TECHNISCHE UNIVERSITÄT DARMSTADT

Department of Computer Science
Ubiquitous Knowledge Processing Lab

Dissertation approved by the Department of Computer Science of the Technische Universität Darmstadt to attain the academic degree Doktor rerum naturalium (Dr. rer. nat.)

Leveraging Lexical-Semantic Knowledge for Text Classification Tasks

Ing. et Ing. Lucie Flekova
born in Prague, Czech Republic

Submitted: February 28, 2017
Defended: April 24, 2017

1. Reviewer Prof. Dr. Iryna Gurevych
Department of Computer Science
Technische Universität Darmstadt

2. Reviewer Prof. Dr. Benno Stein
Department of Computer Science and Media
Bauhaus Universität Weimar

3. Reviewer Prof. Dr. Walter Daelemans
CLiPS Research Center
University of Antwerp

Darmstadt, 2017
D17

Ing. et Ing. Lucie Flekova

Leveraging Lexical-Semantic Knowledge for Text Classification Tasks

Computer Science, Submitted: February 28, 2017

Reviewers: Prof. Dr. Iryna Gurevych, Prof. Dr. Benno Stein and Prof. Dr. Walter Daelemans

Technische Universität Darmstadt

Ubiquitous Knowledge Processing Lab

Department of Computer Science

Hochschulstrasse 10

64289 Darmstadt

Abstract

This dissertation is concerned with the applicability of knowledge, contained in lexical-semantic resources, to text classification tasks. Lexical-semantic resources aim at systematically encoding various types of information about the meaning of words and their relations. Text classification is the task of sorting a set of documents into categories from a predefined set, for example, "spam" and "not spam". With the increasing amount of digitized text, as well as the increased availability of the computing power, the techniques to automate text classification have witnessed a booming interest. The early techniques classified documents using a set of rules, manually defined by experts, e.g. computational linguists. The rise of big data led to the increased popularity of distributional hypothesis [Harris, 1954] - i.e., "a meaning of word comes from its context" - and to the criticism of lexical-semantic resources as too academic for real-world NLP applications [Jorgensen, 1990, Palmer, 2000, Ide and Wilks, 2007]. For long, it was assumed that the lexical-semantic knowledge will not lead to better classification results, as the meaning of every word can be directly learned from the document itself [Resnik, 2006]. In this thesis, we show that this assumption is not valid as a general statement and present several approaches how lexicon-based knowledge will lead to better results. Moreover, we show why these improved results can be expected.

One of the first problems in natural language processing is the lexical-semantic ambiguity [Jurafsky and Martin, 2009]. In text classification tasks, the ambiguity problem has often been neglected. For example, to classify a topic of a document containing the word *bank*, we don't need to explicitly disambiguate it, if we find the word *river* or *finance*. However, such additional word may not be always present. Conveniently, lexical-semantic resources typically enumerate all senses of a word, letting us choose which word sense is the most plausible in our context. What if we use the knowledge-based sense disambiguation methods in addition to the information provided implicitly by the word context in the document? In this thesis, we evaluate the performance of selected resource-based word sense disambiguation algorithms on a range of document classification tasks (Chapter 3). We note that the lexicographic sense distinctions provided by the lexical-semantic resources are not always optimal for every text classification task, and propose an alternative technique for disambiguation of word meaning in its context for sentiment analysis applications.

The second problem in text classification, and natural language processing in general, is the one with synonymy. The words used in training documents represent only a tiny fraction of the words in the total possible vocabulary. If we learn individual words, or senses, as features in the classification model, our system will not be able to interpret the paraphrases, where the synonymous meaning is conveyed using different expressions. How much would the classification performance improve if the system could determine that two very different words represent the same meaning? In this thesis, we propose to

address the synonymy problem by automatically enriching the training and testing data with conceptual annotations accessible through lexical-semantic resources (Chapter 4). We show that such conceptual information ("supersenses"), in combination with the previous word sense disambiguation step, helps to build more robust classifiers and improves classification performance of multiple tasks (Chapter 5). We further circumvent the sense disambiguation step by training a supersense tagging model directly. Previous evidence suggests that the sense distinctions of expert lexical-semantic resources are far subtler than what is needed for downstream NLP applications [Ide and Wilks, 2007, Jorgensen, 1990], and by disambiguating the concepts directly on a supersense level (e.g., "is the *duck* an animal or a food?" rather than choosing between its eight WordNet senses), we can reduce the number of errors.

The third problem in text classification is the curse of dimensionality. We want to know not only if each single word predicts certain document class, but which combinations of words predict it and which ones do not. Our need for training data thus grows exponentially with the number of words monitored. Several techniques for dimensionality reduction were proposed, most recently the representation learning, producing continuous word representations in a dense vector space, also known as word embeddings. However, these vectors are again produced on an ambiguous word level, and the valuable piece of information about possible distinct senses of the same word is lost, in favor of the most frequent one(s). In this thesis, we explore if, or how, we can use lexical-semantic resources to regain the sense-level notion of semantic relatedness back while operating within the deep learning paradigm, therefore still being able to access the high-level conceptual information. We propose and evaluate a method to integrate word and supersense embeddings from large sense-disambiguated resources such as Wikipedia. We examine the impact of different training data for the quality of these embeddings, and demonstrate how to employ them in deep learning text classification experiments. Using convolutional and recurrent neural networks, we achieve a significant performance improvement over word embeddings in a range of downstream classification tasks.

The application of methods proposed in this thesis is demonstrated on experiments estimating the demographics and personality of a text author, and labeling the text with its subjective charge and sentiment conveyed. We therefore also provide empirical insights into which types of features are informative for these document classification problems, and suggest explanations grounded in psychology and sociology. We further discuss the issues that can occur as human experts are prone to diverse biases when classifying data.

To summarize, we could show that lexical-semantic knowledge can improve text classification tasks by supplying the hierarchy of abstract concepts, which enable better generalization over words, and that these methods are effective also in combination with the deep learning techniques.

Zusammenfassung

Mit dem Aufkommen von großen Datensätzen und fortgeschrittenen Klassifikationsalgorithmen wurde für lange Zeit angenommen, dass die Verwendung lexikalisch-semantischen Wissens zu keinen besseren Klassifikationsergebnissen führt. Grundlage dieser Annahme war, dass die Bedeutung von Wörtern direkt aus den verfügbaren Textdokumenten vom gewählten Klassifikationsalgorithmus erlernt werden kann. In der vorliegenden Arbeit wird gezeigt, dass diese Annahme nicht als allgemeine Aussage gelten kann. Insbesondere werden mehrere Ansätze vorgestellt, wie lexikonbasiertes Wissen zu signifikant besseren Ergebnissen führt. Darüber hinaus wird erörtert, unter welchen Bedingungen bessere Ergebnisse zu erwarten sind.

Bei Textklassifikationsaufgaben muss eine Menge von Dokumenten in verschiedene Kategorien automatisch sortiert werden. Mit der zunehmenden Menge an digitalisierten Texten sowie mit der erhöhten Verfügbarkeit von Rechenleistung hat die Forschung und Entwicklung an automatisierten Techniken der Textklassifikation in den vergangenen Jahren enorme Fortschritte erzielt. Trotz dieser Fortschritte bleibt die lexikalisch-semantische Ambiguität wichtiges Problem: Ein einzelnes Wort kann mehrere Bedeutungen haben und kann nur unter Berücksichtigung des Kontextes bestimmt werden.

In den bisherigen Ansätzen zu Textklassifikationsaufgaben wurde dieses Mehrdeutigkeitsproblem oft mit der Annahme vernachlässigt, dass ein Dokument typischerweise genügend Worte enthält, um die mehrdeutigen Fälle ignorieren zu können. Um beispielsweise das Thema eines Textdokuments zu bestimmen, das das Wort “Bank” enthält, muss der Sinn des Wortes “Bank” nicht bestimmt werden, wenn zusätzlich Wörter wie “Fluss” oder “Finanzen” im Text zu finden sind. Jedoch muss ein solches zusätzliche Wort nicht notwendigerweise vorhanden sein. Zweckmäßigerweise zählen lexikalisch-semantische Ressourcen in der Regel alle Sinne eines Wortes auf und organisieren sie durch konzeptuell-semantische und lexikalische Beziehungen zu einem Netzwerk. Dies eröffnet eine Vielzahl von Optionen, um zu entscheiden, welche Wortbedeutung im gegebenen Fall am plausibelsten ist. Der erste Fragenkomplex, der in dieser Arbeit behandelt wird, lautet: Was sind die Konsequenzen, wenn die wissensbasierten Disambiguierungsmethoden zusätzlich zu den implizit im Wortkontext des Dokuments enthaltenen Informationen genutzt werden können? Wie sehr würde sich die Klassifikationsleistung verbessern? In dieser Arbeit wird die Auswirkung von Wort-Ambiguitäten auf eine Reihe von Textklassifikationsaufgaben untersucht und die Leistung ausgewählter Ressourcen-basierter Algorithmen zur Erfassung von Worterklärungen bewertet. Es wird gezeigt, dass die lexikographischen Sinnunterscheidungen, die durch die lexikalisch-semantischen Ressourcen zur Verfügung gestellt werden, nicht für jede Textklassifikationsaufgabe geeignet sind. Daher wurden alternative Techniken zur Begriffsdefinition aus dem Kontext für Anwendungen in der Semantikanalyse entwickelt.

Das zweite Problem in der Textklassifikation ist das Problem der Synonymie. Für jedes Dokument repräsentieren die verwendeten Wörter nur einen kleinen Bruchteil der Wörter aus dem insgesamt möglichen Wortschatz. Wenn wir einzelne Wörter oder Sinne als Merkmale im Klassifikationsmodell verwenden, wird das Klassifikationssystem die Paraphrasen der Trainingswörter nicht interpretieren können. Daher stellt sich die Frage, wie sich die Klassifizierungsleistung verbessern würde, wenn das automatisierte System feststellen könnte, dass zwei sehr unterschiedliche Wörter eine ähnliche Bedeutung teilen? In dieser Arbeit werden mögliche Lösungen untersucht, die mit Hilfe von hierarchischen Relationen ermöglicht werden, welche in lexikalisch-semantischen Ressourcen enthalten sind. In der Arbeit wird ein Ansatz entwickelt, wie das Synonymieproblem durch automatisches Bereichern der Trainings- und Testdaten mit konzeptuellen Annotationen, die über lexikalische-semantische Ressourcen zugänglich sind, vermindert werden kann. Es wird gezeigt, dass solche konzeptionellen Informationen robustere Klassifizierungsleistungen für eine Vielzahl von Aufgaben ermöglichen.

Das dritte untersuchte Problem bei der Textklassifikation, das eng mit den Obigen zusammenhängt, betrifft die Dimensionalität der Trainingsdaten. Soll zum Beispiel nicht nur jedes Wort, sondern auch Kombinationen von Wörtern für die Klassifikation verwendet werden, so wächst die Eingabe an den Klassifikator exponentiell. Mehrere Techniken zur Reduktion der Dimensionalität wurden vorgeschlagen. In den vergangenen Jahren setzte sich hier der Ansatz des sogenannten „Word-embeddings“ zunehmend durch. Da die zugehörigen Trainingsvektoren wieder auf einer mehrdeutigen Wortebene statt Sinnebene erzeugt werden, geht wertvolle Information über mögliche Bedeutungsunterschiede verloren. In dieser Arbeit wurde ein neuartiger Ansatz entwickelt, mit dem durch lexikalisch-semantische Ressourcen, die semantische Verwandtschaft von Eingabedaten zurückzugewonnen werden konnte. Die Vorteile dieses Ansatzes konnten in verschiedenen Anwendungen demonstriert werden. Insbesondere konnte bewiesen werden, dass dieser Ansatz selbst bei der Verwendung von faltenden und wiederkehrenden neuronalen Netzwerken (convolutional and recurrent neural networks) signifikant verbesserte Ergebnisse erzielen kann.

Diese Arbeit zeigt, dass die gezielte Verwendung von lexikalisch-semantischem Wissen bei automatisierten Textklassifikationsaufgaben mittels fortgeschrittener Klassifikationsansätze (z.B. “deep learning”) zu signifikant besseren Ergebnissen führen kann. Dies wird durch die Bereitstellung von Hierarchien abstrakter Konzepte, die eine bessere Verallgemeinerung von Wörtern ermöglichen, erreicht.

Wissenschaftlicher Werdegang der Verfasserin[1]

09/2005 – 09/2009	Studium der Informatik (Bachelor) an der Tschechischen Technischen Universität in Prag.
08/2007 – 05/2008	Studium der Wirtschaftsinformatik an der Technischen Hochschule in Oulu, Finland
06/2009 – 08/2009	Sommerstudenten Programm am Europäischen Kernforschungszentrum (CERN), Genf.
09/2009	**Abschluss - Bachelor of Science (Bc.).** Bachelorarbeit: “Interactive learning application for microeconomics.” Gutachter: Ing. Ivo Koubek
09/2009 – 06/2011	Studium der Informatik (Master) an der Tschechischen Technischen Universität in Prag.
10/2010 – 06/2011	Technical Fellow am Europäischen Kernforschungszentrum (CERN), Genf.
06/2011	**Abschluss - Master of Science (Ing.).** Masterarbeit: “Big data processing optimization in high energy physics.” Gutachter: Ing. Tomas Liska, Ph.D.
09/2005 – 09/2013	Studium an der Wirtschaftswissenschaftlichen Universität Prag.
09/2013	**Abschluss - Master of Economics (Ing.).** Masterarbeit: “Legal aspects of international e-commerce on web search” Gutachter: Prof. JUDr. Martin Bohacek, CSc.
10/2012 – 12/2016	Wissenschaftliche Mitarbeiterin am Ubiquitous Knowledge Processing Lab, Technische Universität Darmstadt.
06/2015 – 09/2015	Gastwissenschaftlerin an der University of Pennsylvania, USA.
seit 01/2017	Mobility Fellow an der University College London (UCL), Department of Computer Science.

[1] Gemäß §20 Abs. 3 der Promotionsordnung der TU Darmstadt

Ehrenwörtliche Erklärung[2]

Hiermit erkläre ich, die vorgelegte Arbeit zur Erlangung des akademischen Grades “Dr. rer. nat.” mit dem Titel “Leveraging Lexical-Semanitc Knowledge for Text Classification Tasks” selbständig und ausschließlich unter Verwendung der angegebenen Hilfsmittel erstellt zu haben. Ich habe bisher noch keinen Promotionsversuch unternommen.

Darmstadt, den 28.2.2017 _____

Lucie Flekova

[2]Gemäß §9 Abs. 1 der Promotionsordnung der TU Darmstadt

Acknowledgements

Despite having only one author, this thesis was a large collaborative effort. First and foremost, I want to thank Prof. Dr. Iryna Gurevych for giving me the opportunity to enter this research field, and for her very valuable guidance, feedback, encouragement and support. I also thank Prof. Dr. Benno Stein and Prof. Dr. Walter Daelemans, not only for investing their time into reviewing this thesis, but also for the very inspiring discussions we had in the past. Additionally, I am indebted to my visiting research supervisors Prof. Lyle Ungar and Prof. Ingemar J. Cox for sharing their knowledge and their excitement for our projects. This work has been supported by the Volkswagen Foundation as part of the Lichtenberg Professorship Program under grant No. I/82806 and by the German Research Foundation under grant No. GU 798/14-1. Additional support was provided by the German Federal Ministry of Education and Research (BMBF) as a part of the Software Campus program under the promotional reference 01-S12054 and by the German Institute for Educational Research (DIPF). Hereby I would also like to thank the Software Campus organization team for the opportunities given, and to my industry mentor Christian Döttinger for exposing his impressive people skills and providing refreshing perspectives.

