hall, D. T., & Lawrence, B. S. (1989). Generating new directions in career theory: The case for a transdisciplinary approach. *Handbook of Career Theory,* 7, 25.

Arun, R., Suresh, V., Madhavan, C. E. V., & Murthy, M. N. N. (2010). On Finding the Natural Number of Topics with Latent Dirichlet Allocation: Some Observations. *Advances in Knowledge Discovery and Data Mining,* 391–402. https://doi.org/10.1007/978-3-642-13657-3_43

Atteveldt, W. van, Kleinnijenhuis, J., Ruigrok, N., & Schlobach, S. (2008). Good News or Bad News? Conducting Sentiment Analysis on Dutch Text to Distinguish Between Positive and Negative Relations. *Journal of Information Technology & Politics,* 5(1), 73–94. https://doi.org/10.1080/19331680802154145

Avrahami, D., Pessach, D., Singer, G., & Chalutz Ben-Gal, H. (2022). A human resources analytics and machine-learning examination of turnover: Implications for theory and practice. *International Journal of Manpower, ahead-of-print*(ahead-of-print). https://doi.org/10.1108/IJM-12-2020-0548

Baskin, I. I., Marcou, G., Horvath, D., & Varnek, A. (2017). Bagging and boosting of classification models. *Tutorials in Chemoinformatics*, 241–247.

Benabderrahmane, S., Mellouli, N., & Lamolle, M. (2018). On the predictive analysis of behavioral massive job data using embedded clustering and deep recurrent neural networks. *Knowledge-Based Systems*, 151, 95–113. https://doi.org/10.1016/j.knosys.2018.03.025

Berg, J. M., Dutton, J. E., & Wrzesniewski, A. (2013). Job crafting and meaningful work. In B. J. Dik, Z. S. Byrne, & M. F. Steger (Eds.), *Purpose and meaning in the workplace* (pp. 81–104). American Psychological Association. https://doi.org/10.1037/14183-005

Bernerth, J. B., & Aguinis, H. (2016). A Critical Review and Best-Practice Recommendations for Control Variable Usage. *Personnel Psychology*, 69(1), 229–283. https://doi.org/10.1111/peps.12103

Berry, M. W., & Castellanos, M. (2008). *Survey of Text Mining II - Clustering, Classification, and Retrieval.* http://www.springer.com/gp/book/9781848000452

Bika, N. (2019). *How to calculate recruitment costs for budget planning.* Workable. https://resources.workable.com/tutorial/recruitment-costs-budget

Bingham, E., & Mannila, H. (2001). Random projection in dimensionality reduction: Applications to image and text data. *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 245–250.* http://dl.acm.org/citation.cfm?id=502546

Binning, J. F., & Barrett, G. V. (1989a). Validity of personnel decisions: A conceptual analysis of the inferential and evidential bases. *Journal of Applied Psychology,* 74(3), 478–494. https://doi.org/10.1037/0021-9010.74.3.478

Binning, J. F., & Barrett, G. V. (1989b). Validity of personnel decisions: A conceptual analysis of the inferential and evidential bases. *Journal of Applied Psychology,* 74(3), 478–494. https://doi.org/10.1037/0021-9010.74.3.478

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with Python.* O'Reilly Media, Inc. https://books.google.nl/books?hl=en&lr=&id=KGIbfiiP1i4C&oi=fnd&pg=PR5&dq=nltk+book&ots=Y2CoE0KEI5&sig=cS5nUZ1YtuLquL4thXiVG5XWgw0

Blei, D. M., & Lafferty, J. D. (2007). A correlated topic model of science. *The Annals of Applied Statistics,* 17–35.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research,* 3, 993–1022.

Borovikov, E. (2014). A survey of modern optical character recognition techniques.

ArXiv:1412.4183 [Cs]. http://arxiv.org/abs/1412.4183

Boselli, R., Cesarini, M., Marrara, S., Mercorio, F., Mezzanzanica, M., Pasi, G., & Viviani, M. (2017). WoLMIS: A labor market intelligence system for classifying web job vacancies. *Journal of Intelligent Information Systems,* 1–26.

Brannick, M. T., Levine, E. L., & Morgeson, F. P. (2007a). *Job and Work Analysis: Methods, Research, and Applications for Human Resource Management.* SAGE Publications.

Brannick, M. T., Levine, E. L., & Morgeson, F. P. (2007b). *Job and Work Analysis: Methods, Research, and Applications for Human Resource Management.* SAGE Publications.

Breiman, L. (1996). *Bagging predictors. Machine Learning,* 24(2), 123–140.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees.* CRC press.

Briscoe, J. P., & Hall, D. T. (2006). The interplay of boundaryless and protean careers: Combinations and implications. *Journal of Vocational Behavior*, 69(1), 4–18. https://doi.org/10.1016/j.jvb.2005.09.002

Brodersen, K. H., Ong, C. S., Stephan, K. E., & Buhmann, J. M. (2010). The balanced accuracy and its posterior distribution. *Pattern Recognition (ICPR), 2010 20th International Conference On*, 3121–3124. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5597285

Brooks, J., McCluskey, S., Turley, E., & King, N. (2015). The Utility of Template Analysis in Qualitative Psychology Research. *Qualitative Research in Psychology,* 12(2), 202–222. https://doi.org/10.1080/14780887.2014.955224

Bsoul, Q., Salim, J., & Zakaria, L. Q. (2013). An Intelligent Document Clustering Approach to Detect Crime Patterns. *Procedia Technology,* 11, 1181–1187. https://doi.org/10.1016/j.protcy.2013.12.311

Buhrmester, M., Kwang, T., & Gosling, S. D. (2011). Amazon's Mechanical Turk A New Source of Inexpensive, Yet High-Quality, Data? *Perspectives on Psychological Science*, 6(1), 3–5. https://doi.org/10.1177/1745691610393980

Burges, C. J. (2010). Dimension reduction: *A guided tour.* Now Publishers Inc. https://books.google.nl/books?hl=en&lr=&id=jpmz8ZpUmJUC&oi=fnd&pg=PA1&dq=dimension+reduction+a+guided+tour&ots=Sk46OcbSIy&sig=3dVvN7RX4gyU7V8ouK5CNk4vpIc

Burke, S. M. (2002). Perl & LWP: *Fetching Web Pages, Parsing HTML, Writing Spiders & More.* O'Reilly Media, Inc.

Bylinskii, Z., Alsheikh, S., Madan, S., Recasens, A., Zhong, K., Pfister, H., Durand, F., & Oliva, A. (2017). Understanding infographics through textual and visual tag prediction. *ArXiv Preprint ArXiv:*1709.09215.

Campbell, D. T., & Fiske, D. W. (1959a). Convergent and discriminant validation by the multitrait-multimethod matrix. *Psychological Bulletin, 56*(2), 81–105. https://doi.org/10.1037/h0046016

Campbell, D. T., & Fiske, D. W. (1959b). Convergent and discriminant validation by the multitrait-multimethod matrix. *Psychological Bulletin, 56*(2), 81–105. https://doi.org/10.1037/h0046016

Campbell, J. P., McCloy, R. A., Oppler, S. H., & Sager, C. E. (1993). A theory of performance. *Personnel Selection in Organizations,* 3570, 35–70.

Campion, M. A., Morgeson, F. P., & Mayfield, M. S. (1999). O* NET's theoretical contributions to job analysis research. *An Occupational Information System for the 21st Century: The Development of O* NET,* 297–304.

Cao, J., Xia, T., Li, J., Zhang, Y., & Tang, S. (2009). A Density-based Method for Adaptive LDA Model Selection. *Neurocomput.*, 72(7–9), 1775–1781. https://doi.org/10.1016/j.neucom.2008.06.011

Cardie, C., & Wilkerson, J. (2008). Text Annotation for Political Science Research. *Journal of Information Technology & Politics,* 5(1), 1–6. https://doi.org/10.1080/19331680802149590

Cavnar, W. B., Trenkle, J. M., & others. (1994). N-gram-based text categorization. *Ann Arbor MI, 48113*(2), 161–175.

Cedefop. (2019). *Online job vacancies and skills analysis: A Cedefop pan-European approach.*

Chan, S. W. K., & Chong, M. W. C. (2017). Sentiment analysis in financial texts. *Decision Support Systems*, 94, 53–64. https://doi.org/10.1016/j.dss.2016.10.006

Chan, S. W. K., & Franklin, J. (2011). A text-based decision support system for financial sequence prediction. *Decision Support Systems,* 52(1), 189–198. https://doi.org/10.1016/j.dss.2011.07.003

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly Detection: A Survey. A*CM Comput. Surv.*, 41(3), 15:1-15:58. https://doi.org/10.1145/1541880.1541882

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research,* 16, 321–357.

Chen, J., Huang, H., Tian, S., & Qu, Y. (2009). Feature selection for text classification with Naïve Bayes. *Expert Systems with Applications,* 36(3, Part 1), 5432–5435. https://doi.org/10.1016/j.eswa.2008.06.054

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining,* 785–794.

Colihan, J., & Burger, G. K. (1995). Constructing Job Families: An Analysis of Quantitative Techniques Used for Grouping Jobs. *Personnel Psychology,* 48(3), 563–586. https://doi.org/10.1111/j.1744-6570.1995.tb01771.x

Conrad, J. G., Al-Kofahi, K., Zhao, Y., & Karypis, G. (2005). Effective Document Clustering for Large Heterogeneous Law Firm Collections. *Proceedings of the 10th International Conference on Artificial Intelligence and Law,* 177–187. https://doi.org/10.1145/1165485.1165513

Dave, K., Lawrence, S., & Pennock, D. M. (2003a). Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews. *Proceedings of the 12th International Conference on World Wide Web,* 519–528. https://doi.org/10.1145/775152.775226

Dave, K., Lawrence, S., & Pennock, D. M. (2003b). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. *Proceedings of the 12th International Conference on World Wide Web,* 519–528. http://dl.acm.org/citation.cfm?id=775226

de Oliveira, J. G. (2019). *A study on Gradient Boosting algorithms.*

Denisi, A. S. (1976). The implications of job clustering for training programmes. *Journal of Occupational Psychology,* 49(2), 105–113. https://doi.org/10.1111/j.2044-8325.1976.tb00335.x

Derpanis, K. G. (2006). *K-Means Clustering*. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.217.5155

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv Preprint ArXiv:1810.04805*.