This work could advance quickly thanks to the contributions of my research assistants Tahir Sousa, Xuan-Son Vu and Radhika Gaonkar, and the bachelor and master students, all of whom I was happy to support in their starting careers. Dr. Richard Eckart de Castilho, Dr. Johannes Daxenberger and Dr. Oliver Ferschke accelerated this thesis notably with their software frameworks. Additionally, I would like to thank all my current and former colleagues for valuable discussions, ideas, feedback, proof-reading of my early drafts in all those years, and for simply sharing and bearing all the research highs and lows. Thank you, Margot, Christian, Ilja, Emily, Ivan, Tristan, Johannes, Lisa, Silvana, Oliver, Steffen, “et al.”. I am also grateful to other members of the scientific community, who inspired me during my PhD time simply by their enthusiasm, unstoppable curiosity, and pure joy of doing research. This includes, but is not limited to, Dr. Daniel Preoţiuc-Pietro, Prof. Chris Biemann, Prof. Dirk Hovy, Prof. Matthias Schott, Prof. Anders Søgaard, Prof. David Bamman and Prof. Torsten Zesch.

Last but not least, I thank my entire family, especially the little Sebastian, Alan, Sandra, and Hanna for making very sure I don’t forget the healthy balance of work and play. Special thanks to my parents for accepting my decision to leave the “good and safe job” for my PhD, and for being just as supportive as they were 12 years ago, when I started to do “the crazy physics stuff”. My thanks belong also to Olga, Ruth, Simone, Vojta and Libbi for being here and caring, and to the “vereins” of Flörsheim for making me feel at home. Yet, above all, I want to thank my partner for trusting in me so much more than I did, and making this possible. This is for you. I might find a different way of spending evenings now.

Contents

1 Introduction

> *Writing is easy. All you do is stare at a blank sheet of paper until drops of blood form on your forehead.*
>
> **— Gene Fowler**

We entered a new era of data-driven society. The information around us and about us is being produced and collected at unprecedented levels. It is estimated that we create 2.5 million terabytes of data per day, with 90% of world's data generated over the last two years [Pathak, 2014]. Since all this data is beneficial only when transformed to actionable knowledge insights, computational processing of unstructured textual information is increasingly essential in all spheres of daily life. This is closely related to the research field of word semantics, which focuses on how the meaning can be conveyed from a word. Some of the word semantic knowledge is captured by experts in machine-readable lexical-semantic resources.

In this thesis we study the connection between word semantics and text classification on a broad variety of tasks. Text classification is the task of sorting a set of documents into a set of predefined categories (also known as classes, or labels). A common example of text classification is spam filtering - given a set of e-mails, we train a computational model to decide if a new message is spam or a legitimate correspondence. However, current text classification applications go well beyond this. Computational models have been used to determine the genre and quality of a text, an appropriate audience, suitable age of a prospective reader, or the demographics and personality of the text author. Other applications include for example language identification and predicting the native language of a writer based on her English style. A large area of text classification research and applications is also labeling the text segments with emotions they convey.

With the increasing availability of both the computing power and the digitized text, the techniques to automate text classification have witnessed a booming interest. At the same time, the rise of big data led to the popularity of distributional hypothesis [Harris, 1954] - i.e., "a meaning of word comes from its context" - and to the criticism of lexical-semantic resources as too academic for real-world NLP applications [Jorgensen, 1990, Palmer, 2000, Ide and Wilks, 2007]. For long, it was assumed that the lexical-semantic knowledge will not lead to better classification results, as the meaning of every word can be directly learned from the document itself. In this thesis, we show that this assumption is not valid

as a general statement, and present several approaches how lexicon-based knowledge will lead to better results. Moreover, we show why these improved results can be expected.

1.1 Challenges in text classification tasks

Probably the most apparent issue in processing unstructured text is the lexical-semantic ambiguity [Jurafsky and Martin, 2009], i.e., a word by itself can have multiple meanings, and the meaning of a word in a particular usage can only be disambiguated by examining its context. In text classification tasks, the ambiguity problem has often been neglected, with the prevalent assumption that the document contains enough words to safely ignore the ambiguous ones [Resnik, 2006]. For example, to classify a topic of a document containing the word *bank*, we don't need to explicitly disambiguate it, if we find the word *river* or *finance*. However, such additional words may not be always present. Conveniently, lexical-semantic resources typically enumerate all senses of a word, and organize them into a network by means of conceptual-semantic and lexical relations. We can therefore decide from a set of options which word sense is the most plausible in our context. The first question we ask in this thesis is: **Which impact does the word sense disambiguation have on document classification and why?** How much exactly would the classification performance improve, if we were able to determine the specific meaning of each word in advance? What if we use the knowledge-based sense disambiguation methods in addition to the information provided implicitly by the word context in the document? In this thesis, we quantify the impact of word ambiguity on a range of document classification tasks and evaluate the performance of selected resource-based word sense disambiguation algorithms (Chapter 3). We note that the lexicographic sense distinctions provided by the lexical-semantic resources are not always optimal for every text classification task, and propose an alternative technique for disambiguation of word meaning in its context for sentiment analysis applications.

The second problem in text classification is the synonymy problem, sometimes referred to as the lexical gap. For any document, the words used represent only a tiny fraction of the words in the total possible vocabulary. If we use individual words, or senses, as features in the classification model, our system will not be able to interpret the paraphrases, which have similar meaning, but use different expressions than those, that the system has encountered in the training phase. How much would the classification performance improve if the system could determine that two very different words represent the same meaning? In this thesis, we explore the solutions possible by using the hierarchical relations contained in lexical-semantic resources. We propose to address the synonymy problem by automatically enriching the training and testing data with conceptual annotations accessible through lexical-semantic resources (Chapter 4). The research questions we are asking here are the following: **Is it helpful to supply the classifier with additional semantic**

information about the content of the document? If so, how? We show that such conceptual information (which we call "supersenses"), in combination with the previous word sense disambiguation step (for example, annotating every occurrence of the word *dog* either with the supersense *animal*, or with the supersense *food* in case of *hot dog*), helps to build more robust classifiers and improves the classification performance of multiple tasks (Chapter 5). We further circumvent the sense disambiguation step by training a supersense tagging model directly. Previous evidence suggests that the sense distinctions of expert lexicographic resources are far subtler than what is needed for downstream NLP applications [Ide and Wilks, 2007], and by disambiguating the concepts directly on a supersense level, we can reduce the number of errors.

The third problem in text classification, closely related to the one above, is the curse of dimensionality. Imagine we want to see every English word in the training data for our classification model. We want to know not only if each single word predicts a certain document class, but which combinations of words predict it and which ones do not. Our need for training data thus grows exponentially with the number of words monitored. Several techniques for dimensionality reduction were proposed, most recently the representation learning, producing continuous word representations in a dense vector space, also known as word embeddings. However, since these vectors are again produced on an ambiguous word level, the valuable piece of information about possible distinct senses of the same word is lost, in favor of the most frequent one(s). **Can we use lexical-semantic resources to regain the sense-level notion of semantic relatedness back while operating within the deep learning paradigm?** We propose and evaluate a method to integrate supersenses into the deep learning setup by building supersense embeddings (as a parallel to word embeddings) from large sense-disambiguated resources (including, e.g., English Wikipedia). We examine the impact of different training data for the quality of these embeddings (Chapter 4), and demonstrate how to employ them in deep learning text classification experiments. Using convolutional and recurrent neural networks, we achieve a significant classification accuracy improvement in a range of downstream classification tasks (Chapter 6).

1.2 Conceptual model of the interplay of explicit and implicit knowledge

The potential and the limits of lexical semantic resources have underlying reasons. In this section we take first steps towards modeling the relations between the implicit knowledge in the documents and the explicit knowledge maintained in external resources in terms of the classification bias, towards which the challenges mentioned in the previous section contribute.

The bias of a learning algorithm can be assessed with respect to two dimensions: representational bias (referring to the size of the hypothesis space, including e.g. the number of features), and procedural bias (referring to the exploration strategy) [Quinlan, 1993]. While a strong representational bias raises a classifier's generalization capability, and is therefore desired, a strong procedural bias raises the sensitivity of a learning algorithm with respect to the training data, and is to be avoided. At the same time, a too strong representational bias will compromise the correctness due to the construction of a too coarse hypothesis space [Stein et al., 2010]. Representational bias can be characterized along several axes, including strength and correctness [Utgoff, 1986]. A strong representational bias for the hypothesis space implies a small hypothesis space (few features), a weak representational bias implies a large hypothesis space (many features). A representational bias is considered correct if it defines a hypothesis space that includes the target concept, otherwise, it is incorrect. Improving a classifier's generalization capability means to address its representational bias. This can be achieved by reducing the number of features (increasing the bias strength), and replacing weak features by discriminative features (increasing the bias correctness). The preferred solution with small training corpora is thus using few features with a coarse domain [Stein et al., 2010].

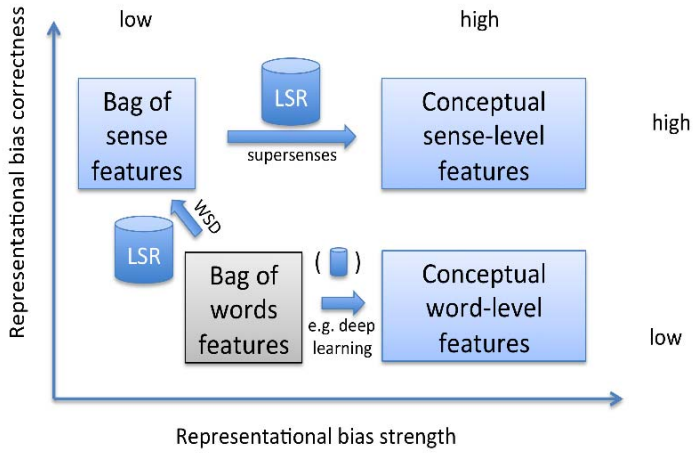

Fig. 1.1: Our model of hypothesized contribution of lexical-semantic features to the generalization of a text classification learning algorithm via representational bias, leading to a desired increase in the bias strength and correctness.

Figure 1.1 illustrates the hypothesized contribution of lexical-semantic features to the generalization of a text classification learning algorithm via representational bias. We expect the word sense disambiguation to increase the bias correctness, i.e., the ability to distinguish between the target classes through making a difference between two senses of the same word feature. At the same, this comes at the expense of the bias strength,

with the larger number of features more likely to overfit a small training corpus. The bias strength can then be increased by grouping the features into a small number of high-level concepts based on the mapping in lexical-semantic resources. By doing this on sense-level, we hope to preserve the higher bias correctness, giving us a comparative advantage over the word-level conceptual approaches (both those implemented via lexicons or through distributional methods).

The application of methods proposed in this thesis is demonstrated on a range of tasks, including experiments estimating the demographics and personality of a text author, and labeling the text with its subjective charge and sentiment conveyed. We therefore also provide **empirical insights into which types of features are informative for these document classification problems**, and suggest explanations grounded in psychology and sociology. We further discuss the issues that can occur as human experts are prone to diverse **judgment biases when classifying data**, and offer strategies to avoid those, leading to classification models of higher quality and higher ethical standard [Barocas and Selbst, 2016, Zafar et al., 2015, Hovy and Spruit, 2016] (Chapter 7).

1.3 Thesis organization

An overview of the components of this thesis, and of the workflows explored to address the above-presented research problems, is displayed in the Figure 1.2. This dissertation is organized in accordance with these workflows as follows:

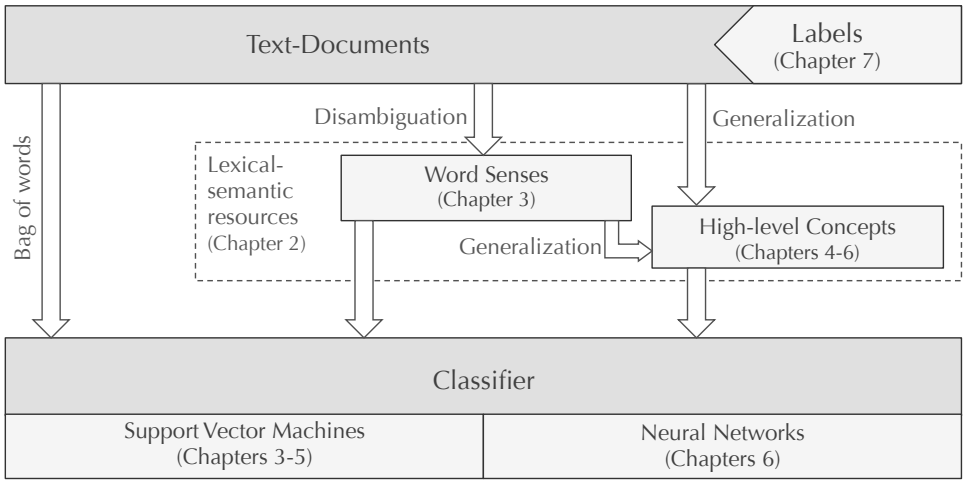

Fig. 1.2: Overview of the workflows and components explored in this thesis. We first examine the problem of lexical-semantic ambiguity, focusing on word senses. Then we move on to the synonymy problem, exploring strategies to group words to high-level concepts. We show how the first sense disambiguation step can be replaced by high-level concept disambiguation directly. We then analyze the impact of these high-level concept features on the classification performance, both in conventional text classification settings (support vector machines) and in deep learning architectures (convolutional and recurrent neural networks). At the end, we discuss the processes of training data creation and the importance of class label quality.

Chapter 2 provides a brief overview of the **lexical-semantic resources** and supervised **classification methods**, reviews the methods and theories which attempted to combine the two in related work, and introduces the terminology used in the following chapters.

Chapter 3 introduces the concept of word senses and the major word sense disambiguation algorithms. Our main research question in this chapter is: Does the **word sense disambiguation**, and subsequently using sense-level features in the classifier, improve the performance over using word-level features? We present and discuss our experimental results across five datasets. Additionally, we propose an alternative, contextual disambiguation algorithm for sentiment-bearing words, and show that our method can be used to improve sentiment polarity lexicons for a new target domain.

Chapter 4 moves from addressing the ambiguity problem towards addressing the synonymy problem in text classification. We explain how we can use lexical-semantic resources

to obtain **high-level concepts** (which we call **supersenses**) to help the classifier generalize over individual words in the training corpus. For accessing this information in a lexical-semantic resource, a previous word sense disambiguation step is necessary. We argue that this step is suboptimal given its too fine granularity, and develop an alternative model which enables to annotate supersenses directly from the documents. For this purpose, we introduce the concept of dense vector representations of these high-level concepts, which we call **supersense embeddings**, and we investigate their semantic properties, including the impact of the choice of a training corpus, from which these embeddings are built.

Chapter 5 presents experiments, in which we evaluate the utility of concept generalization, achieved through the supersense tagging with the annotation model we've built in the previous chapter, or through an intermediary word sense disambiguation step. We assess both of these approaches on five text classification tasks, including a novel task of personality prediction of fictional characters, and analyze **why supersenses contribute to the classification** performance increase. We also demonstrate how our approach can be extended beyond WordNet, using sense-level links to other resources (in this case VerbNet).

Chapter 6 provides an overview of the neural network approaches to text classification, and picks up on our supersense vector (embedding) concept from Chapter 4. We propose a **deep learning approach** combining convolutional and recurrent neural networks, which we enrich with supersense embeddings. We conduct a range of document classification experiments, demonstrating that the additional semantic information from supersense embeddings improves the classification performance also in deep learning settings.

Chapter 7 examines the factors which may influence the quality of obtained labels, that we later use as a ground truth for the classifier training. Specifically, we examine the influence of the formulation of the task, annotator's prior assumptions about the data, and the personal settings of an annotator, on the **quality and possible bias of the obtained labels**.