Dierdorff, E. C., & Morgeson, F. P. (2009). Effects of Descriptor Specificity and Observability on Incumbent Work Analysis Ratings. *Personnel Psychology,* 62(3), 601–628. https://doi.org/10.1111/j.1744-6570.2009.01151.x

Dierdorff, E. C., & Wilson, M. A. (2003). A Meta-Analysis of Job Analysis Reliability. *Journal of Applied Psychology August 2003, 88*(4), 635–646.

Dietterich, T. G. (1997). Machine-Learning Research. *AI Magazine, 18*(4), 97.

Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM, 55*(10), 78–87.

Dong, Y.-S., & Han, K.-S. (2004). A comparison of several ensemble methods for text categorization. *Services Computing, 2004.(SCC 2004). Proceedings. 2004 IEEE International Conference On*, 419–422. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1358033

Duggan, P. (2010). *Methods of matching job profiles and candidate profiles* (Patent No. US20100153290 A1). http://www.google.com/patents/US20100153290

Dumais, S., Platt, J., Heckerman, D., & Sahami, M. (1998). Inductive learning algorithms and representations for text categorization. *Proceedings of the Seventh International Conference on Information and Knowledge Management*, 148–155. http://dl.acm.org/citation.cfm?id=288651

Duriau, V. J., Reger, R. K., & Pfarrer, M. D. (2007). A Content Analysis of the Content Analysis Literature in Organization Studies: Research Themes, Data Sources, and Methodological Refinements. *Organizational Research Methods*, 10(1), 5–34. https://doi.org/10.1177/1094428106289252

Dusi, S., Mercorio, F., & Mezzanzanica, M. (2015). *Big data meets web job vacancies: Trends, challenges and development directions*. Rainer Hampp Verlag, Munchen: p.

El-Hamdouchi, A., & Willett, P. (1989). Comparison of Hierarchic Agglomerative Clustering Methods for Document Retrieval. *The Computer Journal*, 32(3), 220–227. https://doi.org/10.1093/comjnl/32.3.220

Elkan, C. (2001). The Foundations of Cost-sensitive Learning. *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2*, 973–978. http://dl.acm.org/citation.cfm?id=1642194.1642224

Eyheramendy, S., Lewis, D. D., & Madigan, D. (2003). *On the naive bayes model for text categorization.* http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.20.4949

Fabo, B., Beblavỳ, M., & Lenaerts, K. (2017). The importance of foreign language skills in the labour markets of Central and Eastern Europe: Assessment based on data from online job portals. *Empirica*, 44(3), 487–508.

Fabo, B., & Kahanec, M. (2020). THE ROLE OF COMPUTER SKILLS ON THE OCCUPATION LEVEL. *EUROPEAN JOURNAL OF BUSINESS SCIENCE AND TECHNOLOGY, 87*.

Fader, P. S., & Hardie, B. G. (2009). Probability models for customer-base analysis. *Journal of Interactive Marketing*, 23(1), 61–69.

Fader, P. S., & Hardie, B. G. S. (2005). *A note on deriving the Pareto/NBD model and related expressions*.

Faguo, Z., Fan, Z., Bingru, Y., & Xingang, Y. (2010). Research on Short Text Classification Algorithm Based on Statistics and Rules. 2010 *Third International Symposium on Electronic Commerce and Security (ISECS)*, 3–7. https://doi.org/10.1109/ISECS.2010.9

Fernandes, J., Artífice, A., & Fonseca, M. J. (2017). *AUTOMATIC ESTIMATION OF THE LSA DIMENSION*. 301–305. http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0003666103090313

Ferreira, A. J., & Figueiredo, M. A. T. (2012). Boosting Algorithms: A Review of Methods, Theory, and Applications. In C. Zhang & Y. Ma (Eds.), *Ensemble Machine Learning* (pp. 35–85). Springer US. https://doi.org/10.1007/978-1-4419-9326-7_2

Ferreira, P. (2016). *Explaining job mobility: An integrated analysis of the determinants of promotions and firm separations.*

Finn, A., & Kushmerick, N. (2006). Learning to classify documents according to genre. *Journal of the American Society for Information Science and Technology, 57*(11), 1506–1518. https://doi.org/10.1002/asi.20427

Flach, P. (2012). *Machine Learning: The Art and Science of Algorithms That Make Sense of Data.* Cambridge University Press.

Fodor, I. K. (2002). *A survey of dimension reduction techniques.* Technical Report UCRL-ID-148494, Lawrence Livermore National Laboratory. https://e-reports-ext.llnl.gov/pdf/240921.pdf

Ford, J. K., MacCallum, R. C., & Tait, M. (1986). The application of exploratory factor analysis in applied psychology: A critical review and analysis. *Personnel Psychology, 39*(2), 291–314.

Forman, G. (2003). An Extensive Empirical Study of Feature Selection Metrics for Text Classification. *J. Mach. Learn. Res.,* 3, 1289–1305.

Fox, C. (1989). A Stop List for General Text. *SIGIR Forum*, 24(1–2), 19–21. https://doi.org/10.1145/378881.378888

Fox, C. (1992). Lexical analysis and stoplists. In W. B. Frakes & R. Baeza-Yates (Eds.), *Information Retrieval: Data Structures and and Algorithms* (pp. 102–130). Prentice-Hall, Inc. http://dl.acm.org/citation.cfm?id=129687.129694

Frakes, W. B. (1992). *Stemming Algorithms.*

Frakes, W. B., & Baeza-Yates, R. (Eds.). (1992a). Information Retrieval: Data Structures and Algorithms. Prentice-Hall, Inc.

Frakes, W. B., & Baeza-Yates, R. (1992b). *Information retrieval: Data structures and algorithms*. http://www.citeulike.org/group/328/article/308697

Friedman, J., Hastie, T., Simon, N., & Tibshirani, R. (2016). *Package glmnet: Lasso and*

elastic-net regularized generalized linear models ver 2.0.

Fu, Y., Zhu, X., & Li, B. (2013). A survey on instance selection for active learning. *Knowledge and Information Systems, 35*(2), 249–283. https://doi.org/10.1007/s10115-012-0507-8

Gallivan, M., Truex, D. P., III, & Kvasny, L. (2002). An Analysis of the Changing Demand Patterns for Information Technology Professionals. *Proceedings of the 2002 ACM SIGCPR Conference on Computer Personnel Research*, 1–13. https://doi.org/10.1145/512360.512363

Ganzeboom, H. B. (2010). International standard classification of occupations ISCO-08 with ISEI-08 scores. *Version July*, 27, 2010.

Gaur, B. (2020). HR4.0: An Analytics Framework to redefine Employee Engagement in the Fourth Industrial Revolution. *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 1–6. https://doi.org/10.1109/ICCCNT49239.2020.9225456

George, G., Haas, M., & Pentland, A. (2014). FROM THE EDITORS BIG DATA AND MANAGEMENT. *Academy of Management Journal, 57*(2), 321–326.

Glady, N., Baesens, B., & Croux, C. (2009). A modified Pareto/NBD approach for predicting customer lifetime value. *Expert Systems with Applications, 36*(2, Part 1), 2062–2071. https://doi.org/10.1016/j.eswa.2007.12.049

Goldman, S. A. (2010). Computational Learning Theory. In M. J. Atallah & M. Blanton (Eds.), *Algorithms and Theory of Computation Handbook* (2nd ed., Vol. 1, pp. 26–26). Chapman & Hall/CRC. http://dl.acm.org/citation.cfm?id=1882757.1882783

Gonçalves, T., & Quaresma, P. (2005). Is Linguistic Information Relevant for the Classification of Legal Texts? *Proceedings of the 10th International Conference on Artificial Intelligence and Law,* 168–176. https://doi.org/10.1145/1165485.1165512

Graham, I. S. (1995). *The HTML sourcebook.* John Wiley & Sons, Inc. http://dl.acm.org/citation.cfm?id=526978

Grimes, S. (2008, August 1). Unstructured Data and the 80 Percent Rule. *Breakthrough Analysis.* https://breakthroughanalysis.com/2008/08/01/unstructured-data-and-the-80-percent-rule/

Gui, H., Liu, H., Meng, X., Bhasin, A., & Han, J. (2016). Downside management in recommender systems. *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 394–401. https://doi.org/10.1109/ASONAM.2016.7752264

Guo, Y., Li, Y., & Shao, Z. (2015). An Ant Colony-based Text Clustering System with Cognitive Situation Dimensions. *International Journal of Computational Intelligence Systems, 8*(1), 138–157. https://doi.org/10.1080/18756891.2014.963986

Gupta, V., & Lehal, G. S. (2009). A Survey of Text Mining Techniques and Applications. *Journal of Emerging Technologies in Web Intelligence*, 1(1), 60–76.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. T*he Journal of Machine Learning Research*, 3, 1157–1182.

Guyon, I., Gunn, S., Nikravesh, M., & Zadeh, L. (2008). *Feature Extraction: Foundations*

and Applications. Springer.

Hachen, D. S. (1988). The Competing Risks Model: A Method for Analyzing Processes with Multiple Types of Events. *Sociological Methods & Research*, 17(1), 21–54. https://doi.org/10.1177/0049124188017001002

Hachen, D. S. (1990). Three Models of Job Mobility in Labor Markets. *Work and Occupations,* 17(3), 320–354. https://doi.org/10.1177/0730888490017003004

Hachen, D. S. (1992). Industrial Characteristics and Job Mobility Rates. *American Sociological Review,* 57(1), 39–55. https://doi.org/10.2307/2096143

Harish, B. S., Guru, D. S., & Manjunath, S. (2010a). Representation and Classification of Text Documents: A Brief Review. I*nternational Journal of Computer Applications IJCA*, RTIPPR(2), 110–119.

Harish, B. S., Guru, D. S., & Manjunath, S. (2010b). Representation and Classification of Text Documents: A Brief Review. *International Journal of Computer Applications IJCA*, RTIPPR(2), 110–119.

Harlow, L. L., & Oswald, F. L. (2016). Big Data in Psychology: Introduction to the Special Issue. *Psychological Methods*, 21(4), 447–457. https://doi.org/10.1037/met0000120

Harper, R. (2012). The collection and analysis of job advertisements: A review of research methodology. *Library and Information Research,* 36(112), 29–54.

Harvey, R. J. (1986). Quantitative Approaches to Job Classification: A Review and Critique. *Personnel Psychology,* 39(2), 267–289.

Hayton, J. C., Allen, D. G., & Scarpello, V. (2004). Factor retention decisions in exploratory factor analysis: A tutorial on parallel analysis. *Organizational Research Methods*, 7(2), 191–205.

Hindle, D., & Rooth, M. (1993). Structural Ambiguity and Lexical Relations. *Comput. Linguist., 19*(1), 103–120.

Holton, C. (2009). Identifying disgruntled employee systems fraud risk through text mining: A simple solution for a multi-billion dollar problem. *Decision Support Systems*, 46(4), 853–864. https://doi.org/10.1016/j.dss.2008.11.013

Hosen, A., & Alfina, I. (2016). Aggregation of open data information using linked data: Case study education and job vacancy data in Jakarta. *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 579–584. https://doi.org/10.1109/ICACSIS.2016.7872725

House, R. J., Spangler, W. D., & Woycke, J. (1991). Personality and charisma in the US presidency: A psychological theory of leader effectiveness. *Administrative Science Quarterly,* 364–396.

Houvardas, J., & Stamatatos, E. (2006). N-gram feature selection for authorship identification. In J. Euzenat & J. Domingue (Eds.), *Artificial Intelligence: Methodology, Systems, and Applications* (Vol. 4183, pp. 77–86). Springer. https://doi.org/10.1007/11861461_10

How Much Does Email Marketing Cost in 2019? (2019). https://www.webfx.com/email-marketing-pricing.html

Hsieh, H.-F., & Shannon, S. E. (2005). Three Approaches to Qualitative Content Analysis. *Qualitative Health Research, 15*(9), 1277–1288. https://doi.

org/10.1177/1049732305276687

Hsu, C.-W., & Lin, C.-J. (2002). A comparison of methods for multiclass support vector machines. IEEE *Transactions on Neural Networks*, 13(2), 415–425. https://doi.org/10.1109/72.991427

Hu, J., Sun, X., Lo, D., & Li, B. (2015). Modeling the evolution of development topics using Dynamic Topic Models. *2015 IEEE 22nd International Conference on Software Analysis, Evolution and Reengineering (SANER)*, 3–12. https://doi.org/10.1109/SANER.2015.7081810

Hu, M., & Liu, B. (2004a). Mining and summarizing customer reviews. *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 168–177. http://dl.acm.org/citation.cfm?id=1014073

Hu, M., & Liu, B. (2004b). Mining opinion features in customer reviews. *AAAI,* 4, 755–760. http://www.aaai.org/Papers/AAAI/2004/AAAI04-119.pdf

Huang, H., Kvasny, L., Joshi, K. D., Trauth, E. M., & Mahar, J. (2009). Synthesizing IT Job Skills Identified in Academic Studies, Practitioner Publications and Job Ads. *Proceedings of the Special Interest Group on Management Information System's 47th Annual Conference on Computer Personnel Research,* 121–128. https://doi.org/10.1145/1542130.1542154

Huang, H., & Zhang, B. (2009). Text Segmentation. In L. LIU & M. T. ÖZSU (Eds.), *Encyclopedia of Database Systems* (pp. 3072–3075). Springer US. http://link.springer.com/referenceworkentry/10.1007/978-0-387-39940-9_421

Ikonomakis, M., Kotsiantis, S., & Tampakas, V. (2005). Text classification using machine learning techniques. *WSEAS Transactions on Computers, 4*(8), 966–974.

ILO. (2012). *International Standard Classification of Occupations.* ILO Publications. https://www.ilo.org/wcmsp5/groups/public/---dgreports/---dcomm/---publ/documents/publication/wcms_172572.pdf

Inmon, W. H. (1996). The data warehouse and data mining. *Communications of the ACM*, 39(11), 49–50.

Inmon, W. H., & Nesavich, A. (2007). *Tapping into Unstructured Data: Integrating Unstructured Data and Textual Analytics into Business Intelligence* (1 edition). Prentice Hall.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys (CSUR), 31*(3), 264–323.

James, C., Pappalardo, L., Sirbu, A., & Simini, F. (2018). Prediction of next career moves from scientific profiles. ArXiv:1802.04830 *[Physics, Stat].* http://arxiv.org/abs/1802.04830

Jeanneret, P. R., & Strong, M. H. (2003). Linking O*net Job Analysis Information to Job Requirement Predictors: An O*net Application. *Personnel Psychology*, 56(2), 465–492. https://doi.org/10.1111/j.1744-6570.2003.tb00159.x

Jiang, S., Pang, G., Wu, M., & Kuang, L. (2012). An improved K-nearest-neighbor algorithm for text categorization. *Expert Systems with Applications, 39*(1), 1503–1509.

Joachims, T. (1998). T*ext categorization with support vector machines: Learning with many relevant features.* Springer. http://link.springer.com/chapter/10.1007/

BFb0026683

Jolliffe, I. (2005). *Principal component analysi*s. Wiley Online Library. http://onlinelibrary.wiley.com/doi/10.1002/0470013192.bsa501/full

Joseph, D., Ang, S., & Slaughter, S. A. (2015). Turnover or turnaway? Competing risks analysis of male and female IT professionals' job mobility and relative pay gap. *Information Systems Research, 26*(1), 145–164.

Joseph, D., Boh, W. F., Ang, S., & Slaughter, S. A. (2012). The Career Paths Less (or More) Traveled: A Sequence Analysis of IT Career Histories, Mobility Patterns, and Career Success. *MIS Quarterly,* 36(2), 427–452. https://doi.org/10.2307/41703462

Kabanoff, B. (1997). Computers can read as well as count: Computer-aided text analysis in organizational research. *Journal of Organizational Behavior,* 18(S1), 507–511. https://doi.org/10.1002/(SICI)1099-1379(199711)18:1+<507::AID-JOB904>3.0.CO;2-0

Kalleberg, A. L., & Mouw, T. (2018). Occupations, Organizations, and Intragenerational Career Mobility. *Annual Review of Sociology*, 44(1), 283–303. https://doi.org/10.1146/annurev-soc-073117-041249

Kanaan, G., Al-Shalabi, R., Ghwanmeh, S., & Al-Ma'adeed, H. (2009). A comparison of text-classification techniques applied to Arabic text. *Journal of the American Society for Information Science and Technology, 60*(9), 1836–1844. https://doi.org/10.1002/asi.20832

Kao, A., & Poteet, S. R. (2007). *Natural Language Processing and Text Mining.* Springer Science & Business Media.

Khoo, A., Marom, Y., & Albrecht, D. (2006). Experiments with sentence classification. *Proceedings of the 2006 Australasian Language Technology Workshop*, 18–25. http://www.aclweb.org/anthology/U06-1#page=26

Kirkpatrick, S. A., Wofford, J. C., & Baum, J. R. (2002). Measuring motive imagery contained in the vision statement. *The Leadership Quarterly,* 13(2), 139–150. https://doi.org/10.1016/S1048-9843(02)00096-6

Klema, V., & Laub, A. J. (1980). The singular value decomposition: Its computation and some applications. *Automatic Control, IEEE Transactions* On, 25(2), 164–176.

Kloptchenko, A., Eklund, T., Karlsson, J., Back, B., Vanharanta, H., & Visa, A. (2004). Combining data and text mining techniques for analysing financial reports. *Intelligent Systems in Accounting, Finance & Management, 12*(1), 29–41. https://doi.org/10.1002/isaf.239

Kobayashi, V. B., Mol, S., & Kismihok, G. (2014). Labour Market Driven Learning Analytics. *Journal of Learning Analytics, 1*(3), 207–210. https://doi.org/10.18608/jla.2014.13.24

Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihok, G., & Den Hartog, D. N. (2017). *Text Classification for Organizational Research: A Tutorial* [Manuscript submitted for publication].

Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihók, G., & Den Hartog, D. N. (2017). *Text Mining in Organizational Research.*

Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihók, G., & Den Hartog, D. N. (2018a).

Text Classification for Organizational Researchers: A Tutorial. *Organizational Research Methods, 21*(3), 766–799. https://doi.org/10.1177/1094428117719322

Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihók, G., & Den Hartog, D. N. (2018b). Text Mining in Organizational Research. *Organizational Research Methods, 21*(3), 733–765. https://doi.org/10.1177/1094428117722619

Kobayashi, V. B., Mol, S. T., Kismihók, G., & Hesterberg, M. (2016). Automatic Extraction of Nursing Tasks from Online Job Vacancies. In M. Fathi, M. Khobreh, & F. Ansari (Eds.), *Professional Education and Training through Knowledge, Technology and Innovation* (pp. 51–56). UniPrint - University of Siegen. http://dokumentix.ub.uni-siegen.de/opus/volltexte/2016/1057/pdf/Professional_education_and_training.pdf#page=58

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *Ijcai*, 14, 1137–1145. http://frostiebek.free.fr/docs/Machine%20Learning/validation-1.pdf

Kolen, J. F., & Pollack, J. B. (1990). Back Propagation is Sensitive to Initial Conditions. *Proceedings of the 1990 Conference on Advances in Neural Information Processing Systems 3*, 860–867. http://dl.acm.org/citation.cfm?id=118850.119960

Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). Text classification algorithms: A survey. *Information, 10*(4), 150.

Krippendorff, K. (2012). *Content Analysis: An Introduction to Its Methodology.* SAGE Publications.

Kulkarni, V. G. (2016). *Modeling and analysis of stochastic systems.* Chapman and Hall/CRC.

Kumar, S. K. (2017). On weight initialization in deep neural networks. *ArXiv Preprint ArXiv:1704.08863*.

Kureková, L., Beblavỳ, M., & Thum, A.-E. (2013). Online job vacancy data as a source for micro-level analysis of employers' preferences. A methodological enquiry. *First International Conference on Public Policy (ICPP)*.