Chapter 8 draws **conclusions** from the preceding chapters and summarizes both the findings of our analysis in the defined scenarios and challenges that still remain to be addressed in future work.

1.4 Main contributions

We now give a brief summary of the main contributions of this thesis:

Impact of word sense disambiguation on text classification We addressed the ambiguity problem in text classification and evaluated the impact of word sense disambiguation on a range of document classification tasks, replacing the word-based features with the sense-based features. Although the WSD impact on the classification performance on our datasets was inconclusive (see the error analysis and discussion in Chapter 3), the quality of sense annotations was sufficient to use them for accessing other semantic information.

Word polarity disambiguation for sentiment classification We propose an alternative disambiguation algorithm in the context of sentiment analysis tasks. We show that the distinction between WordNet senses is not sufficient to properly determine the sentiment polarity of a word, as the same sense can be both positive or negative dependent on a given context, and propose a semi-supervised contextual disambiguation method benefiting from a large distributional thesaurus. Our curated sentiment expressions are freely available to the research community. This contribution is presented in Chapter 3 and published in [Flekova et al., 2014a] and [Flekova et al., 2015b].

Impact of adding high-level concepts on text classification We propose to address the synonymy problem in text classification by annotating nouns and verbs with their high-level semantic concepts (supersenses) obtained from lexical-semantic resources. We conduct experiments across numerous document classification tasks and show that adding this information into the feature set improves the classification results. This contribution is presented in Chapter 5 and partially published in [Flekova and Gurevych, 2015] and [Flekova and Gurevych, 2016].

Supersense tagging model In order to overcome the issues related to fine-grained word sense disambiguation, which is a necessary preprocessing step to accessing the supersense information in lexical-semantic resources, we propose to train a supersense tagging model directly. We build a model using a multi-layer perceptron architecture and several semantic features, and show that our model performs comparably or better than the state of the art on recently published social media datasets. Our model is open-source, with the code available on our group website. This contribution is presented in Chapter 4 and published in [Flekova and Gurevych, 2016].

Supersense embeddings In order to overcome the dimensionality problem in text classification, and to enable using supersense annotations in deep learning architectures, we introduce the concept of dense vector representations of supersenses, which we call *supersense embeddings*. We investigate their semantic properties, including the impact of the choice of a training corpus, on which these embeddings are built. We are the first to build a joint model of words and supersenses in the same semantic space, which facilitates the qualitative analysis. Our supersense embeddings are

freely available in a generally compatible word2vec format, together with the code to build them. This contribution is presented in Chapter 4 and published in [Flekova and Gurevych, 2016].

Deep learning architecture integrating supersense embeddings We propose a deep learning approach combining convolutional and recurrent neural networks integrating supersense embeddings thanks to the joint word and supersense embedding model. We conduct a range of document classification experiments, demonstrating that the additional semantic information from supersense embeddings improves the classification performance also in deep learning settings. Our neural network architecture is open-source and published on the group website. This contribution is presented in Chapter 6 and published in [Flekova and Gurevych, 2016].

Biases in training data construction We draw attention to several factors which may influence the quality of obtained training labels, therefore impacting the classifier performance regardless which classification model and which semantic features are used. Specifically, we examine the influence of the formulation of the task, annotator's prior assumptions about the data, and the personal settings of an annotator, on the quality and possible bias of the obtained labels. Our main findings include the presence of stereotypes in judging other people's demographics, and higher accuracy of annotators scoring higher in Actively Open-minded Thinking [Baron, 1991]. These contributions are discussed in Chapter 7 and published in [Flekova et al., 2016a, Flekova et al., 2016b, Flekova et al., 2015a, Flekova et al., 2014b] and [Carpenter et al., 2016].

Empirical insights Analyzing the performance and classification errors in various settings of our different models and features used, we also provided empirical insights into the numerous document classification tasks related to author profiling and sentiment analysis. We determined the features predictive for each problem, and interpreted them in context of previous work on these tasks. We also introduce a new task of personality profiling for fictional characters. We evaluate several methods for collecting the gold standard labels for this task, making the resulting data freely available, and we propose and implement a method to semantically process the book texts in order to obtain relevant information for the classification. These empirical contributions can be found across the Chapters 3-7 and across all publications listed below.

1.5 Publication record

We have previously published the research related to this thesis in peer-reviewed journals, conference proceedings, and workshop proceedings of major events in natural language processing and related fields, such as the ACL, EMNLP and WWW conferences and the JLCL and SPSS journals. Chapters building upon these publications are indicated accordingly.

Lucie Flekova and Iryna Gurevych: 'Supersense Embeddings: A Unified Model for Supersense Interpretation, Prediction and Utilization', in: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, vol.1 - Long papers, pp. 2029–2041, Berlin, Germany, August 2016. (Chapters 4, 6)

Lucie Flekova, Salvatore Giorgi, Jordan Carpenter, Daniel Preotiuc-Pietro and Lyle Ungar: 'Analyzing Biases in Human Perception of User Age and Gender from Text', in: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, vol.1 - Long papers, pp. 843–854, Berlin, Germany, August 2016. (Chapter 7)

Lucie Flekova, Daniel Preotiuc-Pietro and Lyle Ungar: 'Exploring Stylistic Variation with Age and Income on Twitter', in: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, vol.2 - Short papers, pp. 313–319, Berlin, Germany, August 2016. (Chapter 7)

Jordan Carpenter, Daniel Preotiuc-Pietro, **Lucie Flekova**, Salvatore Giorgi, Carl Hagan, Margaret L. Kern, Anneke Buffone, Lyle Ungar, and Martin Seligman: 'Real men don't say "cute": Using automatic language analysis to isolate inaccurate aspects of stereotypes', in: *Social Psychological and Personality Science, SPPS*, pp. , 2016. (Chapter 7)

Lucie Flekova and Iryna Gurevych: 'Personality Profiling of Fictional Characters using Sense-Level Links between Lexical Resources', in: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1805–1816, Lisbon, Portugal, September 2015. (Chapter 5)

Lucie Flekova, Salvatore Giorgi, Jordan Carpenter, Daniel Preotiuc-Pietro and Lyle Ungar: 'Analyzing Crowdsourced Assessment of User Traits through Twitter Posts', in: *Proceedings of the Third AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*, Online Proceedings, Palo Alto, CA, USA, November 2015. (Chapter 7)

Lucie Flekova, Eugen Ruppert and Daniel Preotiuc-Pietro: 'Analysing Domain Suitability of a Sentiment Lexicon by Identifying Distributionally Bipolar Words', in: *Association for*

Computational Linguistics: Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 77–84, Lisbon, Portugal, September 2015. (Chapter 3)

Lucie Flekova, Oliver Ferschke, and Iryna Gurevych: 'UKPDIPF: A Lexical Semantic Approach to Sentiment Polarity Prediction on Twitter Data', in: *In: Preslav Nakov and Torsten Zesch: SemEval-2014 Task 9: Sentiment Analysis in Twitter, Proceedings of the 8th International Workshop on Semantic Evaluation*, pp. 704–710, Dublin, Ireland, August 2014. (Chapter 3)

Lucie Flekova, Oliver Ferschke, and Iryna Gurevych: 'What Makes a Good Biography? Multidimensional Quality Analysis Based on Wikipedia Article Feedback Data', in: *Proceedings of the 23rd International World Wide Web Conference (WWW 2014)*, pp. 855–866, Seoul, Korea, April 2014. (Chapter 7)

Lucie Flekova and Iryna Gurevych: 'Can We Hide in the Web? Large Scale Simultaneous Age and Gender Author Profiling in Social Media - Notebook for PAN at CLEF 2013', in: *CLEF 2013 Labs and Workshops - Online Working Notes*, Valencia, Spain, September 2013. (Chapter 5)

In parallel with the work discussed in this thesis, several additional research publications, which exceed the scope of this dissertation, were completed. These include:

Lucie Flekova, Tahir Sousa, Margot Mieskes, and Iryna Gurevych: 'Document-level School Lesson Quality Classification Based on German Transcripts', in: *Journal for Language Technology and Computational Linguistics*, vol. 30, no. 1, pp. 99–124. 2015.

Florian Stoffel, **Lucie Flekova**, Daniel Keim and Iryna Gurevych: 'Feature-Based Visual Exploration of Text Classification', in: *Proceedings of the Symposium on Visualization in Data Science at IEEE VIS 2015*, Online Proceedings, Atlanta, USA, June 2015.

Patrick Lerner, Andras Csanadi, Johannes Daxenberger, **Lucie Flekova**, Christian Ghanem, Ingo Kollar, Frank Fischer and Iryna Gurevych: 'A User Interface for the Exploration of Manually and Automatically Coded Scientific Reasoning and Argumentation', in: *Proceedings of the 12th International Conference of the Learning Sciences*, pp. 938–941, Singapore, Rep. of Singapore, June 2016.

Lucie Flekova and Matthias Schott: 'Fast Precision Reconstruction of Micropattern Detector Signals via Convolutional Neural Networks', in: *Proceedings of the 22nd International Conference on Computing in High Energy and Nuclear Physics (CHEP)*, Online Proceedings, San Francisco, USA, October 2016.

2 Background

> *In the beginning the Universe was created. This has made a lot of people very angry and been widely regarded as a bad move.*
>
> **— Douglas Adams**

This chapter provides the background knowledge for this thesis. As illustrated on Figure 2.1 in context of the entire dissertation overview, we will mostly clarify the conceptual blocks building the foundation for the workflows in the next chapters. To do so, we will deal with the following questions:

- What is text classification?
- Which classifiers exist and what are typical text classification tasks?
- Which linguistic information can be used to process, and subsequently classify, textual documents?
- What are features in text classification and what types of features exist?
- What is lexical-semantic knowledge and which lexical-semantic resources are available?
- Which previous work has been done in using lexical semantic knowledge in text classification tasks, and which approaches were used to extract this knowledge?

2.1 Classification

Text classification is the task of sorting a set of documents into classes (also known as labels, or categories) from a predefined set. A common example of text classification is spam filtering - given a set of e-mails, training a computational model to decide if a new message is spam or a legitimate correspondence. However, current text classification applications go well beyond this. Computational models have been used to determine the genre of the

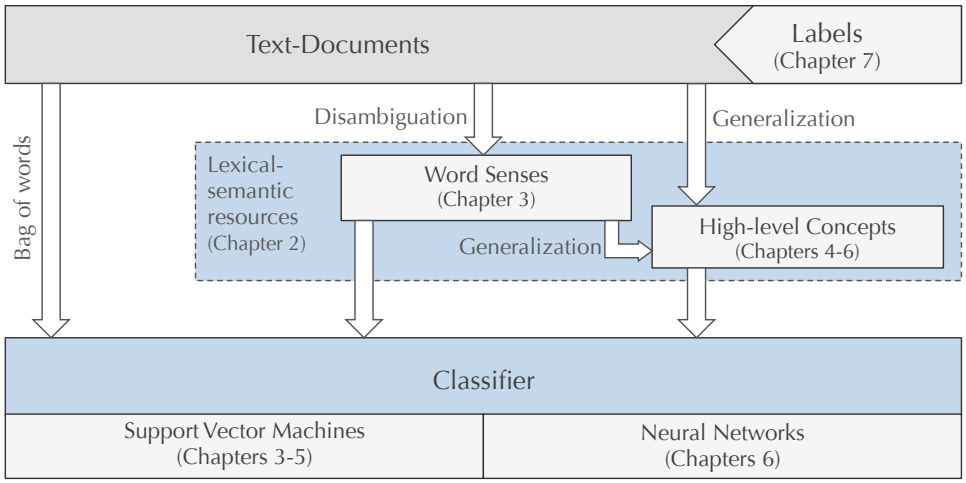

Fig. 2.1: The concepts of this thesis explored in this chapter (highlighted blue).

text, an appropriate audience, suitable age of a prospective reader, or the demographics and personality of the text author. Other applications include language identification and predicting the native language of a writer based on her English-language style. A large area of text classification research and applications is also labeling the text with emotions conveyed.

Within the scope of this thesis, when we use the term *text classification*, we refer to the *supervised text classification*, i.e., we require that the classifier derives decision rules for new documents from a set of previously (typically manually) classified samples, called training data. Similarly, another set of classified samples, called test data, is used for the evaluation of a classifier. This initial manual classification step is very important for the classification performance, as the errors in it would propagate through the entire learning process, i.e., the classifier would learn to do the same mistakes as the annotators.[1] Related challenges of training and test data construction are discussed in Chapter 7. An alternative to our text classification understanding would be an *unsupervised text classification*, where the outcomes are based on the computational analysis of a text without the expert providing sample classes. The classification model uses techniques to determine which documents are related and groups them into classes. An expert can specify the grouping method and the desired number of output classes, but otherwise does not aid in the classification process. A major challenge in this approach is that the expert has no explicit information about what the class labels are or what types of documents they contain and why, and has to derive this information by interpreting the result classes herself, which typically requires a significant additional effort.

[1]The description of labels being always created manually by annotators is simplifying. In the abundance of internet data, there are cases where it is easier to obtain class labels from existing information, e.g. from the assigned stars for the product review sentiment, or from the profiles of social media users for their gender, age, and so on.

2.1.1 Formal definition

A supervised document classification task can be formalized as the task of approximating the unknown target function

$$\phi : D \times C \rightarrow \{True, False\}$$

which characterizes how the decisions about documents would be made by an expert, by a function

$$f : D \times C \rightarrow \{True, False\}$$

which describes the classification model.

In this function, the $C = \{c_1, ..., c_n\}$ is a predefined set of classes (also called labels or categories) and $D = \{d_1, ..., d_m\}$ is a set of documents. If $\phi(d_j, c_i) = True$, then d_j is called a positive example (or a member) of class c_i, while if $\phi(d_j, c_i) = False$ it is called a negative example of c_i. Typically, no exogenous knowledge about the meaning of the classes is available to the classifier, and the classification shall be accomplished only on the basis of endogenous information, i.e., knowledge extracted from the documents themselves. The classification task is called a *single-label classification* if exactly one $c_i \in C$ must be assigned to each $d_j \in D$, or a *multi-label classification* if any number of categories is allowed for one document. *Binary classification* is a special case of single-label classification, where each document $d_j \in D$ has to be assigned either to a given category c_1 or to its complement $\bar{c_1}$. A binary classification model is therefore a function

$$f : D \rightarrow \{True, False\}$$

Supervised text classification typically includes the following steps, illustrated on Figure 2.2: In the training phase, the documents and their labels are first loaded from the source. The documents are then annotated with additional linguistic information, e.g. lemmas of words, part-of-speech, syntactic and semantic dependencies. Based on the text of the document and the linguistic information added during preprocessing, quantifiable information is extracted to be employed by the machine learning algorithm as a feature vector. Some simple examples of a feature and its value are *dog: 0* (indicating that a word 'dog' is not present in the document) or *nouns: 0.389* (indicating that 38.9% of all words in the document are nouns) etc. A machine learning algorithm is then applied to train a classifier model based on the extracted features. In the testing phase, the same features are extracted from new documents in a similar manner, and the trained classifier model is used to predict the labels of those new documents. If we want to evaluate the classifier performance, we then compare the model predictions on test data to expert-given labels. However, supplying these labels for the test data is not a necessary input for the classifier prediction.