Lan, M., Tan, C. L., Su, J., & Lu, Y. (2009). Supervised and Traditional Term Weighting Methods for Automatic Text Categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 31(4), 721–735. https://doi.org/10.1109/TPAMI.2008.110

Landauer, T. K., Foltz, P. W., & Laham, D. (1998a). An introduction to latent semantic analysis. *Discourse Processes,* 25(2–3), 259–284.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998b). An introduction to latent semantic analysis. *Discourse Processes*, 25(2–3), 259–284. https://doi.org/10.1080/01638539809545028

LaValle, S., Lesser, E., Shockley, R., Hopkins, M. S., & Kruschwitz, N. (2013). Big data, analytics and the path from insights to value. *MIT Sloan Management Review, 21*. http://sloanreview.mit.edu/article/big-data-analytics-and-the-path-from-insights-to-value/

Lee, J., & Hong, Y. S. (2013). Business model mining: Analyzing a firm's business model with text mining of annual report. *DS 75-3: Proceedings of the 19th International Conference on Engineering Design (ICED13) Design For Harmonies, Vol. 3: Design*

Organisation and Management, Seoul, Korea 19-22.08.2013.

Lee, S., Baker, J., Song, J., & Wetherbe, J. C. (2010). An Empirical Comparison of Four Text Mining Methods. *2010 43rd Hawaii International Conference on System Sciences (HICSS),* 1–10. https://doi.org/10.1109/HICSS.2010.48

Levy, O., Goldberg, Y., & Dagan, I. (2015). Improving Distributional Similarity with Lessons Learned from Word Embeddings. *Transactions of the Association for Computational Linguistics,* 3, 211–225. https://doi.org/10.1162/tacl_a_00134

Lewis, D. D. (1992a). Feature Selection and Feature Extraction for Text Categorization. *Proceedings of the Workshop on Speech and Natural Language,* 212–217. https://doi.org/10.3115/1075527.1075574

Lewis, D. D. (1992b). *Representation and learning in information retrieval* [University of Massachusetts]. http://ciir.cs.umass.edu/pubfiles/UM-CS-1991-093.pdf

Li, L., Jing, H., Tong, H., Yang, J., He, Q., & Chen, B.-C. (2017). NEMO: Next Career Move Prediction with Contextual Embedding. *Proceedings of the 26th International Conference on World Wide Web Companion,* 505–513. https://doi.org/10.1145/3041021.3054200

Li, Y. H., & Jain, A. K. (1998a). Classification of text documents. *The Computer Journal, 41*(8), 537–546.

Li, Y. H., & Jain, A. K. (1998b). Classification of Text Documents. *The Computer Journal, 41*(8), 537–546. https://doi.org/10.1093/comjnl/41.8.537

Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. *R News, 2*(3), 18–22.

Lindell, M. K., Clause, C. S., Brandt, C. J., & Landis, R. S. (1998). Relationship Between Organizational Context and Job Analysis Task Ratings. *Journal of Applied Psychology October 1998, 83*(5), 769–776.

Liu, H., & Motoda, H. (1998). *Feature Extraction, Construction and Selection: A Data Mining Perspective.* Springer Science & Business Media.

Lopez, F. M., Kesselman, G. A., & Lopez, F. E. (1981). An Empirical Test of a Trait-Oriented Job Analysis Technique. *Personnel Psychology, 34*(3), 479–502. https://doi.org/10.1111/j.1744-6570.1981.tb00490.x

Lustgarten, J. L., Gopalakrishnan, V., Grover, H., & Visweswaran, S. (2008). Improving classification performance with discretization on biomedical datasets. *AMIA Annual Symposium Proceedings, 2008,* 445.

Lyons, S. T., Schweitzer, L., & Ng, E. S. W. (2015). How have careers changed? An investigation of changing career patterns across four generations. *Journal of Managerial Psychology, 30*(1), 8–21. Scopus. https://doi.org/10.1108/JMP-07-2014-0210

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, 142–150. http://dl.acm.org/citation.cfm?id=2002491

Mainert, J., Niepel, C., Murphy, K. R., & Greiff, S. (2019). The incremental contribution of complex problem-solving skills to the prediction of job level, job complexity, and salary. *Journal of Business and Psychology, 34*(6), 825–845.

Mainiero, L. A., & Sullivan, S. E. (2005). Kaleidoscope Careers: An Alternate Explanation for the "Opt-out" Revolution. *The Academy of Management Executive (1993-2005)*, *19*(1), 106–123. JSTOR.

Malinowski, J., Keim, T., Weitzel, T., & Wendt, O. (2006). Matching People and Jobs: A Bilateral Recommendation Approach. *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06)(HICSS), 06*, 137c. https://doi.org/10.1109/HICSS.2006.266

Malinowski, J., Keim, T., Wendt, O., & Weitzel, T. (2006). Matching People and Jobs: A Bilateral Recommendation Approach. *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06)*, 6, 137c–137c. https://doi.org/10.1109/HICSS.2006.266

Mang, C. (2012). Online job search and matching quality. *Ifo Institute for Economic Research at the University of Munich. Ftp://Ftp. Zew. de/Pub/Zewdocs/Veranstaltungen/ICT2012/Papers/Mang.* Pdf. http://zinc.zew.de/pub/zew-docs/veranstaltungen/ICT2012/Papers/Mang.pdf

Marler, J. H., & Boudreau, J. W. (2017). An evidence-based review of HR Analytics. *The International Journal of Human Resource Management, 28*(1), 3–26. https://doi.org/10.1080/09585192.2016.1244699

McEntire, L. E., Dailey, L. R., Osburn, H. K., & Mumford, M. D. (2006). Innovations in job analysis: Development and application of metrics to analyze job data. *Human Resource Management Review, 16*(3), 310–323. https://doi.org/10.1016/j.hrmr.2006.05.004

McKenny, A. F., Short, J. C., & Payne, G. T. (2013). Using Computer-Aided Text Analysis to Elevate Constructs An Illustration Using Psychological Capital. *Organizational Research Methods, 16*(1), 152–184. https://doi.org/10.1177/1094428112459910

Méndez, J. R., Iglesias, E. L., Fdez-Riverola, F., Díaz, F., & Corchado, J. M. (2006). Tokenising, stemming and stopword removal on anti-spam filtering domain. *In Current Topics in Artificial Intelligence* (pp. 449–458). Springer. http://link.springer.com/chapter/10.1007/11881216_47

Messum, D., Wilkes, L., Peters, K., & Jackson, D. (2017). Content analysis of vacancy advertisements for employability skills: Challenges and opportunities for informing curriculum development. *Journal of Teaching and Learning for Graduate Employability, 7*(1), 72–86. https://doi.org/10.21153/jtlge2016vol7no1art582

Mihalcea, R., Corley, C., & Strapparava, C. (2006). Corpus-based and knowledge-based measures of text semantic similarity. *AAAI, 6,* 775–780. http://www.aaai.org/Papers/AAAI/2006/AAAI06-123.pdf

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *ArXiv Preprint ArXiv:1301.3781.* http://arxiv.org/abs/1301.3781

Mitchell, T. M. (1997). Machine learning. *Burr Ridge, IL: McGraw Hill, 45.*

Mobley, W. H., & Ramsay, R. S. (1973). Hierarchical Clustering on the Basis of Inter-Job Similarity as a Tool in Validity Generalization. *Personnel Psychology,* 26(2), 213–225.

Montanelli Jr, R. G., & Humphreys, L. G. (1976). Latent roots of random data correlation

matrices with squared multiple correlations on the diagonal: A Monte Carlo study. *Psychometrika, 41(3),* 341–348.

Morgeson, F. P. (2017). Job Analysis Methods. In S. G. Rogelberg (Ed.), *The SAGE Encyclopedia of Industrial and Organizational Psychology* (pp. 769–771).

Morgeson, F. P., & Campion, M. A. (1997). Social and Cognitive Sources of Potential Inaccuracy in Job Analysis. *Journal of Applied Psychology October 1997, 82*(5), 627–655.

Morgeson, F. P., & Campion, M. A. (2000). Accuracy in Job Analysis: Toward an Inference-Based Model. *Journal of Organizational Behavior, 21*(7), 819–827.

Morgeson, F. P., Delaney-Klinger, K., Mayfield, M. S., Ferrara, P., & Campion, M. A. (2004). Self-Presentation Processes in Job Analysis: A Field Experiment Investigating Inflation in Abilities, Tasks, and Competencies. *Journal of Applied Psychology August 2004, 89*(4), 674–686.

Morgeson, F. P., & Dierdorff, E. C. (2011). Work analysis: From technique to theory. In *APA handbook of industrial and organizational psychology, Vol 2: Selecting and developing members for the organization* (pp. 3–41). American Psychological Association. https://doi.org/10.1037/12170-001

Moschitti, A., & Basili, R. (2004). Complex Linguistic Features for Text Classification: A Comprehensive Study. *Advances in Information Retrieval,* 181–196. https://doi.org/10.1007/978-3-540-24752-4_14

Multilingual CV and job parsing. (n.d.). Textkernel. Retrieved August 12, 2019, from https://www.textkernel.com/hr-software/extract-cv-parsing/

Nenkova, A., & Bagga, A. (2003). Email Classification for Contact Centers. *Proceedings of the 2003 ACM Symposium on Applied Computing,* 789–792. https://doi.org/10.1145/952532.952689

Ng, T. W., & Feldman, D. C. (2010). The relationships of age with job attitudes: A meta-analysis. *Personnel Psychology, 63*(3), 677–718.

Ng, T. W. H., Sorensen, K. L., Eby, L. T., & Feldman, D. C. (2007). Determinants of job mobility: A theoretical integration and extension. *Journal of Occupational and Organizational Psychology,* 80(3), 363–386. https://doi.org/10.1348/096317906X130582

Nimczik, J. S. (2017). *Job mobility networks and endogenous labor markets.*

Nomden, K. (2012). *ESCO and the envisaged role of the European Taxonomy of Occupations, Qualifications, Skills and Competences for (Vocational) Education and Training.*

Ogura, H., Amano, H., & Kondo, M. (2011). Comparison of metrics for feature selection in imbalanced text classification. *Expert Systems with Applications, 38*(5), 4978–4989. https://doi.org/10.1016/j.eswa.2010.09.153

Olston, C., & Najork, M. (2010). Web crawling. Foundations and Trends in Information Retrieval, 4(3), 175–246.

Osinski, S., & Weiss, D. (2005). A Concept-Driven Algorithm for Clustering Search Results. *IEEE Intelligent Systems,* 20(3), 48–54. https://doi.org/10.1109/MIS.2005.38

Pacuk, A., Sankowski, P., Węgrzycki, K., Witkowski, A., & Wygocki, P. (2016). RecSys Challenge 2016: Job recommendations based on preselection of offers and

gradient boosting. *Proceedings of the Recommender Systems Challenge,* 10.

Palmer, D. (2010). Text Preprocessing. In N. Indurkhya & F. J. Damerau (Eds.), *Handbook of Natural Language Processing* (2nd ed., pp. 9–30). Chapman & Hall/CRC.

Pang, B., & Lee, L. (2008). Opinion Mining and Sentiment Analysis. *Foundations and Trends® in Information Retrieval, 2*(1–2), 1–135. https://doi.org/10.1561/1500000011

Panigrahi, P. K. (2012). A Comparative Study of Supervised Machine Learning Techniques for Spam E-mail Filtering. *2012 Fourth International Conference on Computational Intelligence and Communication Networks,* 506–512. https://doi.org/10.1109/CICN.2012.14

Parry, E., & Wilson, H. (2009). Factors influencing the adoption of online recruitment. *Personnel Review, 38*(6), 655–673. https://doi.org/10.1108/00483480910992265

Penn Part of Speech Tags. (n.d.). Retrieved March 15, 2015, from http://cs.nyu.edu/grishman/jet/guide/PennPOS.html

Pennebaker, J. W., Francis, M. E., & Booth, R. J. (2001). Linguistic Inquiry and Word Count (LIWC): A computerized text analysis program. *Mahwah (NJ), 7.*

Peterson, N. G., Mumford, M. D., Borman, W. C., Jeanneret, P. R., Fleishman, E. A., Levin, K. Y., Campion, M. A., Mayfield, M. S., Morgeson, F. P., Pearlman, K., Gowing, M. K., Lancaster, A. R., Silver, M. B., & Dye, D. M. (2001). Understanding Work Using the Occupational Information Network (o*net): Implications for Practice and Research. *Personnel Psychology, 54*(2), 451–492. https://doi.org/10.1111/j.1744-6570.2001.tb00100.x

Phan, X.-H., Nguyen, L.-M., & Horiguchi, S. (2008). Learning to Classify Short and Sparse Text & Web with Hidden Topics from Large-scale Data Collections. *Proceedings of the 17th International Conference on World Wide Web, 91–100.* https://doi.org/10.1145/1367497.1367510

Platzer, M., & Reutterer, T. (2016). Ticking away the moments: Timing regularity helps to better predict customer activity. *Marketing Science, 35*(5), 779–799.

Polikar, R. (2012). Ensemble Learning. In C. Zhang & Y. Ma (Eds.), *Ensemble Machine Learning* (pp. 1–34). Springer US. http://link.springer.com/chapter/10.1007/978-1-4419-9326-7_1

Popescu, A.-M., & Etzioni, O. (2007). Extracting Product Features and Opinions from Reviews. In A. Kao & S. R. Poteet (Eds.), *Natural Language Processing and Text Mining* (pp. 9–28). Springer London. https://doi.org/10.1007/978-1-84628-754-1_2

Popping, R. (2012). Qualitative Decisions in Quantitative Text Analysis Research. *Sociological Methodology,* 42, 88–90.

Porteous, I., Newman, D., Ihler, A., Asuncion, A., Smyth, P., & Welling, M. (2008). Fast collapsed gibbs sampling for latent dirichlet allocation. *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 569–577. http://dl.acm.org/citation.cfm?id=1401960

Porter, M. F. (1980). An algorithm for suffix stripping. *Program, 14*(3), 130–137. https://doi.org/10.1108/eb046814

Powers, D. M. (2011). *Evaluation: From precision, recall and F-measure to ROC,*

informedness, markedness and correlation. http://dspace2.flinders.edu.au/xmlui/handle/2328/27165

ProgrammableWeb. (n.d.). *API Directory.* ProgrammableWeb. Retrieved March 13, 2015, from http://www.programmableweb.com/apis/directory

R Programming/Text Processing. (2014, June 26). http://en.wikibooks.org/wiki/R_Programming/Text_Processing

Ragas, H., & Koster, C. H. (1998). Four text classification algorithms compared on a Dutch corpus. *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* 369–370. http://dl.acm.org/citation.cfm?id=291059

Raghavan, V. V., & Wong, S. M. (1986). A critical analysis of vector space model for information retrieval. *Journal of the American Society for Information Science, 37*(5), 279–287.

Ramraj, S., Saranya, S., & Yashwant, K. (2018). Comparative study of bagging, boosting and convolutional neural network for text classification. *Indian Journal of Public Health Research & Development, 9*(9), 1041–1047.

RANKS NL. (n.d.). *Stopwords.* RANKS NL. Retrieved March 13, 2015, from http://www.ranks.nl/stopwords

Rendón, E., Abundez, I., Arizmendi, A., & Quiroz, E. M. (2011). Internal versus external cluster validation indexes. *International Journal of Computers and Communications, 5*(1), 27–34.

Roberts, M. E., Stewart, B. M., Tingley, D., Lucas, C., Leder-Luis, J., Gadarian, S. K., Albertson, B., & Rand, D. G. (2014). Structural Topic Models for Open-Ended Survey Responses. *American Journal of Political Science, 5*8(4), 1064–1082.

Rokach, L., & Maimon, O. (2005). Top-down induction of decision trees classifiers—A survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 35*(4), 476–487. https://doi.org/10.1109/TSMCC.2004.843247

Rosenfeld, R. A. (1992). Job Mobility and Career Processes. *Annual Review of Sociology, 18*(1), 39–61. https://doi.org/10.1146/annurev.so.18.080192.000351

Rullo, P., Cumbo, C., & Policicchio, V. L. (2007). Learning Rules with Negation for Text Categorization. *Proceedings of the 2007 ACM Symposium on Applied Computing*, 409–416. https://doi.org/10.1145/1244002.1244098

Russ, D. E., Ho, K.-Y., Colt, J. S., Armenti, K. R., Baris, D., Chow, W.-H., Davis, F., Johnson, A., Purdue, M. P., Karagas, M. R., Schwartz, K., Schwenn, M., Silverman, D. T., Johnson, C. A., & Friesen, M. C. (2016). Computer-based coding of free-text job descriptions to efficiently identify occupations in epidemiological studies. *Occupational and Environmental Medicine, 73*(6), 417–424. https://doi.org/10.1136/oemed-2015-103152

Sackett, P. R., & Laczo, R. M. (2003a). Job and Work Analysis. In *Handbook of Psychology.* John Wiley & Sons, Inc. http://onlinelibrary.wiley.com/doi/10.1002/0471264385.wei1202/abstract

Sackett, P. R., & Laczo, R. M. (2003b). Job and Work Analysis. In *Handbook of Psychology.* John Wiley & Sons, Inc. http://onlinelibrary.wiley.com/doi/10.1002/0471264385.wei1202/abstract

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management, 24*(5), 513–523.

Salton, G., Wong, A., & Yang, C. S. (1975a). *A Vector Space Model for Automatic Indexing. Commun. ACM, 18*(11), 613–620. https://doi.org/10.1145/361219.361220

Salton, G., Wong, A., & Yang, C.-S. (1975b). A vector space model for automatic indexing. *Communications of the ACM,* 18(11), 613–620.

Sanchez, J. I., & Levine, E. L. (2000a). Accuracy or Consequential Validity: Which Is the Better Standard for Job Analysis Data? *Journal of Organizational Behavior, 21*(7), 809–818.

Sanchez, J. I., & Levine, E. L. (2000b). Accuracy or Consequential Validity: Which Is the Better Standard for Job Analysis Data? *Journal of Organizational Behavior, 21*(7), 809–818.

Sanchez, J. I., & Levine, E. L. (2012a). The Rise and Fall of Job Analysis and the Future of Work Analysis. *Annual Review of Psychology, 63*(1), 397–425. https://doi.org/10.1146/annurev-psych-120710-100401

Sanchez, J. I., & Levine, E. L. (2012b). The Rise and Fall of Job Analysis and the Future of Work Analysis. *Annual Review of Psychology, 63*(1), 397–425. https://doi.org/10.1146/annurev-psych-120710-100401

Schapire, R. E. (2003). The Boosting Approach to Machine Learning: An Overview. In D. D. Denison, M. H. Hansen, C. C. Holmes, B. Mallick, & B. Yu (Eds.), *Nonlinear Estimation and Classification* (pp. 149–171). Springer New York. https://doi.org/10.1007/978-0-387-21579-2_9

Scharkow, M. (2013). Thematic content analysis using supervised machine learning: An empirical evaluation using German online news. *Quality & Quantity, 47*(2), 761–773. https://doi.org/10.1007/s11135-011-9545-7

Scherbaum, C. A. (2005). Synthetic validity: Past, present, and future. *Personnel Psychology, 58*(2), 481–515.

Schlee, R. P., & Karns, G. L. (2017). Job requirements for marketing graduates: Are there differences in the knowledge, skills, and personal attributes needed for different salary levels? *Journal of Marketing Education, 39*(2), 69–81.

Schömann, K., & O'Connell, P. J. (2002). *Education, Training and Employment Dynamics.* Edward Elgar Publishing. https://www.elgaronline.com/view/9781840642780.xml#page=376

Scott, S., & Matwin, S. (1999a). Feature Engineering for Text Classification. *Proceedings of the Sixteenth International Conference on Machine Learning,* 379–388. http://dl.acm.org/citation.cfm?id=645528.657484

Scott, S., & Matwin, S. (1999b). Feature engineering for text classification. *ICML, 99,* 379–388. http://comp.mq.edu.au/units/comp348/reading/scott99feature.pdf

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys (CSUR), 34*(1), 1–47.

Segers, J., Inceoglu, I., Vloeberghs, D., Bartram, D., & Henderickx, E. (2008). Protean and boundaryless careers: A study on potential motivators. *Journal of Vocational Behavior, 73*(2), 212–230. https://doi.org/10.1016/j.jvb.2008.05.001

Settles, B. (2010). Active learning literature survey. *University of Wisconsin, Madison,*

52(55–66), 11.