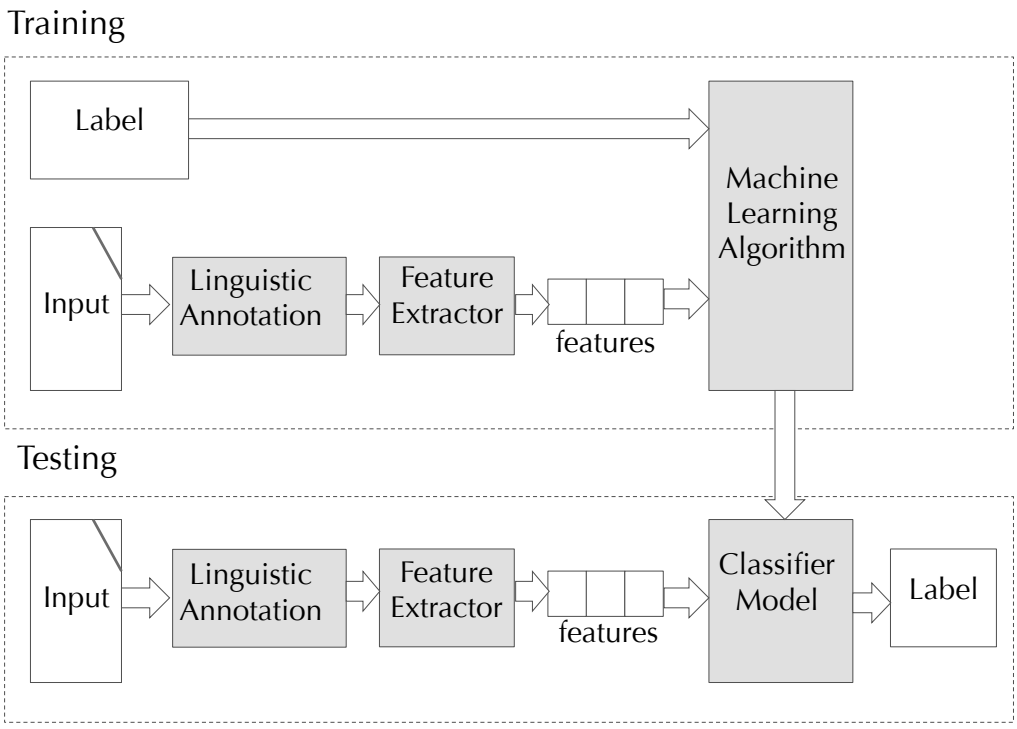

Fig. 2.2: Text classification workflow.

2.1.2 Supervised classification algorithms

As already outlined above, the dominant approach to text classification is based on machine learning techniques, which mostly replaced the previous knowledge engineering approaches. Instead of hand-written rules, the classifier identifies the most important features automatically. Furthermore, such approach is more effective in terms of scalability and domain adaptation. Some of the major algorithm families, commonly used for text classification, are:

Bayesian classifiers The idea of Bayesian learning is to build a probabilistic classifier based on modeling the probabilities of the underlying word features in different classes. A text is then classified based on the posterior probability of the documents belonging to the different classes on the basis of the word presence in the documents. The posterior probability is calculated using the Bayes Theorem:

$$P(c_i|\mathbf{x}) = \frac{P(\mathbf{x}|c_i)P(c_i)}{P(\mathbf{x})}$$

where $P(c_i|\mathbf{x})$ is the posterior probability of a target class c_i given the feature vector $\mathbf{x}$, $P(c_i)$ is the prior probability of the class c_i, $P(\mathbf{x}|c_i)$ is the likelihood which is the probability of the feature given the class, and $P(\mathbf{x})$ is the prior probability of the feature.

The most popular simple probabilistic approach is the Naive Bayes classifier, based on applying Bayes' theorem with strong (naive) independence assumptions between the features. For example, a fruit may be considered to be an *APPLE* if it is *red* and *round*, and the classifier assumes that the presence of *red* in the *APPLE* class is unrelated to the presence of *round*. Naive Bayes remains a popular (baseline) method for text classification. Despite its naive design and apparently oversimplified assumptions, naive Bayes classifier has worked quite well in many complex real-world situations. With appropriate preprocessing, it is competitive in this domain with more advanced methods including support vector machines [Rennie et al., 2003]. Naive Bayes classifiers are highly scalable, requiring a number of parameters scaling linearly with the number of features. With the maximum-likelihood training, linear training time can be achieved, which is as advantage to classifiers using an iterative approximation.

Decision trees Decision trees hierarchically divide the underlying data space using different text features. For a given document, we determine the partition that it is most likely to belong to, and use it for the purposes of classification. In other words, decisions on feature values shape a model, represented as a tree. Typically, decision trees use Entropy (H)and Information Gain (IG) criteria to construct a decision tree. Entropy expresses the purity of a sample. The entropy value is zero if the sample is homogeneous, and one if the sample is equally distributed. The Information Gain represents the decrease in entropy resulting from splitting the training sample T on a particular value of feature *f*. In other words, the entropy $H(T,f)$ after the split is subtracted from the entropy $H(T)$ before the split:

$$IG(T,f) = H(T) - H(T,f)$$

where the entropy of the sample is defined as:

$$H(T) = \sum_{i=1}^{C} -p_i log_2 p_i$$

with p_i being the probability of the class c_i. The entropy of a split, given the feature *f*, a discrete set of values *V(f)* that the feature can have, and a training sample subset T_v, where the feature *f* has the value *v*, is then calculated as:

$$H(T) = \sum_{v \in V(f)} P(T_v)E(T_v)$$

Due to their tree-like structure, decision tree algorithms tend to produce intuitive models.

Support vector machines The support vector machine (SVM) method has been introduced in text classification by [Joachims, 1998] and subsequently often used in many other text classification works. In geometrical terms, it may be seen as the attempt to find, among all the possible decision surfaces (called hyperplanes) separating the positive from the negative training examples, the one that separates the positives from the negatives by the widest possible margin, i.e. such that the minimal distance between the hyperplane and a training example is maximal.

Formally, we define a hyperplane as a set of feature vectors $\mathbf{x}$ which satisfy the condition $\mathbf{w} \cdot \mathbf{x} + \mathbf{b} = 0$ with $\mathbf{w}$ being the weight vector and $\mathbf{b}$ the bias. A subset of training examples which are the closest to the actual hyperplane is referred to as *support vectors*, $\mathbf{x}_{sub}$. The distance d between the support vectors and the hyperplane is then:

$$d = \frac{|\mathbf{w} \cdot \mathbf{x} + \mathbf{b}|}{||\mathbf{w}||}$$

Finding a hyperplane such that the distance of the support vectors to the hyperplane is maximized can be then defined as a Lagrangian optimization problem over $\mathbf{w}$ and $\mathbf{b}$.

SVMs tend to be fairly robust to overfitting and can scale up to considerable dimensionalities, which is perhaps why they have been a very popular choice in natural language processing tasks. We use them in our classification experiments in Sections 3-5.

Neural network classifiers Neural networks are used in a wide variety of domains for the purpose of classification. Neural network classifiers are, together with SVM, in the category of discriminative classifiers, as opposed to the generative Bayesian classifiers. They are able to process a large number of input signals from the data, which activate layers of interconnected neurons. The layers are made of computational nodes, loosely patterned on a neuron in the human brain, which fires when it encounters sufficient stimuli. Neural networks, especially the recent deep neural networks, have been shown to produce state-of-the-art results for many NLP problems [Collobert et al., 2011, Kim, 2014, Kalchbrenner et al., 2014, Johnson and Zhang, 2014, Nguyen and Grishman, 2015]. We discuss the deep learning approaches in more detail in Chapter 6, where we also apply those in our experiments.

Other Further algorithm families for supervised text classification, frequently used in the NLP community, include Logistic Regression models and sequence classification models. Other methods are, e.g., nearest neighbor classifiers and genetic algorithm-based classifiers. Techniques to improve existing classifiers include kernel-based algorithms, which extend otherwise linear classifiers to solve non-linear problems by mapping the input space into a higher-dimensional space, and ensemble models, which combine multiple classification models into a single predictor, often producing better results than a single model.

2.1.3 Classification tasks

There is a myriad of document classification problems, where the above mentioned techniques can be applied. Below, we introduce the text classification applications adressed in the experiments within this thesis (Chapters 3-6).

Demographic author profiling Studying gender differences has been a popular psychological interest over the past decades [Gleser et al., 1959, McMillan et al., 1977]. Traditional studies worked on small datasets, which sometimes led to contradictory results – [Mulac et al., 1990] cf. [Pennebaker et al., 2003]. Over the past years, researchers discovered a wide range of gender differences using large collections of data from social media or books combined with more sophisticated techniques. For example, [Schler et al., 2006] apply machine learning techniques to a corpus of 37,478 blogs from the Blogger platform and find differences in the topics males and females discuss. [Newman et al., 2008] conducted a detailed gender study on 14,324 samples from 70 different corpora (conversation, exams, fiction etc.) and showed that females are more likely to include pronouns, verbs, references to home, family, friends and to various emotions. Males use longer words, more articles, prepositions and numbers. Topical differences include males writing more about current concerns (e.g., money, leisure or sports). We made similar findings on web and social media data in [Flekova and Gurevych, 2013]. More recent author profiling experiments [Rangel et al., 2014, Rangel et al., 2015] revealed that gender can be well predicted from a large spectrum of text features, ranging from emotions, part-of-speech [Johannsen et al., 2015] and abbreviation usage to social network metadata, web traffic [Culotta et al., 2015], apps installed [Seneviratne et al., 2015] or Facebook likes [Kosinski et al., 2013]. [Bamman et al., 2014] also examine individuals whose language does not match their automatically predicted gender. Most of these experiments were based on self-reported gender in social media profiles. Recent results on social media data report a performance of over 90% for gender classification [Sap et al., 2014]. We have recently shown that crowdsourcing the annotations for gender labels is problematic due to human prejudice and stereotypes applied [Flekova et al., 2016a, Flekova and Gurevych, 2013, Flekova et al., 2015a].

Also the relationship between age and language has been extensively studied by both psychologists and computational linguists. [Pennebaker et al., 2003] connect language use with personality, while [Schler et al., 2006] automatically classified blogposts into three classes based on self-reported age using features from the Linguistic Inquiry and Word Count (LIWC) Framework [Pennebaker et al., 2001], online slang and part-of-speech information. [Rosenthal and McKeown, 2011] analysed how both stylistic and lexical cues relate to gender on blogs. On Twitter, [Nguyen et al., 2013] analyzed the relationship between language use and age, modelled as a continuous variable. They found similar language usage trends for both genders, with increasing word and tweet length with age, and an increasing tendency to write more grammatically correct, standardized text. State-

of-the-art results report a correlation of $r \sim 0.85$ with gold labels for age prediction [Sap et al., 2014]. Recently, [Nguyen et al., 2014] showed that age prediction is more difficult as age increases, specifically over 30 years. We confirm this finding in [Flekova et al., 2016a]. [Hovy and Sogaard, 2015] showed that the author age is a factor that influences the training of part-of-speech taggers. We also investigate the relationship between age and income and find that while many predictive linguistic features are similar, a clear distinction can be made between the two classification models [Flekova et al., 2016b].

Psychological author profiling The Big Five Factor Model of Personality (FFM) has become standard in psychology over the last 50 years, and has been shown to influence many aspects of task-related individual behavior. The independent Big Five Dimensions of Personality are (1) Extraversion vs. Introversion - being sociable, assertive, playful vs. aloof, reserved, shy, (2) Emotional stability vs. Neuroticism - being calm, unemotional vs. insecure, anxious, (3) Agreeableness - being friendly, cooperative vs. antagonistic, faultfinding, (4) Conscientiousness - being self-disciplined, organized vs. inefficient, careless, (5) Openness to experience - being intellectual, insightful vs. shallow, unimaginative.

Correlations between lexical and stylistic aspects of text and the five FFM personality traits of the author have been found in numerous experiments, with extraversion receiving the most attention [Pennebaker and King, 1999, Dewaele and Furnham, 1999, Gill and Oberlander, 2002, Mehl et al., 2006, Aran and Gatica-Perez, 2013, Lepri et al., 2010]. The LIWC lexicon [Pennebaker et al., 2001] established its position as a powerful means of such analysis.

The first machine learning experiments in this area were conducted by [Argamon et al., 2005], [Oberlander and Nowson, 2006] and [Mairesse et al., 2007]. Researchers predicted the five personality traits of the authors of stream-of-conscientiousness essays, blog posts and recorded conversation snippets. Given balanced datasets, [Mairesse et al., 2007] report binary classification accuracy of 50-56% on extraversion in text and 47-57% in speech, using word ngrams, LIWC, MRC psycholinguistic database [Coltheart, 1981] and prosodic features. Additional improvement is reported when the extraversion is labeled by external judges rather than by self-testing. Extended studies on larger datasets achieve accuracies around 55% [Nowson, 2007, Estival et al., 2007]. More recent work in this area focuses on the personality prediction in social networks [Kosinski et al., 2013, Kosinski et al., 2014] and multimodal personality prediction [Biel and Gatica-Perez, 2013, Aran and Gatica-Perez, 2013], emphasizing the correlation of network features and audiovisual features with extraversion. These trends inspired the creation of the Workshop on Computational Personality Recognition (for an overview see [Celli et al., 2013, Celli et al., 2014]. We further investigate the task of personality prediction across all of the following chapters of this thesis, with some of the results published in [Flekova and Gurevych, 2015].

Sentiment polarity prediction Sentiment research has tremendously expanded in the past decade, being of the utmost interest for researchers as well as commercial organizations. A good overview of the biggest challenges in this task is provided by [Pang and Lee, 2008]. Most of the experiments divide the data into two or three classes (positive or negative message, or positive/negative/neutral), some attempt to assign a 5-class score (positive/very positive/negative/very negative/neutral).

Initial sentiment classification approaches relied heavily on explicit, manually crafted sentiment lexicons [Kim and Hovy, 2004, Pang and Lee, 2004, Hu and Liu, 2004]. There have been efforts to infer the polarity lexicons automatically. [Turney and Littman, 2003] determined the semantic orientation of a target word *t* by comparing its association with two seed sets of manually crafted target words. Others derived the polarity from other lexicons [Baccianella et al., 2010, Mohammad et al., 2009], and adapted lexicons to specific domains, for example using integer linear programming [Choi and Cardie, 2009].

The Movie Review dataset, published by [Pang and Lee, 2005][2], has become a standard machine learning benchmark task for binary sentence classification. [Socher et al., 2011] address this task with recursive autoencoders and Wikipedia word embeddings, later improving their score using recursive neural network with parse trees [Socher et al., 2012]. Competitive results were achieved also by a sentiment-analysis-specific parser [Dong et al., 2015a], with a fast dropout logistic regression [Wang and Manning, 2013], and with convolutional neural networks [Kim, 2014]. While the state-of-the-art performance has been achieved with deep learning approaches, some researchers emphasize the importance of an in-depth semantic understanding of emotional constructs [Trivedi and Eisenstein, 2013, Cambria et al., 2013, De Marneffe et al., 2010]. Negation and its scope has been studied extensively [Moilanen and Pulman, 2008, Pang and Lee, 2004, Choi and Cardie, 2009]. Other experiments revealed that some nouns can carry sentiment per se (e.g. *chocolate*, *injury*). Recently, several noun connotation lexicons have been built [Feng et al., 2013, Klenner et al., 2014]. Interestingly, despite the dramatic progress in academic machine learning research, recent sentiment prediction challenges show that the vast majority of currently used applied systems is still based on traditional supervised learning techniques such as support vector machines with the most important features derived from pre-existing sentiment lexica [Rosenthal et al., 2014, Rosenthal et al., 2015]. One of the biggest disadvantages of polarity lexicons, however, is that they rely on either positive or negative score of a word, while in reality it can be used in both contexts even within the same domain [Volkova et al., 2013]. Our methodological and empirical contributions to the sentiment prediction task are published in [Flekova et al., 2014a, Flekova et al., 2015b, Flekova and Gurevych, 2016].