Sharma, A., & Sharma, T. (2017). HR analytics and performance appraisal system: A conceptual framework for employee performance improvement. *Management Research Review, 40*(6), 684–697. https://doi.org/10.1108/MRR-04-2016-0084

Shen, J., Brdiczka, O., & Liu, J. (2013). Understanding Email Writers: Personality Prediction from Email Messages. *User Modeling, Adaptation, and Personalization,* 318–330. https://doi.org/10.1007/978-3-642-38844-6_29

Sheng, V. S., Provost, F., & Ipeirotis, P. G. (2008). Get Another Label? Improving Data Quality and Data Mining Using Multiple, Noisy Labelers. *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 614–622. https://doi.org/10.1145/1401890.1401965

Short, J. C., Broberg, J. C., Cogliser, C. C., & Brigham, K. H. (2010). Construct Validation Using Computer-Aided Text Analysis (CATA) An Illustration Using Entrepreneurial Orientation. *Organizational Research Methods, 13*(2), 320–347. https://doi.org/10.1177/1094428109335949

Shreve, J., Schneider, H., & Soysal, O. (2011). A methodology for comparing classification methods through the assessment of model stability and validity in variable selection. *Decision Support Systems, 52*(1), 247–257. https://doi.org/10.1016/j.dss.2011.08.001

Sicherman, N., & Galor, O. (1990). A Theory of Career Mobility. *Journal of Political Economy, 98*(1), 169–192.

Siddique, C. M. (2004). Job analysis: A strategic human resource management practice. *The International Journal of Human Resource Management, 15*(1), 219–244. https://doi.org/10.1080/0958519032000157438

Singh, N., Hu, C., & Roehl, W. S. (2007). Text mining a decade of progress in hospitality human resource management research: Identifying emerging thematic development. I*nternational Journal of Hospitality Management, 26*(1), 131–147. https://doi.org/10.1016/j.ijhm.2005.10.002

Singh, P. (2008). Job analysis for a changing workplace. *Human Resource Management Review, 18*(2), 87–99. https://doi.org/10.1016/j.hrmr.2008.03.004

Sirbu, D., Secui, A., Dascalu, M., Crossley, S. A., Ruseti, S., & Trausan-Matu, S. (2016). Extracting Gamers' Opinions from Reviews. *2016 18th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC),* 227–232. https://doi.org/10.1109/SYNASC.2016.044

Smith, D., & Ali, A. (2014). Analyzing Computer Programming Job Trend Using Web Data Mining. *Issues in Informing Science and Information Technology, 11.* http://iisit.org/Vol11/IISITv11p203-214Smith0494.pdf

Snell, D., Gatt, K., & Gekara, V. (2016). *Cross-occupational skill transferability: Challenges and opportunities in a changing economy.*

Sobotka, D., Leung, K. L., Inn, Y. J., & Tokuda, L. (1993). *Method and apparatus for automatic categorization of applicants from resumes* (United States Patent No. US5197004A). https://patents.google.com/patent/US5197004A/en

Sodhi, M. S., & Son, B.-G. (2007). *Industry Requirements of Operations Research Skills Based on Statistical Content Analysis of Job Ads* (SSRN Scholarly Paper ID 1011468).

Social Science Research Network. http://papers.ssrn.com/abstract=1011468

Sodhi, M. S., & Son, B.-G. (2010). Content analysis of OR job advertisements to infer required skills. *Journal of the Operational Research Society, 61*(9), 1315–1327. https://doi.org/10.1057/jors.2009.80

Solka, J. L. (2008). Text Data Mining: Theory and Methods. *Statistics Surveys, 2*(0), 94–112. https://doi.org/10.1214/07-SS016

Song, F., Liu, S., & Yang, J. (2005). A comparative study on text representation schemes in text categorization. *Pattern Analysis and Applications,* 8(1–2), 199–209. https://doi.org/10.1007/s10044-005-0256-3

Sørensen, A. B., & Tuma, N. B. (1978). *Labor market structures and job mobility.* University of Wisconsin–Madison.

Sousa-Poza, A., & Henneberger, F. (2004). Analyzing Job Mobility with Job Turnover Intentions: An International Comparative Study. *Journal of Economic Issues, 38*(1), 113–137. https://doi.org/10.1080/00213624.2004.11506667

Steinbach, M., Karypis, G., & Kumar, V. (2000). A comparison of document clustering techniques. *KDD Workshop on Text Mining, 400,* 525–526. https://wwws.cs.umn.edu/tech_reports_upload/tr2000/00-034.ps

Sullivan, S. E., & Baruch, Y. (2009). Advances in Career Theory and Research: A Critical Review and Agenda for Future Exploration. *Journal of Management, 35*(6), 1542–1571. https://doi.org/10.1177/0149206309350082

Sullivan, S. E., & Mainiero, L. (2007). Women's kaleidoscope careers: A new framework for examining women's stress across the lifespan. In *Exploring the work and non-work interface* (pp. 205–238). Emerald Group Publishing Limited.

Sullivan, S., Forret, M., Carraher, S., & Mainiero, L. (2009). Using the Kaleidoscope Career Model to Examine Generational Differences in Work Attitudes. *Career Development International.* https://digitalcommons.fairfield.edu/business-facultypubs/62

Super, D. E. (1980). A life-span, life-space approach to career development. *Journal of Vocational Behavior, 16*(3), 282–298. https://doi.org/10.1016/0001-8791(80)90056-1

Super, D., Savickas, M., & Super, S. (1996). The life-span, life-space approach to careers. In D. Brown & L. Brooks (Eds.), *Career choices and development* (3rd ed., pp. 121–178). Jossey-Bass.

Taylor, L. R. (1978). Empirically Derived Job Families as a Foundation for the Study of Validity Generalization. *Personnel Psychology, 31*(2), 325–340. https://doi.org/10.1111/j.1744-6570.1978.tb00450.x

Tett, R. P., Guterman, H. A., Bleier, A., & Murphy, P. J. (2005). Development and Content Validation of a Hyperdimensional Taxonomy of Managerial Competence. *Human Performance, 13*(3), 205–251.

Thangaraj, M., & Sivakami, M. (2018). TEXT CLASSIFICATION TECHNIQUES: A LITERATURE REVIEW. *Interdisciplinary Journal of Information, Knowledge & Management, 13.*

Theeboom, T., Van Vianen, A. E. M., Beersma, B., Zwitser, R., & Kobayashi, V. (2017). A practitioner's perspective on coaching effectiveness. In L. Nota & S. Soresi

(Eds.), *Counseling and coaching in times of crisis and transitions: From research to practice.* Routledge Publishers.

Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., & Kappas, A. (2010). Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology, 61*(12), 2544–2558.

Thompson, L. F., Braddy, P. W., & Wuensch, K. L. (2008). E-recruitment and the benefits of organizational web appeal. *Computers in Human Behavior, 24*(5), 2384–2398. https://doi.org/10.1016/j.chb.2008.02.014

Toman, M., Tesar, R., & Jezek, K. (2006). Influence of word normalization on text classification. *Proceedings of InSciT,* 354–358. http://www.kiv.zcu.cz/research/groups/text/publications/inscit20060710.pdf

Torunoğlu, D., Çakırman, E., Ganiz, M. C., Akyokuş, S., & Gürbüz, M. Z. (2011). Analysis of preprocessing methods on classification of Turkish texts. *Innovations in Intelligent Systems and Applications (INISTA), 2011* International Symposium On, 112–117. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5946084

Tsuge, S., Shishibori, M., Kuroiwa, S., & Kita, K. (2001). Dimensionality reduction using non-negative matrix factorization for information retrieval. *2001 IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace (Cat.No.01CH37236), 2,* 960–965 vol.2. https://doi.org/10.1109/ICSMC.2001.973042

Turney, P. (1999). *Learning to extract keyphrases from text.* http://nparc.cisti-icist.nrc-cnrc.gc.ca/npsi/ctrl?action=rtdoc&an=8913245

Uysal, A. K., & Gunal, S. (2014). The impact of preprocessing on text classification. *Information Processing & Management, 50*(1), 104–112. https://doi.org/10.1016/j.ipm.2013.08.006

van der Maaten, L. J., Postma, E. O., & van den Herik, H. J. (2009). Dimensionality reduction: A comparative review. *Journal of Machine Learning Research, 10*(1–41), 66–71.

Van Wel, L., & Royakkers, L. (2004). Ethical issues in web data mining. E*thics and Information Technology, 6*(2), 129–140.

Verwaeren, B., Van Hoye, G., & Baeten, X. (2016). Getting bang for your buck: The specificity of compensation and benefits information in job advertisements. *The International Journal of Human Resource Management, 1–20.*

Vicknair, J., Elkersh, D., Yancey, K., & Budden, M. C. (2010). The Use of Social Networking Websites as a Recruiting Tool for Employers. *American Journal of Business Education, 3*(11), 7–12.

Vo, D.-T., & Ock, C.-Y. (2015). Learning to classify short text from scientific documents using topic models with various types of knowledge. *Expert Systems with Applications, 42*(3), 1684–1698. https://doi.org/10.1016/j.eswa.2014.09.031

Voigt, M., & Ruppert, A. (2019). Salary Negotiations: Highlights and Surprises From 10 Years of Research. *International Conference on Gender Research,* 639–645.

Voskuijl, O. (2005). Job Analysis: Current and Future Perspectives. In A. Evers, N. Anderson, & O. Voskuijl (Eds.), *The Blackwell Handbook of Personnel Selection* (pp. 27–47). Blackwell.

Voutilainen, A. (2003). Part-of-speech tagging. *The Oxford Handbook of Computational Linguistics,* 219–232.

Wallach, H. M., Murray, I., Salakhutdinov, R., & Mimno, D. (2009). Evaluation methods for topic models. *Proceedings of the 26th Annual International Conference on Machine Learning,* 1105–1112. http://dl.acm.org/citation.cfm?id=1553515

Wang, L., & Chen, Y. (2008). Conversation Extraction in Dynamic Text Message Stream. *Journal of Computers, 3*(10), 86–93.