[2] `http://www.cs.uic.edu/liub/FBS/sentiment-analysis.html`

Subjectivity prediction Subjectivity analysis has received a lot of attention in the recent sentiment analysis literature, since the detection of subjective messages can be a useful input for a sentiment classifier [Pang and Lee, 2004]. Previous works in this field have mainly focused on online product reviews [Ly et al., 2011]. Others used subjectivity analysis to improve question answering in social media [Li et al., 2008a, Gurevych et al., 2009] and multi-document summarization [Carenini et al., 2013].

[Wilson et al., 2009] developed a system which automatically identifies subjective sentences and marks the subjectivity source and words expressing positive or negative sentiments. [Jiang and Argamon, 2008] classified political blogs as either liberal or conservative by identifying sentences that contain strong subjective clues based on a dictionary. [Biyani et al., 2014] analyzed subjectivity orientation of online forum threads using the combinations of words and their POS tags as features. [Li et al., 2008a] labeled 987 questions from *Yahoo! Answers* for subjectivity, and employed a supervised learning algorithm to predict it, utilizing features from both questions and answers.

[Pang and Lee, 2004] compose a publicly available dataset[3] of 5000 subjective and 5000 objective sentences, classifying them with a reported accuracy of 90-92% and further show that predicting this information improves the sentiment classification on a movie review dataset. [Kim, 2014] and [Wang and Manning, 2013] further improve the performance through different machine learning methods. We employ this dataset in our experiments in Chapter 6.

2.2 Linguistic preprocessing

The major natural language processing challenge in text classification is how to convert the raw document text into a quantifiable information that can be provided as an input to the machine learning classifier. In the simplest case, the document is converted to plain normalized counts of occurences of individual words. This is known as the bag-of-words approach. However, a common assumption held by natural language processing researchers is that this strategy can be notably improved by adding more sophisticated linguistic information to each of the documents. Below, we discuss the most common linguistic preprocessing techniques for text classification.

2.2.1 Segmentation

Given a document as a character sequence, the first step is to identify the boundaries of words and sentences. This step is known as **segmentation** or **tokenization**. During this

[3] `https://www.cs.cornell.edu/people/pabo/movie-review-data/`

process, the text is separated to a sequence of tokens, which are often loosely referred to as words, however, tokens can represent any characters or their sequence, such as punctuation. In some applications for certain languages, such as English, a simple splitting at whitespaces is a sufficient approximation to identify words. Yet for example for Chinese, this approach would bitterly fail. Even for English, we may require a more sophisticated approach to segmentation. Take for example the following sentence:

I haven't realized, that on 24/1/2012, A. B. O'Neill would celebrate his 100th birthday.

For further processing, we may want to conveniently treat *haven't* as a combination of two words and split it as *have* and *n't*, while we may want to keep the name *A. B. O'Neill* or *O'Neill* as one item, and preserve or segment the date *24/1/2012*. Therefore, many applications use trained supervised machine learning models to segment text. In our experiments, we mostly employ the Stanford Tokenizer, contained in the Stanford CoreNLP toolkit [Manning et al., 2014].

2.2.2 Lemmatization

A very common phenomenon in many languages is inflection. Inflection is the modification of a word to express tense, gender, number, or other grammatical categories in language. It is often convenient to group together all the inflections of the same word, e.g. *think, thinks* and *thought*. This is usually done by identifying for each inflected word its base form, called lemma. Therefore this step is known as **lemmatization**.

Using lemma frequencies, instead of token frequencies has been shown to improve performance for several NLP tasks [Bubenhofer, 2009]. Besides, lemmatization is also important for the usage of lexical-semantic resources, since it is the lemma, which is found within lexicons at the beginning of an entry.

However, lemmatization may also lead to a loss of information [Tognini-Bonelli, 2001] since the form of a word might include valuable information for further processing [Sinclair, 1991]. Using lemmas as supplementary annotations rather than replacements, e.g. as described in Apache UIMA [Ferrucci and Lally, 2004], allows for adding information (e.g. resolving syntactic ambiguity) without removing existing information (potentially valuable for identifying its meaning).

2.2.3 Part-of-speech tagging

Many inflected words are syntactically ambiguous, e.g. the term *saw* can be either the form of a verb or a noun. Hence we need to first identify the **parts of speech** (POS) of

the words, such as determiners, nouns, pronouns, verbs etc., in order to lemmatize them correctly.

For this, a POS tagger can be used to predict the POS of each word. Similarly to segmenters, POS taggers are algorithms, which are mostly trained on annotated data to induce a model. This model is then in turn used to predict POS tags. These tags can be of different granularity, dependent on the language as well as the application. The POS tags used within the thesis are described in Table 2 in Marcus et al. (1994). Also in this case, we mostly rely on the Stanford CoreNLP toolkit [Manning et al., 2014].

2.2.4 Syntactic parsing

With so called dependency parsers the syntactic structure of a sentence is extracted, expressing dependencies between words. A dependency relation is drawn from a governor word to a dependent word and is labeled with a dependency relation name. The governor word is the head of its dependent word.

For example, in the sentence *The dog barks*, the word *dog* is the governor for the determiner word *The* but is governed itself by the verb *barks*. In dependency grammars, each word has by definition only one governor word but can have several dependent words.

Each dependency relation is labeled with an abbreviated relation name: for example nsubj refers to a nominal subject relation, prep to a preposition relation and pobj denotes relation between an object and a preposition. A full listing of the dependency relations used by the Stanford parser is described in [De Marneffe and Manning, 2008].

2.2.5 Semantic parsing

Usually preceded by the syntactic parsing, semantic parsing is the process of mapping a natural-language sentence into a formal representation of its meaning. A shallow form of semantic representation is a case-role analysis, labeling phrases of a sentence with semantic roles with respect to a target word. A deeper semantic analysis provides a representation of the sentence in predicate logic or other formal language which supports automated reasoning.

One of the popular approaches to shallow semantic parsing is frame-semantic parsing with FrameNet. FrameNet is a semantic resource for English that consists of relational concepts/scenarios known as frames. FrameNet contains inventory of over 1000 frames, each mapped to linguistic expressions (lexical units) and structured in terms of participants/props (frame elements). For example, lie, deceive, deception, and hoodwink are

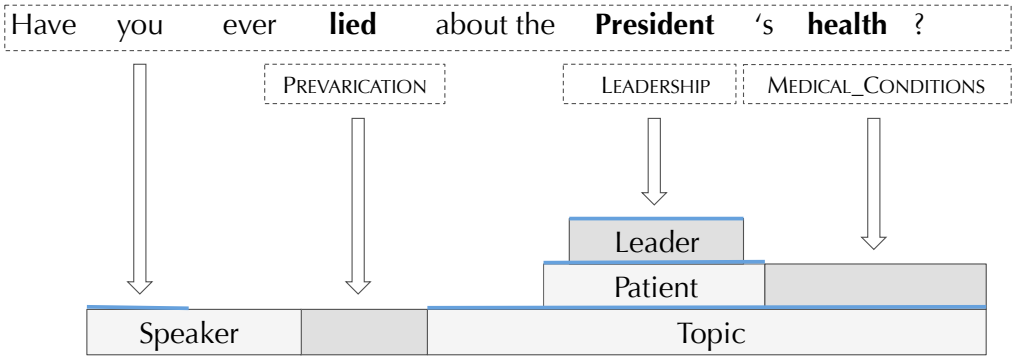

Fig. 2.3: A sample of semantic parsing with the SEMAFOR parser. Figure displays an example of three semantic frames assigned to various segment of the sentence - the PREVARICATION frame (evoked by the word *lied*), whose frame elements include the Speaker and Topic, the LEADERSHIP frame (evoked by the word *President*), containing the frame element Leader, and the MEDICAL CONDITIONS frame (evoked by the word *health*), identifying the Patient

all known to evoke the Prevarication frame, whose frame elements include the Speaker (prevaricator), Addressee, and Topic. An example of semantic parsing of a sentence with the SEMAFOR[4] parser[5] is given at Figure 2.3. Semantic parsing is a very active area of modern NLP research. The error rate of currently available open-source tools is still relatively high, often with a demanding computational overhead. We therefore do not investigate semantic parsing further within the scope of this thesis.

2.3 Features used in text classification

One of the most important tasks of a machine learning classifier for textual data is to learn meaningful patterns over the actual content of the document. In the previous section we have discussed which additional linguistic information we can assign to a piece of text. In the following, we explain how this information can be converted into measurable features serving as a classifier input.

2.3.1 Types of features

In this thesis, we use the conventional terminology of referring with the word "feature" to a text property, instantiated as a numerical input to a classifier.

[4] `http://www.cs.cmu.edu/~ark/SEMAFOR/`
[5] parsing result obtained from `http://demo.ark.cs.cmu.edu/parse/`

The basic type of features, used since the beginning of text classification research, are **lexical features**. In the simplest case, these features are obtained by counting individual words in all training documents. Let's say we want to classify documents discussing animals and irrelevant documents. Our features, based on the words found in the documents, can include the features *cat*, *dog* and *taxes*. An instantiation of these features on three specific documents in the data can look like:
Document 1: (*cat*:1), (*dog*:1), (*taxes*:0), ...
Document 2: (*cat*:0), (*dog*:1), (*taxes*:0), ...
Document 3: (*cat*:0), (*dog*:0), (*taxes*:1), ...
This approach to representing features is known as a **bag of words** approach. It has been repeatedly found that the exact counts are not necessary for most applications, and expressing only a binary occurrence of a word in a document results in similar performance while accelerating the computation. A generalization of the bag of words are the **word ngram** features, which capture also fixed sequences of several consecutive words. Representing individual words is a special case of ngram features, referred to as unigrams, but we can also choose to represent the occurrence of all word pairs (bigrams) or triples (trigrams). This method can be also applied on individual characters, capturing e.g. character trigrams instead of full words. The disadvantage of all such approaches is the exponential increase of the feature space, therefore it is usually combined with a feature selection technique.

Another simple type of features representing a document are **surface features**, sometimes referred to as length-based. These express for example the overall length of a document, or an average or maximal length of a word or a sentence. Despite their simplicity, such features can be an important indicator for tasks where the writing style differences are relevant, e.g. objectivity, personality or demographic predictions [Flekova et al., 2014b, Flekova and Gurevych, 2013, Flekova et al., 2016b].

Syntactic features are used to capture the frequencies and patterns in syntactic annotations, such as part-of-speech tags or syntactic dependencies. After annotating the data, as explained in the previous section, we can count the occurrence of the tags of interests, e.g. pronouns, or the occurence of their ngrams (such as part-of-speech ngrams, e.g. a sequence *noun,verb*). Often related to syntax are also the broader defined **stylistic features** which can measure properties such as the frequency of commas, smileys, or fillers.

Semantic features typically attempt to group words to certain categories based on their meaning, and then quantify the occurrence of words (or phrases) belonging to this semantic category. Usually an external knowledge source is used, in the simplest case a named list (e.g., Family) containing the words belonging to the category. If we then see in our documents words such as *mum* and *dad,* the representation for the classifier would be (*Family*:2). Semantic features can be much more complex than that, operating on word senses, semantic roles (see the semantic parsing above), or measures of semantic

relatedness of words, sentences and documents (for an overview of semantic relatedness measures see [Zesch, 2010])

There can be many additional feature types dependent on the text classification task, for example morphological or temporal features, and our list is by no means exhaustive. Furthermore, there is no general consensus on grouping the classification features used, and an appropriate granularity and naming of the groups is usually determined based on the task at hand. Grouping features by type is often beneficial in a qualitative analysis of possible classification models for a task, as examining performance after removing a set of features with certain properties (feature ablation) can provide insights valuable for the model interpretation.

Feature selection techniques

A major difficulty in the text classification problems is the high dimensionality of the feature space. The longer and the more specific units of meaning we attempt to capture, the more the feature space increases. In feature selection, we attempt to determine the features which are most relevant to the classification process. For example, some of the words are much more likely to be correlated to the class distribution than others, and there is no need to count all of them in all the test data. Therefore, a wide variety of methods have been proposed in the literature in order to determine the most important features for the purpose of classification. Feature selection methods which are commonly used for text classification are the Information Gain, Mutual Information and the χ^2 test. For other algorithms, we refer the interested reader to [Guyon and Elisseeff, 2003].

Information Gain is closely linked to the concept of entropy, which expresses the impurity of an arbitrary collection of examples. Information Gain quantifies the expected reduction in entropy, caused by partitioning the examples according to a given attribute. In other words, it measures the number of bits of information obtained for category prediction by knowing the presence or absence of a feature in a document.

Mutual Information is a measure of the mutual dependence between two random variables. It quantifies the amount of information (number of bits) obtained about one random variable through the other random variable. In the text classification case, these two variables are the feature and the category, i.e., we consider how often a feature and the category co-occur for one document, compared to when a feature occurs with a different category, and a category occurs while the feature being absent in the document.

χ^2 test measures the lack of independence between two variables, in this case the feature and the category. Similarly to mutual information, we compare how often a feature and the

category co-occur for one document, compared to when they appear separately. A major difference is that χ^2 is a normalized value and hence the χ^2 values are directly comparable for different features of the same category.

2.4 Lexical-semantic knowledge

Lexical-semantic knowledge has been generally captured through two main approaches - the resource-based and the corpus-based one. The former is based on using knowledge from structured lexical-semantic resources, while the second, more recent one aims at learning directly from unstructured textual corpora, often in an implicit way. While the corpus-based methods were successful in many applications, the output can be noisy and often erroneous on fine-grained level or unfrequent cases.

2.4.1 Terminology

Lexical-semantic resources strive to encode the human knowledge of language in machine-readable form, so that the machines can interpret natural language in accordance with human perception. A **lexical-semantic resource** is hence a digital knowledge base which provides lexical information on words and multiword expressions of a particular language. This information is typically accessed through a **lemma**, i.e. one base form for all inflected versions of a word. In some cases the term **lexeme** is used, which is a combination of a lemma and its part of speech. In this work, we focus on resources which provide lexical-semantic information on a word sense level. A **word sense** is a pairing between a lemma and one of the definitions (or examples) of its possible meanings.

Note that different words may have the same sense (synonymous words) and the same words can also have multiple senses (polysemous words). Words are thus always subject to interpretations. If a word does not have multiple interpretations, it is called monosemous.

Lexical-semantic resources are typically either **expert-built**, or, more recently, **collaboratively-built**. The expert-built resources are created by a limited set of editors, such as computational linguists, which use their personal introspection to compile the knowledge. Collaboratively-built lexicons are open to any volunteer who wishes to contribute, with no or very few restrictions such as the existence of an account. Collaboratively-built resources are therefore more frequently updated, and while they usually contain less syntactic information than the expert-built ones, their advantage is the coverage of neologisms and topical domains that are close to the online community, e.g. technology. By far the most widely used resource operating on the word sense level is the expert-built English WordNet [Fellbaum, 1998]. Very often, there are only subtle differences between senses, e.g. there

Type / POS	Total	Nouns	Verbs	Adjectives	Adverbs
Synsets	117,659	82,115 (70%)	13,767 (12%)	18,156 (15%)	3,621 (3%)
Word-Sense pairs	206,941	146,312	25,047	30,002	5,580
Monosemous words	128,391	101,863	6,277	16,503	3,748
Polysemous words	26,896	15,935	5,252	4,976	1,832
Polysemous senses	79,450	44,449	18,770	14,399	1,832
Avgerage polysemy	-	1.24	2.17	1.40	1.25
- excl. monosemous	-	2.79	3.57	2.71	2.50

Tab. 2.1: Number of senses contained in WordNet and the proportion of parts of speech covered by these senses.

are 39 different senses listed for the word *go* in WordNet, differentiating the meanings of *go* e.g. in the sense of moving and departing.

Lexical-semantic resources typically group senses by the same lemma, which is convenient for word sense disambiguation - we can access all possible senses for a word easily, and decide, which one is the most suitable in a given context.

Information provided in a lexical-semantic resource at the word sense level can include any of the following: definition of a sense, example of a word usage in this sense in a sentence or a phrase, relation of the sense to other senses (for example, what are the synonymy and antonymy of it, or hypernymy and hyponymy), syntactic behavior of the sense, its semantic predicates and arguments, semantic roles (agent, patient...), or selectional preference information (e.g., not every subject can speak).

Below, we present two of the most popular English lexical-semantic resources, WordNet and VerbNet, which we employ in the experiments in this thesis (we also operate on Wikipedia, but rather as a corpus than a resource). However, the methods we propose in this thesis (Chapters 4 - 6) can be generalized to any other resource which groups word senses into semantic categories. For an overview of other lexical resources available, we refer an interested reader to [Gurevych et al., 2016].

It is important to note that there are also domain-specific lexical-semantic resources such as the Unified Medical Language System (UMLS). While these may lead to better results in highly specialized tasks, the focus of this thesis is on the general-purpose resources and possibly widely applicable, task independent solutions.

2.4.2 WordNet

English WordNet [Fellbaum, 1998], created at Princeton University, is by far the most widely used resource operating on the word sense level, and probably the most popular lexical-semantic resource in general. WordNet was later adapted to numerous other languages (e.g., Italian, Japanese and German) and extended with various annotations (e.g., Extended

WordNet, WordNet Domains), hence also the generic name "wordnets" came in use for this type of resources with a particular structure.

Wordnets group senses by their synonymy relations to other senses, forming **synsets**, which are represented by their textual definitions (glosses), and, in most cases, contain also one or more short sentences illustrating the use of the synset members. Word forms with several distinct meanings are represented in several distinct synsets. Most of the sense relations beyond synonymy are then defined on the synset level, i.e., by links between synsets. Princeton WordNet of English in version 3.0, which we use here, contains 117,659 synsets and 206,941 lexical items, most of which are nouns (see Table 2.1).

WordNet structure The characteristic property of WordNet is that the senses and lexical items are organized into a network by means of conceptual-semantic and lexical relations. A hierarchical organization is most frequently induced via superordinate and subordinate semantic relations, i.e., hypernymy/hyponymy (a *bike* being subordinate to a *vehicle*). This arrangement is psycholinguisticially motivated, i.e. WordNet aims to represent real-world concepts and relations between them, as they are commonly perceived. These relations are transitive for noun synsets (if the synset A *is a* B and a B *is a* C, then A is also a C). Noun synsets can be also linked via meronymy relations (part-whole), e.g. *chair*, *seat* and *leg*. These relations are not hierarchically transitive.

Verb synsets are hierarchically arranged based on their specificity. For example, the more abstract synset containing *communicate* is higher than the synset containing *whisper*, and *move* higher than *run*. Verb synset are also linked when they describe events entailing one another, such as *succeed* and *try*, or *buy* and *pay*.

Adjective synsets are organized in pairs with their antonymy and linked to semantically similar synsets. Adverbs are typically linked to the adjectives from which they are derived. Relations between other parts of speech are rare, typically occurring when semantically similar words sharing a stem with the same meaning (e.g. *observe* and *observatory*).

The senses of each word are ordered in frequency based on the sense-annotated SemCor corpus [Miller et al., 1994].

Apart from synset definitions and usage examples, WordNet contains also definitions of their syntactic and semantic behavior, in particular **verb frames** and **lexicographer files**. WordNet verb frames, not to be confused with the semantic frames of FrameNet [Ruppenhofer et al., 2010], are specified by the lexicographers to illustrate the types of simple sentences in which the verbs in the synset can be used. There is 35 verb frames in total, for example *Someone –s something to somebody*, *Someone –s somebody* or *It is –ing*.

WordNet Lexicographer Files Lexicographer files, as the name suggests, were historically used by lexicographers to maintain manually curated WordNet parts in text files of a manageable size for computational processing, while enabling an easy overview for a manual lookup. Lexicographer files correspond to the subsets of syntactic categories implemented in WordNet - noun, verb, adjective and adverb. All of the synsets in a lexicographer file are in the same syntactic category. The names of the lexicographer files are of the form POS.SUFFIX, where POS is either noun, verb, adjective or adverb. Suffix may be used to organize groups of synsets into different files, for example NOUN.ANIMAL and NOUN.PLANT. In this thesis, we provide the list of lexicographer file names used in building WordNet in Chapter 4 (Table 4.1), where we also introduce their later proposed name, **supersenses**. While the term *supersenses* and *lexicographer files* can be used interchangeably when we use this WordNet information as a semantic inventory, *supersenses* can also refer to another set of abstract semantic categories assigned to word senses. For example, in Chapter 5 (Section 5.1.2) we use also VerbNet as a more fine-grained supersense inventory for verb senses.

Below we briefly introduce some of the WordNet extensions.

Extended WordNet Extended WordNet is a project of The University of Texas at Dallas. In this project, the WordNet synset glosses are syntactically parsed, transformed into logic forms and content words are semantically disambiguated, which provides a broader context for each concept [Moldovan and Rus, 2001].

WordNet Domains WordNet Domains [Magnini et al., 2001] is another extension of WordNet, created in a semi-automatic way by augmenting WordNet with domain labels. WordNet Synsets have been annotated with at least one semantic domain label, selected from a set of about two hundred hierarchically structured labels such as *geography*, *engineering* or *soccer*.[6]

SentiWordNet SentiWordNet [Esuli and Sebastiani, 2006] is a lexical resource for opinion mining on sense level, providing for each synset of WordNet three sentiment scores: positivity, negativity, and objectivity.

2.4.3 VerbNet

Probably the largest verb lexicon available for English is VerbNet [Kipper-Schuler, 2005], maintained by a research group at the University of Colorado at Boulder. The idea of VerbNet is that the verbs which share common syntactic argument alternation patterns also have particular meaning in common and thus can be grouped into semantic **verb classes**.

[6] For the full domain list, see `http://wndomains.fbk.eu/hierarchy.html`.

For example the verbs *give* and *sell* both refer to a change of possession. The classification in VerbNet is based on Levin's verb classes [Levin, 1993], which are refined and extended, yet keeping to the core idea proposed by Levin - if the members of a set of verbs share some meaning component, then the members can be expected to exhibit the same syntactic behavior and vice versa. VerbNet is hierarchically structured.

English VerbNet contains 3,769 verb lemmas in 274 first-level verb classes (further divided to subclasses with a numerical appendix, e.g., *build-26.1-1*), resulting into 5,257 verb senses. Verb senses are implicitly defined by their usage patterns rather than by explicit glosses.

Each verb class in VerbNet is exhaustively described by **thematic roles, selectional restrictions** on the arguments, and **frames** consisting of a **syntactic description** and **semantic predicates** with a **temporal function** [Kipper et al., 2008]. Syntactic descriptions depict the possible surface realizations of the argument structure for constructions such as transitive, intransitive, prepositional phrases, or resultatives. Selectional restrictions, such as *animate, human, organization*, constrain the types of roles allowed by the arguments. Temporal function relates the verb with the event characterized by it, and can take one of the three values: start(E), during(E), and end(E).

For example, the verb *play* in the class *performance-26.7* contains the roles AGENT, THEME and BENEFICIARY, with the restrictions that the agent and beneficiary have to be animate. Besides *play*, the class includes members such as *dance* and *sing*. It contains a frame with a syntactic description `Agent Verb Beneficiary Theme (NP V NP NP)`, semantic predicates `perform(during(E), Agent, Theme) benefit(E, Beneficiary)`, where `during(E)` indicates the temporal function. A usage example for the entire verh class is provided, *Sandy sang me a song*.

VerbNet also provides a sense mapping of its classes to other lexical-semantic resources, including WordNet. There is also a VerbNet for French language [Pradet et al., 2014].

2.4.4 Wikipedia

Wikipedia is a collaboratively constructed online encyclopedia produced by a community of volunteers. In a collaborative resource construction approach, a community of users gathers and edits the lexical information in an open process. This approach is a beneficial complement to expert-built lexicons, especially as it facilitates capturing neologisms, sense shifting over time, and lexical-semantic information for resource-poor languages.

The English Wikipedia currently contains over 5,300,000 articles. Encyclopedias do not have the same established history of use in NLP as, for example, wordnets, however,

a pairing of an article title and an article body can be interpreted as a sense. Articles themselves disambiguate senses ("Dog (animal)" vs "Dog (engineering)"). Additionally, there is a network of hyperlinks between concepts mentioned in the articles, which can be leveraged similarly to the WordNet structure [Zesch et al., 2007]. Intuitively, the articles are almost exclusively focused on describing noun concepts, rather than any other part of speech.

2.4.5 Linked lexical-semantic resources

Researchers have soon realized that it would be beneficial to profit from a combined information from multiple resources at the same time [Shi and Mihalcea, 2005]. The main challenge, however, is - how to combine these resources effectively? The center of this challenge is the task of *sense linking*, i.e., solving which sense from one resource relates to the sense from another resource. This problem is non-trivial, as the entries in various resources have different granularity, structure, and coverage. Sense-linking is therefore an active research area on its own, and we refer an interested reader to [Gurevych et al., 2016].

There has been several large-scale attempts at resource integration. The two major ones are UBY [Gurevych et al., 2012] and BabelNet [Navigli and Ponzetto, 2012]. UBY combines the English WordNet, Wiktionary, Wikipedia, FrameNet, VerbNet, and the German Wikipedia, Wiktionary, GermaNet, OpenThesaurus, IMSLex-Subcat and OmegaWiki, providing a standardized unified representation for all of them, accessible through a Java-based API or a web interface. BabelNet integrates WordNet, Wikipedia, VerbNet, OmegaWiki, Wikidata, Wikiquote, Microsoft Terminology, GeoNames and ImageNet. While UBY provides access to all information types covered by the underlying resources, BabelNet only captures their intersection, focusing on sense definitions and multilingual connections, with nouns and named entities being more emphasized than in UBY, i.e., BabelNet containing more resources with this focus.

2.5 Lexical semantics in text classification tasks

In the past decade, several researchers have reported on the contribution of word sense disambiguation to sentiment prediction tasks. Similarly to our experiments, most previous studies have used WordNet as a sense inventory for disambiguation. [Rentoumi et al., 2009] applied WSD in combination with SentiWordNet[7] [Esuli and Sebastiani, 2006] to predict sentiment polarity of figurative and non-figurative sentences. They evaluate the performance of three experimental settings - no disambiguation, most frequent sense (MFS)

[7] SentiWordNet assigns sentiment polarity scores to all synsets in WordNet, thus assigning different polarity score to different senses of the same word.

baseline, and a Lesk-inspired disambiguation algorithm based on the word vector similarity of glosses for each sense of each word in an 8-word context. They report that using the non-disambiguated words performs better than MFS when considering the full dataset, but MFS brings a mild improvement on figurative expressions. The gloss-based method then further improves the performance of WSD on figurative expressions, increasing the precision and recall by about 20 percentage points. [Akkaya et al., 2011] train one supervised model for each of the 90 target words, to disambiguate between the word's subjective and objective senses. They show that using such models can lead to improvement in sentiment classification - their best system architectures improve the accuracy by 3-5%. [Sumanth and Inkpen, 2015] use WSD techniques in the task of sentiment prediction of Twitter posts and SMS messages.They use a Babelfy graph-based algorithm [Moro et al., 2014] as a WSD model, assigning the disambiguated senses a SentiWordNet sentiment score, and using the summed up positive, negative and neutral scores of the senses in a message as features. They report an F-score improvement from 0.39 to 0.50 on Twitter and from 0.39 to 0.49 on SMS data compared to the best word-based system.

Regarding other document classification tasks, [Vossen et al., 2006] report an improvement from 0.70 to 0.76 F-score on topic classification of news articles in the Reuters corpus [Rose et al., 2002], applying coarse-grained WSD using WordNet domains [Magnini et al., 2001]. They train the WSD system as a supervised classifier on English and Spanish domain-annotated text, and apply it within each article using a window of 10 expressions.

The usefulness of wordnets for document classification is not commonly accepted [Vossen et al., 2006]. This scepticism partly originates from the information retrieval field, where for example [Voorhees and Harman, 1997] demonstrated that applying WSD on query words harms the performance, and conclude that "linguistic techniques are only useful if they perform close to perfect". Similarly, [Sanderson, 1994] and [Gonzalo et al., 1998] speculate that WSD is only useful when achieving at least 90% accuracy in detecting the appropriate sense. [Kilgarriff, 1997] suggests that "a task-independent set of word senses for a language is not a coherent concept", and sense distinctions shall be defined "relative to a set of interest" as determined by the NLP application. Similarly, also [Krovetz, 2002] argues that "different language applications need different sense distinctions."

WSD has been also assumed to improve performance in machine translation tasks. [Carpuat and Wu, 2005] compare the WSD to statistical language models trained on parallel sentences, and show, that these models often capture similar contextual features necessary for disambiguation, reaching higher BLEU scores. Later, the same authors [Carpuat and Wu, 2007] propose phrase-sense disambiguation, showing that it leads to better results in machine translation than WSD using predefined senses drawn from manually constructed sense inventories. They redefine the WSD task to be exactly the same as lexical choice task faced by the multi-word phrasal translation disambiguation task faced by the phrase-based machine translation system. The WSD system then directly disambiguates between

all phrasal translation candidates seen during the machine translation system training. Similar concepts were followed by [Giménez and Màrquez, 2007] and [Chan et al., 2007], using all possible translation phrases to determine a correct translation of a source phrase. [Xiong and Zhang, 2014] introduce a "word sense induction", predicting the sense of a target word by clustering words together using their neighbouring words as context to induce unsupervised senses. Regarding explicit sense inventories, [Neale et al., 2015] have recently shown including WSD as contextual features in a maxent-based transfer model results in a slight improvement in the quality of machine translation.

Negative results in document classification have been reported by [Kehagias et al., 2003]. They use 178 documents from the SemCor [Miller et al., 1994] corpus, which is manually expert-annotated with word senses, and classify the documents based on their topic. They use several text classification algorithms, comparing the bag-of-words to the bag-of-senses features for each of the settings, and do not find any significant classification improvement, although in most cases the bag-of-senses performs marginally better. Similar conclusions are reported by [Moschitti and Basili, 2004], where the authors perform cross-validation over 4 different corpora in two languages, testing two different classifiers. Using two algorithms for WSD - the most frequent sense baseline and the sense gloss overlap, they conclude that word senses are not adequate to improve text classification accuracy.

On a more general level, [Plank et al., 2014] point out that the publication bias toward positive results impedes the comparison to experiments with the opposite conclusion. [Ciaramita and Altun, 2006] mention that the contribution of WSD to downstream document classification tasks remains "mostly speculative". Others [Navigli, 2009, Izquierdo et al., 2009, Resnik, 2006, Ide and Wilks, 2007, Jorgensen, 1990] argue that WordNet's sense distinctions are too fine-grained for end-level applications.

This is why *supersenses*, the coarse-grained word labels based on WordNet's [Fellbaum, 1998] lexicographer files, have recently gained attention for text classification tasks. Supersenses contain 26 labels for nouns, such as ANIMAL, PERSON or FEELING and 15 labels for verbs, such as COMMUNICATION, MOTION or COGNITION. Usage of supersense labels has been shown to improve dependency parsing [Agirre et al., 2011], named entity recognition [Marrero et al., 2009, Rüd et al., 2011], non-factoid question answering [Surdeanu et al., 2011], question generation [Heilman, 2011], semantic role labeling [Laparra and Rigau, 2013], personality profiling [Flekova and Gurevych, 2015], semantic similarity [Severyn et al., 2013], and metaphor detection [Tsvetkov et al., 2013].