Wegener, B. (1991). Job Mobility and Social Ties: Social Resources, Prior Job, and Status Attainment. *American Sociological Review, 56*(1), 60–71. https://doi.org/10.2307/2095673

Weiss, S. M., Indurkhya, N., & Zhang, T. (2015). *Fundamentals of Predictive Text Mining* (2nd ed.). Springer-Verlag London. https://doi.org/10.1007/978-0-387-34555-0_2

Wiebe, J., Wilson, T., Bruce, R., Bell, M., & Martin, M. (2004). Learning Subjective Language. *Computational Linguistics, 30*(3), 277–308. https://doi.org/10.1162/0891201041850885

Wiebe, J., Wilson, T., & Cardie, C. (2005). Annotating Expressions of Opinions and Emotions in Language. *Language Resources and Evaluation, 39*(2–3), 165–210. https://doi.org/10.1007/s10579-005-7880-9

Willett, P. (2006). The Porter stemming algorithm: *Then and now. Program, 40*(3), 219–223. https://doi.org/10.1108/00330330610681295

Xu, H., Yu, Z., Xiong, H., Guo, B., & Zhu, H. (2015). Learning career mobility and human activity patterns for job change analysis. *2015 IEEE International Conference on Data Mining,* 1057–1062.

Yamnill, S., & McLean, G. N. (2001). Theories supporting transfer of training. *Human Resource Development Quarterly, 12*(2), 195–208. https://doi.org/10.1002/hrdq.7

Yang, Y. (1999). An evaluation of statistical approaches to text categorization. *Information Retrieval, 1*(1–2), 69–90.

Yang, Y., & Pedersen, J. O. (1997a). A Comparative Study on Feature Selection in Text Categorization. *Proceedings of the Fourteenth International Conference on Machine Learning,* 412–420. http://dl.acm.org/citation.cfm?id=645526.657137

Yang, Y., & Pedersen, J. O. (1997b). A comparative study on feature selection in text categorization. *ICML,* 97, 412–420. http://www.surdeanu.info/mihai/teaching/ista555-spring15/readings/yang97comparative.pdf

Yarkoni, T. (2010). Personality in 100,000 Words: A large-scale analysis of personality and word use among bloggers. *Journal of Research in Personality, 44*(3), 363–373. https://doi.org/10.1016/j.jrp.2010.04.001

Youn, S., & McLeod, D. (2007). A comparative study for email classification. In K. Elleithy (Ed.), *Advances and Innovations in Systems, Computing Sciences and Software Engineering* (pp. 387–391). Springer. http://link.springer.com/chapter/10.1007/978-1-4020-6264-3_67

Yu, B., Kaufmann, S., & Diermeier, D. (2008). Classifying Party Affiliation from Political Speech. *Journal of Information Technology & Politics, 5*(1), 33–48. https://doi.org/10.1080/19331680802149608

Yu, C. H., Jannasch-Pennell, A., & DiGangi, S. (2011). Compatibility between text mining and qualitative research in the perspectives of grounded theory, content analysis, and reliability. *The Qualitative Report, 16*(3), 730.

Yu, H.-F., Ho, C.-H., Arunachalam, P., Somaiya, M., & Lin, C.-J. (2012). *Product Title Classification versus Text Classification* [Technical]. http://www.csie.ntu.edu.tw/~cjlin/papers/title.pdf

Zenko, B., Todorovski, L., & Dzeroski, S. (2001). A comparison of stacking with meta decision trees to bagging, boosting, and stacking with other methods. *Proceedings 2001 IEEE International Conference on Data Mining,* 669–670. https://doi.org/10.1109/ICDM.2001.989601

Zhang, J., Jin, R., Yang, Y., & Hauptmann, A. (2003). Modified logistic regression: An approximation to SVM and its applications in large-scale text categorization. *ICML*, 888–895. http://www.aaai.org/Papers/ICML/2003/ICML03-115.pdf

Zhang, W., Yoshida, T., & Tang, X. (2008). Text classification based on multi-word with support vector machine. *Knowledge-Based Systems, 21*(8), 879–886. https://doi.org/10.1016/j.knosys.2008.03.044

Zhang, X., & LeCun, Y. (2015). Text understanding from scratch. *ArXiv Preprint ArXiv:1502.01710.* http://arxiv.org/abs/1502.01710

Zhang, Y., Chen, M., & Liu, L. (2015). A review on text mining. *2015 6th IEEE International Conference on Software Engineering and Service Science (ICSESS),* 681–685. https://doi.org/10.1109/ICSESS.2015.7339149

Zhou, W., Zhu, Y., Javed, F., Rahman, M., Balaji, J., & McNair, M. (2016). Quantifying skill relevance to job titles. *Big Data (Big Data), 2016 IEEE International Conference On*, 1532–1541.

Zhu, X. (2005). *Semi-supervised learning literature survey.* http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.99.9681&rep=rep1&type=pdf

Zhu, X., & Ghahramani, Z. (2002). *Learning from labeled and unlabeled data with label propagation.* Citeseer. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.14.3864&rep=rep1&type=pdf

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67*(2), 301–320.

Zu, G., Ohyama, W., Wakabayashi, T., & Kimura, F. (2003). Accuracy Improvement of Automatic Text Classification Based on Feature Transformation. *Proceedings of the 2003 ACM Symposium on Document Engineering,* 118–120. https://doi.org/10.1145/958220.958242

Zurada, J. M., Ensari, T., Asl, E. H., & Chorowski, J. (2013). Nonnegative Matrix Factorization and its application to pattern analysis and text mining. *2013 Federated Conference on Computer Science and Information Systems (FedCSIS),* 11–16.

Summary

Organizational researchers collect and generate text data. Since text data are abundant, and likely to contain rich and detailed information (as compared to numerical data), they may contribute to the understanding of organizations. Texts capture details of context and may reveal the nuances of the individuals and the organization itself compared to more reductionistic quantitative approaches. Analyzing sizable amounts of texts from varied sources may eliminate bias, permit triangulation, and enhance the validity of research outcomes. In this dissertation, the analysis of text is used to generate knowledge about jobs and people (e.g., what people need to know to be able to perform their jobs) that may serve human resource (HR) practices, such as recruitment, training, and development (Arthur Jr et al., 2003; Schömann & O'Connell, 2002; Yamnill & McLean, 2001).

In the field of organizational research, qualitative approaches had been used to analyze text data. However, the voluminous text data that exist about HR issues and related topics make the qualitative analysis tedious, impractical, and inadequate. This dissertation aims to address research questions relevant to organizational research using text analytics, and to furnish new analytical strategies that can be useful for job analysis, career, and HR research in general. The primary goal is to extract information to produce deeper insights about jobs and people. These insights have implications not only for making practical HR decisions but also for advancing knowledge in the HR field. This dissertation can contribute to praxis in HR field such as career planning, fitting jobs to candidates, candidate sourcing, career path, and job mobility).

Since organizational research is a broad area (including job analysis and career research) that cuts across many disciplines, the ***first chapter*** of this dissertation discusses the motivation of employing text mining methods and proposes a framework for text analysis in this realm. This framework may guide researchers in taking a principled approach to applying text analytics to actual research. We thus set out to elucidate the text mining process to organizational researchers with the purpose of popularizing text mining and empowering organizational researchers to analyze large corpora of text. The complexities of organizational research need a combination of different data, enabling the aggregation and triangulation of results. Additionally, this chapter highlights opportunities for text analysis-based research, explains the text mining process in detail, and discusses specific techniques for each problem or task.

The ***second chapter*** provides a state-of-the-art review of, and tutorial on, conducting text classification, as it is one of the important methods in many prediction

tasks. Indeed, many problems in organizational research, such as classifying employee performance and classifying the sentiments of employees into positive or negative, can be treated as classification problems. The tutorial further illustrates how text classification can be used in a nursing job analysis. Specifically, an algorithm was proposed that can automatically extract tasks from nursing vacancies and validate the results by comparing the extracted tasks to tasks obtained from conducting observation and subject matter expert interviews. This chapter and the first have already been published and are well cited which is evidence of the uptick in interest among organizational researchers regarding text mining.

Vacancy notices are posted to various job boards on a daily basis, and they contain information about jobs including roles, responsibilities, and tasks. They also provide information about the qualifications of potential candidates, such as knowledge, skills, and attitudes required to effectively perform a particular job. Company information, compensation scheme, type of employment, and contact information are also found in the notices. The main purpose of the job vacancies is to advertise a job to attract candidates, but the information contained therein could also be used for job analysis. The primary challenge is to disentangle the information relevant for job analysis. The ***third chapter*** demonstrates the building of classification models for automating the extraction of job information from job vacancies. Through the classification model we were able to obtain job specific information applicable to understanding the skill requirements and tasks of jobs. Moreover, the output of the model can be used to automatically build a skills taxonomy and to cluster jobs based on the similarity of the skills that comprise them. Consequently, applying the classification models ensures that we can analyze vacancies and extract job information in a shorter period. This contrasts with traditional job analysis, which is time consuming and painstakingly laborious. Moreover, the results might not be timely since many jobs are dynamic, such as in the information technology (IT) industry where technology and tools are continuously updated), and new jobs appear regularly. Finally, this chapter managed to show the benefits of using vacancy data because job information can be obtained across different organizations and geographical areas and the information can be continuously updated.

The ***fourth chapter*** demonstrates the applicability of text analytics to career research. The introduction of text analysis to organizational research can be expected to expand the available tools used by researchers in this field and to exemplify new avenues of research using text data. Moreover, we report on the building of a model that can predict salary from vacancies. The model can be examined to understand differences in salary structures among jobs and, by investigating further, we can determine the factors that drive such differences. By mapping job characteristics (i.e.,

industry, role, skill requirement, and location), the model shed light on pay differences and jobs that tend to pay more. The model showed that for example, managerial and financial roles tend to pay more and jobs that explicitly mention "communication skill" in the vacancies tend to pay low.

Lastly, the ***fifth chapter*** reports on the extraction of job histories from resumes that were used to create models that can predict the timing of job change and the possible next job. Such models may help career researchers determine factors that propel people to change jobs. Also, the models can help recruiters predict the timing of job change, so that they can approach the candidate who is most likely to accept a job offer. By investigating the models, we acquired an understanding of career progression, job mobility, and timing of turnover. In candidate sourcing, the goal is not only to find the best matching candidate but also to find candidates who are ready to switch jobs. Finally, using the patterns in job changes we can reconstruct job similarity based entirely on job history which has clear implications for job analysis.