2.6 Chapter summary

In this chapter, we provided the background knowledge for the rest of this thesis. We defined the **text classification problem** and presented the most common classification algorithms and tasks. We explained the automatic linguistic processing used for further manipulation with unstructured text and we presented common groups of features typically used in text classification tasks. We characterized **lexical-semantic resources**, described in detail the most popular ones, and discussed which of those we select for further analysis in this thesis. We presented **related work** which has been done in using lexical-semantic features in text classification tasks, showing that further investigation in this direction is needed to better understand their impact. In the next chapter, we first investigate the issue of lexical-semantic ambiguity in the context of text classification tasks.

3 Lexical-semantic Features for Concept Disambiguation

> *Part of the inhumanity of the computer is that, once it is competently programmed and working smoothly, it is completely honest.*
>
> **— Isaac Asimov**

In this chapter, we examine the problem of lexical-semantic ambiguity [Jurafsky and Martin, 2009], i.e., a word by itself can have multiple meanings, and the meaning of a particular usage of a word can only be disambiguated by examining its context. In text classification tasks, the ambiguity problem has been often neglected, with the prevalent assumption that the document contains enough words to safely ignore the ambiguous ones [Resnik, 2006].

The first research question we ask in this thesis is: How much exactly would the classification performance improve if we were able to determine the specific meaning of each word in advance? What if we use the knowledge-based sense disambiguation methods in addition to the information provided implicitly by the word context in the document? In this chapter, we quantify the impact of word ambiguity on a range of document classification tasks and evaluate the performance of selected resource-based word sense disambiguation algorithms. As illustrated on figure 3.1 by the blue highlight, we do so by moving from the bag of words to the bag of senses classification setup, and comparing the two.

Subsequently, we note that the lexicographic sense distinctions provided by the lexical-semantic resources such as WordNet are not always optimal for every text classification task, and propose an alternative technique for disambiguation of word meaning in its context for sentiment analysis applications.

3.1 Approaches to WSD

The meaning of a word in a particular usage can only be determined by examining its context. Word Sense Disambiguation (WSD) is the process of identifying the sense of a polysemous word.

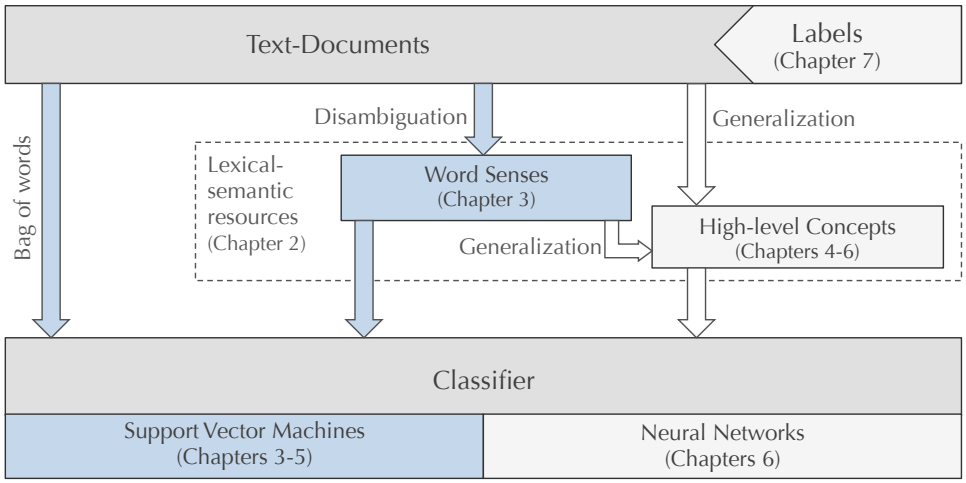

Fig. 3.1: The concepts and workflows of this thesis explored in this chapter (highlighted blue).

There is a large body of work on WSD such as [Agirre and Edmonds, 2006, Navigli, 2009, Navigli, 2012]. Different approaches to WSD include **knowledge-based systems** such as the Lesk algorithm [Lesk, 1986], which use the knowledge encoded in lexical-semantic resources, **unsupervised corpus-based systems**, which induce word senses by clustering occurrences of words [Manning and Schütze, 1999], and **supervised corpus-based systems** [Mihalcea and Csomai, 2005], in which a classifier is trained for each distinct word on a corpus of manually sense-annotated examples.

Studies have shown that even for human annotators the distinction between rather fine-grained senses is very hard [Véronis, 1998]. According to WordNet version 3.1, for example, the verb *watch* has eleven word senses, distinguishing also watching a game in the television and watching a game on the stadium. In English, top accuracies from 59.1% to 69.0% have been reported on fine-grained (WordNet-level) sense distinctions [Agirre et al., 2010, Palmer et al., 2001, Snyder and Palmer, 2004], where the baseline accuracy of the simplest possible algorithm of always choosing the most frequent sense was 51.4% to 57%. Generally, the senses of a concrete noun are easier to distinguish than different senses of an abstract noun, a verb, or an adjective; also for humans [Véronis, 1998].

3.2 WSD with lexical-semantic resources

Supervised, resource-based disambiguation requires two inputs: a lexical-semantic resource to specify the senses which are to be disambiguated (i.e., a sense inventory) and a corpus of language data to be disambiguated. From the lexical-semantic resources introduced in the previous chapter, the sense inventories for English which we use in this thesis include WordNet 3.0 [Fellbaum, 1998] and VerbNet 2.0 [Kipper et al., 2008], which we access

through the linked lexical-semantic resources UBY [Gurevych et al., 2012] and BabelNet [Navigli and Ponzetto, 2012]. With regards to the sense inventories, WordNet has a high proportion of nouns (see Table 2.1), while VerbNet, as its name suggests, specializes on providing richer semantic and syntactic information about verbs, largely based on the classification by [Levin, 1993].

From the algorithms based on lexical-semantic resources, the fundamental one is the Lesk algorithm [Lesk, 1986], which has undergone numerous modifications and extensions since. We discuss it in more detail later in this section. In general, algorithms related to Lesk are based on the definions of senses, which can be found in the resource. An alternative to the use of the definitions is to consider the word-sense relatedness using path-based methods, computing the semantic similarity of each pair of word senses based on a structure of a given resource, e.g. a distance in the WordNet graph. Such graph-based methods have been explored for example in [Navigli and Velardi, 2005, Navigli et al., 2007, Navigli and Lapata, 2010] and were shown to perform well on specific domains [Agirre et al., 2009b]. One of the implementations of graph-based WSD algorithms is the Babelfy model [Moro et al., 2014], results of which we employ in our experiments in Chapter 6. It is based on a loose identification of candidate meanings coupled with a densest subgraph heuristic which selects high-coherence semantic interpretations. However, due to the limited availability of the Babelfy interface (1000 word sense queries/day), we focus in our experiments mostly on the Lesk-based algorithms, which are made available to use with the UBY linked lexical-semantic resource through the DKPro Word Sense Disambiguation module [Miller et al., 2013].

Below, we introduce the original Lesk algorithm and its descendants.

The original Lesk algorithm

The Lesk algorithm [Lesk, 1986] is the seminal word sense disambiguation method, as it was one of the first computational attempts to disambiguate all words in an unrestricted text. It requires nothing but a machine-readable dictionary containing senses and their definitions, and a target document in which the words appear in context. For this reason, it is also easily applicable to a large number of languages, since the lexicons describing meanings of different senses are for historical reasons more widely available than wordnet-like graphs or sense-annotated corpora.

The intuition behind Lesk algorithm is that if several words appear close together, they are likely to share a similar topic. This "topic" can be accessed through their definitions, considering all possible ones and selecting those which are the closest together, as the most plausible. Hence a sequence of two or more words is disambiguated by simultaneously

retrieving the dictionary definitions of all their senses and finding the largest overlap between each combination of those.

We can formalize the algorithm as follows. Let T be a target document context consisting of a sequence of at least two words w_i:

$$T = (w_1, w_2, ..., w_n)$$

For simplicity, let's start with $T = (w_1, w_2)$. Then given a machine-readable dictionary I projecting the lexicon words $w \in L$ to a set of candidate senses $I(w) \subseteq S$:

$$I : L \rightarrow \mathcal{P}(S)$$

and a glossary G associating each sense $s \in S$ with a gloss $G(s)$ consisting of a set of words g_i:

$$G(s) = \{g_1, g_2, ..., g_n\}$$

we can disambiguate any pair of words w_1,w_2 by selecting from all possible sense candidates in the two sets of candidate senses $I(w_1)$ and $I(w_2)$ that pair of senses ($s_{i,i=1,...k}$, $s_{j,j=1,...l}$), for which the size of the intersection of the sets of gloss words $G(s_i)$ and $G(s_j)$ is the largest from the tested sense gloss pairs:

$$Lesk(w_1, w_2) = \arg\max_{s_i \in I(w_1) \subseteq S, s_j \in I(w_2) \subseteq S} |G(s_i) \cap G(s_j)|$$

In his paper, Lesk uses an example of the context *pine cone* to disambiguate the senses of both *pine* and *cone*. A lexicon defines the following set of candidate senses $I(w_1)$ for w_1 =*pine*:

1. Kinds of evergreen trees with needle-shaped leaves
   $G(s_1) = \{$*kinds, evergreen, trees, needle-shaped, leaves*$\}$
2. Waste away through sorrow or illness
   $G(s_2) = \{$*waste, away, through, sorrow, illness*$\}$

and the following set of candidate senses $I(w_2)$ for w_2 =*cone*:

1. Solid body which narrows to a point
   $G(s_1) = \{$*solid, body, narrows, point*$\}$
2. Something of this shape whether solid or hollow
   $G(s_2) = \{$*something, this, shape, whether, solid, hollow*$\}$
3. Fruit of certain evergreen trees
   $G(s_3) = \{$*fruit, certain, evergreen, trees*$\}$

Following the rule defined above, we find an overlap in *evergreen* and *trees*, therefore the best intersection $|G(s_i) \cap G(s_j)| = |\{evergreen, trees\}| = 2$ from all senses $s_i \in I(pine), s_j \in I(cone)$ occurs for the pair of (s_1, s_3).

For its straigtforward implementation and easy interpretation, Lesk algorithm has been also popular as a baseline for WSD tasks.[1]

Simplified Lesk algorithm

A major disadvantage of the original Lesk algorithm is its scalability, as the computational complexity rapidly increases with the size of the context. A popular Lesk variant, more practical to use than the original, is known as simplified Lesk [Kilgarriff and Rosenzweig, 2000]. In the simplified Lesk, we disambiguate one word w at a time by comparing each of its definitions to the word context T. Given the previous definitions, we can formally characterize the simplified Lesk as:

$$Simplified\_Lesk(w) = \arg\max_{s_i \in I(w)} |G(s_i) \cap T|$$

Extended Lesk algorithm

Both the original and simplified Lesk algorithms are susceptible to the lexical gap problem. If the algorithm finds no overlapping words at all between the sense glosses and the context, it is unable to disambiguate the word. Lesk himself in [Lesk, 1986] also discusses the option of including example sentences ($E(s_i)$) along with the sense definitions, although he considers it noisy and probably not necessary. The optimization problem would then be modified as follows:

$$Extended\_Lesk(w_1, w_2) = \arg\max_{s_i \in I(w_1), s_j \in I(w_2)} |(G(s_i) \cup E(s_i)) \cap (G(s_j) \cup E(s_j))|$$

A refinement to the extended Lesk algorithm was suggested by [Banerjee and Pedersen, 2002]. Instead of the examples, they use semantic relations between senses from a lexical-semantic resource such as WordNet to augment the sense definitions (glosses) with the glosses of the neighborings senses $N(s_i)$, i.e., hypernyms and hyponyms, of each sense:

$$Extended\_Lesk(w_1, w_2) = \arg\max_{s_i \in I(w_1), s_j \in I(w_2)} |(G(s_i) \cup G(N(s_i))) \cap (G(s_j) \cup G(N(s_j)))|$$

[1] Another common WSD baseline is using the most frequent sense, calculated on some larger manually sense-annotated corpus.

Simplified extended Lesk algorithm

Recently, a popular approach has been to combine the simplified and extended Lesk into a "simplified extended" algorithm [Ponzetto and Navigli, 2010], in which the sense definitions augmented with the examples $E(s_i)$ or the hypernymy and hyponymy sense definitions $N(s_i)$ are compared not with each other, but with the target word context T, keeping the computational time manageable:

$$Simplified\_Extended\_Lesk(w) = \arg\max_{s_i \in I(w)} |(G(s_i) \cup G(N(s_i))) \cap T|$$

3.3 Experiments: Resource-based WSD for document classification

Previous findings concerning the impact of word sense disambiguation on the performance of document classification systems have been controversial. While some experiments report minor improvements, others argue that the sense information is redundant given the document context, and the errors introduced by imperfect sense disambiguation models are actually harmful for the overall performance. To better understand these conflicting results, we conduct our own WSD experiments on several document classification tasks in the area of personality and demographic profiling.

3.3.1 Working hypothesis

Since most of the research directly exploring the impact of WSD on text classification is more than a decade old, we conducted multiple experiments on our own, aiming at obtaining a more detailed understanding of why exactly the senses may or may not be more beneficial than words in certain cases. To investigate the impact of WSD, we run experiments on 6 different tasks across 5 different corpus types, comparing bag-of-words to bag-of-senses features.

3.3.2 Corpora used

We test our settings on the tasks of author's gender prediction and author's personality prediction.

Studying gender differences has been a popular psychological interest over the past several decades [Gleser et al., 1959, McMillan et al., 1977]. There have been multiple larger-scale computational studies which have explored the correlation between various linguistic

features and the gender of the author [Schler et al., 2006, Newman et al., 2008, Rangel et al., 2014, Rangel et al., 2015]. Most find that women are more likely to include pronouns, verbs, references to home, family, friends and to various emotions, while men tend to use longer words, more articles, prepositions and numbers. Men also swear more often and discuss topics such as money, leisure, and sports more often. Most of these experiments were based on self-reported gender in blogs and social media profiles. One of these blog datasets is used in our experiments in this chapter.

The personality prediction consists of five independent tasks, predicting the personality traits of the Big Five Factor Model (FFM) of Personality [McCrae and Costa, 1987, Goldberg, 1990], which is well-known and widely accepted in psychology and other research fields. The FFM defines personality along five bipolar scales: Extraversion (sociable vs. reserved), Emotional stability (secure vs. neurotic), Agreeableness (friendly vs. unsympathic), Conscientiousness (organized vs. careless) and Openness to experience (insightful vs. unimaginative). Psychologists have shown that these five personality traits are stable across individual lifespan and demographical and cultural differences [John and Srivastava, 1999] and affect many aspects of behavior in daily life situations [Terracciano et al., 2008, Rentfrow et al., 2011]. We conduct our experiments on four personality-annotated corpora introduced below.

Stream of consciousness essays

The dataset of stream-of-consciousness essays, which we use in our experiments, was collected between 1997 and 2007 by [Pennebaker and King, 1999]. It contains 2,479 essays from psychology students, who were told to write whatever comes into their mind for 20 minutes. Each document contains the raw text, an ID of the author, and a binary label for each of the five personality classes. The labels were obtained by asking students to take the FFM personality test in the version of [John et al., 1991], i.e., to fill a behavioral questionnaire. This dataset was also used in the supervised classification experiments of [Mairesse et al., 2007], who attempts to predict personality with several classifiers, using several word-based psychological lexicons. We discuss this and similar works on this corpus in Chapter 5, while the focus of this chapter is on word sense disambiguation effect rather than the overall performance with additional lexicon features. The dataset version we use was published by [Celli et al., 2013], who derive the binary labels for the Five Factor Model (i.e., Extravert vs. Introvert, Neurotic vs. Stable, Agreeable vs. Unsympathic, Conscientious vs. Careless, and Open to experience vs. Unimaginative) by z-scores computed by [Mairesse et al., 2007] and convert it from scores to nominal classes with a median split.