In summary, we managed to achieve the following: demonstrate the applicability of text mining techniques in organizational research, create models that are of interest to job analysts and career researchers, derive knowledge from the models that can be useful for theory validation and building, and demonstrate the practicability of the models to job analysis, recommendation, and recruitment.

Samenvatting

Organisatieonderzoekers verzamelen en genereren tekstgegevens. Aangezien tekstgegevens in overvloed aanwezig zijn, en vaak rijke en gedetailleerde informatie bevatten (in vergelijking met numerieke gegevens), kunnen zij bijdragen aan ons begrip van organisaties. Teksten omvatten details van de context en kunnen de nuances van zowel individuen als de organisatie zelf onthullen in vergelijking met meer reductionistische kwantitatieve onderzoeksmethoden. Het analyseren van grote hoeveelheden tekst uit verschillende bronnen kan vooringenomenheid wegnemen, triangulatie mogelijk maken, en de geldigheid van onderzoeksresultaten vergroten. In dit proefschrift wordt tekstanalyse (of 'tekst mining') gebruikt om nieuwe inzichten te genereren over banen en mensen (bijv. wat mensen moeten weten om hun baan uit te kunnen voeren) die van nut kunnen zijn voor human resource (HR) praktijken, zoals werving, training en ontwikkeling (Arthur Jr et al., 2003; Schömann & O'Connell, 2002; Yamnill & McLean, 2001).

Op het gebied van organisatieonderzoek worden kwalitatieve benaderingen regelmatig gebruikt om tekstgegevens te analyseren. Echter, de volumineuze tekstdata die bestaan over HR-vraagstukken en gerelateerde onderwerpen maken de kwalitatieve analyse langdradig, onpraktisch en ontoereikend. Deze dissertatie heeft als doel om met behulp van tekstanalyse onderzoeksvragen te beantwoorden die relevant zijn voor organisatieonderzoek, en nieuwe analytische strategieën aan te reiken die nuttig kunnen zijn voor functieanalyse, carrière-, en HR-onderzoek in het algemeen. Het hoofddoel is om informatie te extraheren om diepere inzichten te verkrijgen over banen en mensen. Deze inzichten hebben niet alleen implicaties voor het nemen van praktische HR-beslissingen, maar ook voor het bevorderen van kennis in het HR-veld. Dit proefschrift hoopt bij te dragen aan de praktijk in het HR-veld, zoals carrièreplanning, het aanpassen van banen aan kandidaten, rekrutering, carrièrepad, en baan mobiliteit).

Omdat organisatieonderzoek (inclusief functie-analyse en loopbaanonderzoek) een breed gebied is dat vele disciplines doorsnijdt, bespreekt het eerste hoofdstuk van dit proefschrift de motivatie voor het gebruik van 'text mining' methoden en introduceert het een raamwerk voor tekstanalyse in dit domein voor. Dit raamwerk kan een leidraad zijn voor onderzoekers bij het kiezen van een principiële benadering voor het toepassen van tekstanalyse op feitelijk onderzoek. We hebben ons dus ten doel gesteld het text mining proces te verduidelijken voor organisatie-onderzoekers met het doel om text mining te populariseren en organisatie-onderzoekers in staat te stellen grote tekstcorpora te analyseren. De complexiteit

van organisatie-onderzoek resulteert vaak in een combinatie van gegevens uit verschillende bronnen, waardoor aggregatie en triangulatie van resultaten mogelijk wordt. Daarnaast belicht dit hoofdstuk mogelijkheden voor onderzoek gebaseerd op tekstanalyse, legt het het text mining proces in detail uit, en bespreekt het technieken voor specifieke problemen of taken die men in de praktijk kan tegenkomen.

Het tweede hoofdstuk geeft een state-of-the-art overzicht van, en gedetailleerde handleiding voor, het uitvoeren van tekstclassificatie, aangezien dit een van de belangrijkste methoden is in veel voorspellingstaken. Veel problemen in organisatieonderzoek, zoals het classificeren van de prestaties van werknemers of het classificeren van positieve dan wel negatieve sentimenten van werknemers, kunnen immers worden behandeld als classificatieproblemen. De handleiding illustreert verder hoe tekstclassificatie kan worden gebruikt in een verpleegkundige taakanalyse. Specifiek werd een algoritme voorgesteld dat automatisch taken uit verpleegkundige vacatures kan extraheren en de resultaten kan valideren door deze te vergelijken met taken verkregen door middel van observatie- en inhoudsdeskundige interviews. Dit hoofdstuk en het eerste zijn al gepubliceerd en worden goed geciteerd, wat bewijst dat de belangstelling onder organisatie-onderzoekers voor text mining toeneemt.

Vacatures worden dagelijks op verschillende vacaturebanken geplaatst, en ze bevatten informatie over banen, inclusief rollen, verantwoordelijkheden en taken. Ze bieden ook informatie over de kwalificaties van potentiële kandidaten, zoals de kennis, vaardigheden en attitudes die nodig zijn om een bepaalde functie effectief uit te voeren. Ook bedrijfsinformatie, vergoedingsregelingen, soort werk en contactinformatie zijn in de vacatures terug te vinden. Het belangrijkste doel van deze teksten is om een vacature bekend te maken teneinde kandidaten aan te trekken, maar de daarin vervatte informatie kan ook worden gebruikt voor functieanalyse. De belangrijkste uitdaging is om die informatie die relevant is voor functieanalyse te ontwarren uit de tekst, die vaak ook irrelevante informatie bevat. Het derde hoofdstuk demonstreert de constructie van classificatiemodellen voor het automatiseren van de extractie van relevante informatie uit vacatures. Door middel van het classificatiemodel zijn we in staat geweest om functie-specifieke informatie te verkrijgen die van toepassing is op het begrijpen van de vaardigheidseisen en taken van banen. Bovendien kan de output van het model worden gebruikt om automatisch een taxonomie van vaardigheden op te bouwen en om specifieke functies te clusteren op basis van de gelijkenis tussen de vaardigheden waaruit ze bestaan. Bijgevolg zorgt de toepassing van de classificatiemodellen ervoor dat we in een relatief korte periode vacatures kunnen analyseren en baan informatie kunnen extraheren. Dit in tegenstelling tot de traditionele functieanalyse, die vaak extreem tijdrovend en moeizaam is. Bovendien brengen wij verslag uit over de bouw van een model dat uit vacatures het salaris

kan voorspellen. Dit model kan worden gebruikt om verschillen in salarisstructuren tussen banen te begrijpen en, door verder onderzoek, kunnen we de factoren bepalen die dergelijke verschillen veroorzaken. Door de kenmerken van de functie in kaart te brengen (d.w.z. de bedrijfstak, de functie, de vereiste vaardigheden en de locatie), hebben we met dit model licht kunnen werpen op de loonverschillen en kenmerken van die functies die beter beloond worden. Het model toonde bijvoorbeeld aan dat leidinggevende en financiële functies meestal beter betalen en dat functies waarin expliciet "communicatievaardigheden" worden vermeld, vaak slechter worden betaald.

Het vijfde en laatste hoofdstuk rapporteert over de extractie van carrièrepaden uit cv's die vervolgens werden gebruikt om modellen te construeren die het tijdstip van baanverandering en de mogelijke daaropvolgende baan kunnen voorspellen. Dergelijke modellen kunnen loopbaanonderzoekers helpen om die factoren te bepalen die mensen ertoe aanzetten van baan te veranderen. Ook kunnen de modellen recruiters helpen het tijdstip van baanverandering te voorspellen, zodat ze die kandidaat kunnen benaderen die het meest waarschijnlijk een baanaanbod zal aanvaarden. Door de modellen te onderzoeken, hebben we inzicht gekregen in loopbaanprogressie, baanmobiliteit en timing van verloop. Bij de rekrutering van kandidaten is het doel niet alleen om de best passende kandidaat te vinden, maar ook om kandidaten te vinden die klaar zijn om van baan te wisselen. Tenslotte kunnen we, gebruikmakend van de patronen in baanwisselingen, functiegelijkenissen reconstrueren die volledig gebaseerd zijn op de carrièrepaden, wat duidelijke implicaties heeft voor functieanalyse.

Samenvattend hebben we de toepasbaarheid van text mining technieken in organisatieonderzoek aangetoond, modellen gemaakt die interessant kunnen zijn voor functieanalisten en loopbaanonderzoekers, kennis uit die modellen kunnen afleiden die nuttig kan zijn voor de validatie theorieën en theorievorming, en de bruikbaarheid van de modellen aangetoond voor functieanalyse, matching, en werving.

List of Authors

The empirical chapters presented in this dissertation were designed by V. Kobayashi, who acted as lead in study design, data collection, data analysis, and writing of the manuscript. S. T. Mol and G. Kismihók contributed to design of the empirical studies and to the writing of the manuscript by providing valuable comments. E. Kanoulas contributed to the design and data collection in the empirical studies presented within Chapter 3 and Chapter 4. All co-authors recognize V. Kobayashi as lead author of all the chapters reported in this dissertation.

TEXT ANALYTICS APPLICATIONS IN JOB ANALYSIS AND CAREER RESEARCH

Organizational researchers collect and generate text data. Since text data are abundant and likely to generate knowledge, they often reveal aspects of and therewith contribute to our understanding of organizations. As compared with more reductionistic quantitative approaches, text captures richer context and may reveal the nuances of individuals and organizations. Analyzing huge amounts of text from varied textual sources may eliminate bias, permit triangulation, and enhance the validity of the outcomes. Moreover, the analysis of text which come from myriad of sources are expected to generate knowledge about jobs and people (e.g., what people need to know to be able to perform their jobs) that are necessary to serve HR practices of recruitment, training, and development.

The book presents the development of models and/or analyses of large text data. The primary goal is to extract information and produce deeper insights about jobs and people. This book addresses research questions relevant to organizational research using text analytics and furnishing new analytical strategies that can be useful for job analysis, career, and HR research in general. The insights have implications not only for making practical HR decisions but also for advancing knowledge in the HR field (e.g., understanding career path and physical job mobility).

In summary, the contributions are the following: create models that are of interest primarily to job analysis and career researchers, derive knowledge from the models that can be useful for theory validation and building, and demonstrate the practicability of the models to job recommendation and recruitment.