Facebook personality dataset

The Facebook dataset is a sample of personality scores and Facebook profile data, collected by [Kosinski et al., 2013] by means of a Facebook application that implements the FFM personality questionnaire in a 100-item long version of [Costa and McCrae, 2008]. The application obtained the consent from its users to record their data and use it for the research purposes. We use the dataset version of [Celli et al., 2013], which is a public subset of the full Kosinski's sample. They selected only the users for which they had both information about personality and social network structure. The status updates have been manually anonymized. The final dataset contains 9,917 Facebook statuses of 250 users in raw text, gold standard (self-assessed) personality labels, and several social network measures (which we do not use), such as network size, betweenness, centrality, density, brokerage and transitivity. Binary personality labels have been derived from scores with a median split.

Twitter personality dataset

This dataset was published as a part of the PAN author profiling software challenge in 2015 [Rangel et al., 2014]. Personality traits of 328 Twitter users in 4 languages were self-assessed with the FFM personality questionnaire in its BFI-10 online version [Rammstedt and John, 2007] and reported as scores normalized between -0.5 and +0.5. For consistency with the other experiments, we derived binary labels from the scores with a median split, and use only the English subset of the data, consisting of 153 Twitter users.

YouTube video transcripts

This dataset, published by [Biel and Gatica-Perez, 2013], consists of transcripts of around 28 hours of videos provided by 404 YouTube vloggers, and annotated for personality. In contrast to the self-reported personality, used in the previous datasets, this one uses observed personality, as perceived by crowdsourcing workers. The workers were asked to watch a one-minute video sample and then answer the FFM personality questionnaire in its 10-item version of [Gosling et al., 2003] about the observed person. With the five judgments collected for each vlogger (video author), the authors of the dataset report Cronbach's alpha reliability coefficient between 0.46 and 0.63 depending on the trait, a value range similar to that reported on the original questionnaire publication [Gosling et al., 2003]. Also here, we use binary personality trait labels.

Dataset	Authors	Documents	Tokens	Tokens/Sentence	OOV-Words
Essays	2,479	2,479	1,807,410	15.2	30.80%
Facebook	250	9,917	195,840	10.6	23.30%
Twitter	153	4,562	42,315	9.03	51.27%
YouTube	404	404	293,068	17.4	8.05%
Blogs	3,100	3,100	1,978,560	14.7	15.32%

Tab. 3.1: Number of labels (authors), documents and tokens for each corpus, an average sentence length in tokens, and percentage of words not found in WordNet.

Dataset	Noun [%]	Verb [%]	Adj [%]	Pron [%]	Adv [%]	Det [%]	Prep [%]
Essays	11.01	12.10	7.16	**11.33**	8.68	6.33	8.95
Facebook	**17.88**	14.07	6.86	4.53	5.03	6.16	7.18
Twitter	13.7	**15.1**	5.1	6.8	4.6	6.5	8.7
YouTube	10.84	10.86	4.29	**11.76**	7.93	6.67	8.29

Tab. 3.2: Distribution of part-of-speech tags for each corpus, expressed as percentage of all part-of-speech tags in the corpus. Anomalies, distinctive for the corpus, are highlighted.

Overall statistics comparing the properties of the corpora are provided in Tables 3.1 and 3.2. The dataset consisting of stream of consciousness essays (hereafter ESSAYS) is the largest, containing a relatively high proportion of pronouns (mostly personal pronouns). The Facebook dataset (FACEBOOK) contains the highest proportion of nouns of all datasets. This is partly due to the fact that during the collection period Facebook experimented with an interface in which authors talked about themselves in third person, starting with their name. The Twitter dataset (TWITTER) is characterized by a higher proportion of verbs and nouns, and almost half of all word occurrences not appearing in the WordNet resource. The dataset of video transcripts (YOUTUBE) is, to the contrary, very stylistically clean, with only 8% of words being out of vocabulary. Similarly to ESSAYS, it contains a high proportion of pronouns, possibly because the vloggers frequently discuss themselves and the viewers.

Gender of bloggers

The gender dataset consists of a set of 3,100 blogs, collected by [Mukherjee and Liu, 2010]. The gender of the author was determined by visiting the profile of the author. When the gender information was not available explicitly, it was annotated based on profile pictures or avatars and the content of the actual blog pages. Out of 3,100 posts, 1,588 (51.2%) were written by men and 1512 (48.8%) were written by women. The average post length is 250 words for men and 330 words for women.

3.3.3 Experimental setup

Our personality tasks are formulated as five binary classification tasks (e.g. introvert / extravert), one along each of the five personality dimensions. A sixth binary classification task is the gender classification.

Our experimental setup is illustrated in Figure 3.2. There are three possible processing pipelines. These pipelines differ only in the way lexical features are created. In the upper, simplest pipeline (through Bag of words), the documents are segmented to words and each of the words in the training data is used as a binary feature (i.e., present or absent in a document). We further refer to this setup as **WORD**. The subsequent feature selection and classification, specified below, is the same for all pipelines. In the second processing pipeline (through Bag of words - WordNet only), the documents are segmented to words, and the words further annotated with their part-of-speech and lemma. This allows to look them up in WordNet. Only those words, which are present in WordNet, are then use as bag or words features. This intermediary step is helpful to understand which changes in performance can be contributed to the lexicon coverage as opposed to the WSD quality. We further refer to this setup as **WN-WORD**. The third processing pipeline (through Bag of senses) is similar to the previous one, but after the WordNet lookup step performs, in addition, the word sense disambiguation. For each of the words present in WordNet, the resulting sense is then used as a binary feature. This pipeline has three possible configurations, which differ in the WSD algorithm used. We experiment with the most frequent sense baseline (denoted further as **WN-MFS**), Simplified Lesk algorithm (**WN-S-LESK**) and Simplified Extended Lesk algorithm (**WN-S-E-LESK**), explained in the paragraphs below, together with the detailed experimental settings.

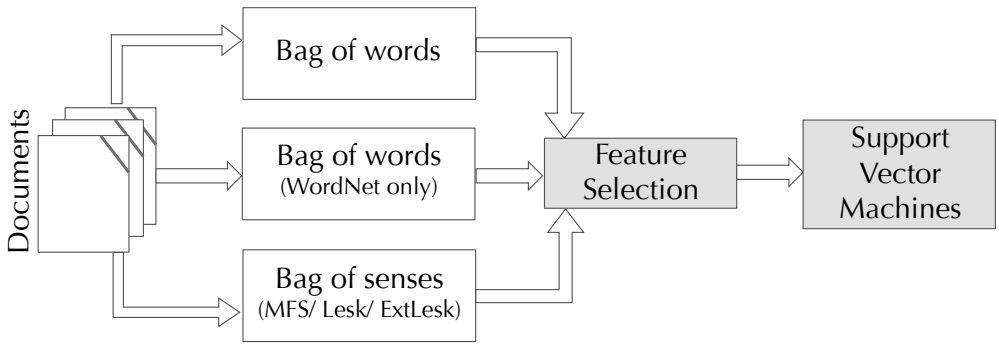

Fig. 3.2: Classification process. Dependent on the experimental settings, we collect for each document either all the words, the words present in WordNet, or all the WordNet senses. The top-ranked 10,000 words or senses in the training data are then selected as binary features.

Linguistic preprocessing The document text is segmented and annotated with part-of-speech tags using the English Stanford CoreNLP toolkit [Manning et al., 2014] accessed through the DKPro text processing API [Gurevych et al., 2007], namely using the Stanford Segmenter, POS Tagger and Lemmatizer.

Word sense disambiguation algorithms We use three WSD algorithms to compare to the bag-of-words setup, using the English WordNet [Miller, 1995] as our lexicon:

- the most frequent sense (MFS) baseline, i.e., from all candidate senses $s \in I(w)$ of a lexicon word $w \in L$ always assigning its first WordNet sense:

$$MFS : L \rightarrow S, w \in L, s \in I(w), MFS(w) = s_1$$

- the Simplified Lesk (SL) algorithm, i.e., comparing each of the possible sense glosses $G(s)$ of a word $w \in L$ to the word context T:

$$SL : L \rightarrow S, w \in L, s \in I(w), SL(w) = \underset{s_i \in I(w)}{\operatorname{argmin}} |G(s_i) \cap T|$$

- the Simplified Extended Lesk (SEL) algorithm – the definitions of the subject of disambiguation are extended with those of neighboring WordNet senses, $N(s)$, in our case its direct hypernyms and hyponyms[2]:

$$SEL : L \rightarrow S, w \in L, s \in I(w), SEL(w) = \underset{s_i \in I(w)}{\operatorname{argmin}} |(G(s_i) \cup G(N(s_i))) \cap T|$$

Feature selection We use the χ^2 feature selection algorithm to select the top 10,000 features before processing the features with a classifier. The feature selection strategy was chosen empirically based on our initial experiments with the personality data, where the χ^2 feature selection outperformed Information Gain, Mutual Information, and Document Frequency thresholding.

Classification algorithm The classification is conducted with the SVM classifier (see Chapter 2.1.2), using its implementation in the SVMLight package [Joachims, 1999]. The choice of the classifier is based on our previous bag-of-words experiments on similar datasets [Flekova and Gurevych, 2013, Flekova et al., 2014b], where the SVM classifier outperformed other tested ones (Naive Bayes, Logistic Regression, Decision Trees).

[2]This version of the algorithm is described in [Miller et al., 2013] and implemented open-source at `https://github.com/dkpro/dkpro-wsd`

Evaluation We evaluate all tasks using 10-fold cross-validation scheme, i.e., evaluating 10 classification models, rotating the 10% test data selection over the dataset. In the experiments presented in this section, we do not perform any additional parameter tuning or feature optimization, as testing all combinations of parameter configurations was beyond the scope of this thesis. Another variation of feature pre-/post-processing and parameter setting may lead to different classification results and shall be verified in future work. We compare our results only with the majority baseline rather than state of the art, as we are interested here in the *difference in performance* between the word-based and the sense-based classification setups, not in the maximal performance achievable[3]. The best results of previous research conducted on these datasets are discussed in the following chapters, where we also extend our feature set.

3.3.4 Results

To get a rough estimate of the performance of the WSD algorithms on our types of data, we manually evaluated the automatic sense attributions of the first 100 target words (i.e., verbs and nouns present in WordNet) in each corpus. The results, produced by one annotator, are listed in Table 3.3.

Dataset	MFS [%]	Simplified Lesk [%]	Simplified Extended Lesk [%]
Essays	81	67	73
Facebook	68	58	61
Twitter	70	60	60
YouTube	69	56	55
Blogs	80	65	69

Tab. 3.3: Rough, approximative WSD performance estimation in each dataset. Percentage of correct senses out of the first 100 disambiguated words evaluated for each of the three WSD algorithms applied - most frequent sense baseline (MFS), Simplified Lesk and Simplified Extended Lesk.

We can see from the table that the approximative overall accuracy of correctly assigned senses is higher than the accuracies reported in all-words sense disambiguation challenges [Palmer et al., 2001, Snyder and Palmer, 2004]. This is caused by several factors. First, our WSD pipeline annotates also frequent verbs such as *to be*, which is typically ignored in the WSD challenges. The verb *to be* accounts for around 20% of our evaluated senses, and has most of the times assigned its sense correctly. Secondly, our pipeline assigns also senses to all the monosemous words, which are present in WordNet. The nouns in WordNet have on average 1.24 senses and verbs 2.17 senses.[4] Thirdly, this approximative evaluation was performed only by one annotator. Given that [Snyder and Palmer, 2004] report an inter-annotator agreement rate of 72.5% on their all-words data, it is likely that

[3] Majority baseline is a sanity check if the algorithm performs better than if the model learned nothing and assigned to all samples the label of the class with the highest number of training documents.

[4] `https://wordnet.princeton.edu/wordnet/man/wnstats.7WN.html`

a disagreement between additional annotators would occur, decreasing the overall score. It is also to be taken into account that the annotator operated in evaluation mode, i.e., manually corrected the automatically assigned senses. This has been shown to lead into lower error rates than generating labels independently [Marcus et al., 1993]. We detail on the last issue further in Chapter 7.

Table 3.3 shows that the most frequent sense (MFS) disambiguation strategy performs better than the Lesk-based algorithms. This is mostly due to its good accuracy on common words, where Lesk-based algorithms often introduce errors by attributing the word to a very unlikely sense. At the same time, if the probability distribution of word senses in the general corpus, used for the MFS calculation (in this case SemCor [Miller et al., 1994]), and the target corpus are different, MFS is likely to fail, while Lesk performs better. For example, in our experiments the Lesk-based algorithms usually deal correctly with the cognitive senses of *see* or *hear* (as in "Oh, I see what you mean.") while the MFS incorrectly assigns those to their perception sense. Such distinctions can be of importance for the personality assessment. We also observe that the performance of all algorithms is higher on the ESSAYS and BLOGS data than in social media. This is partly due to the errors in part-of-speech tagging, partly due to the frequent occurrence of senses which are non-existent or infrequent in WordNet. For example, the verb form *are* is commonly abbreviated to *r* in social media, as in "We r here". This phenomenon is always incorrectly annotated as noun, and subsequently disambiguated as *a unit of radiation exposure*. Similarly, common sentences such as "It will be ok" are problematic, as the most frequent sense of *OK* in WordNet, when incorrectly annotated as a noun, is *Oklahoma, a state in south central United States*.

Below, we present the classification results on each of the datasets, using bag of words/bag of senses as features, comparing the following approaches (explained previously in the Experimental setup section):

- **WORD:** Simple bag of words classification.
- **WN-WORD:** Bag of words classification, where only the words which are present in WordNet are used, in order to examine the impact of the lexicon coverage.
- **WN-MFS:** Bag of senses using the first sense of each word assigned by MFS. Differs from WN-WORD only by POS tagging and lemmatization (in WN-WORD, the POS match is ignored)
- **WN-S-LESK:** Bag of senses using the Simplified Lesk WSD algorithm. In contrast to WN-MFS, the same word can get different senses assigned, dependent on the context.
- **WN-S-E-LESK:** Bag of senses using the Simplified Extended Lesk algorithm, i.e., extended with the hypernymy and hyponymy glosses of each candidate sense.

Data Method:	1. WORD	2. WN-WORD	3. WN-MFS	4. WN-S-LESK	5. WN-S-E-LESK	1.+4. W+S
ESSAYS						
Extraversion	0.546	0.570*	0.580	0.570	**0.589***	0.570*
Agreeableness	0.523	**0.565***	**0.565**	0.556	0.556	0.540*
Conscientiousness	0.606	**0.611**	0.595*	0.505*	0.508	0.576*
Openness	0.642	**0.651**	0.609*	0.642*	0.646	0.642
Neuroticism	0.482	0.554*	0.554	**0.576***	**0.576**	0.482*
FACEBOOK						
Extraversion	0.567	0.592	0.592	0.585	**0.601**	0.585
Agreeableness	0.530	**0.558**	**0.558**	0.550	0.546	0.530
Conscientiousness	**0.568**	0.540	0.540	0.540	0.552	0.568
Openness	**0.721**	0.642*	0.642	0.625	0.613	0.721
Neuroticism	0.563	0.563	0.563	**0.600***	0.567*	0.568
TWITTER						
Extraversion	**0.707**	0.664	0.664	0.643	0.643	0.707
Agreeableness	0.786	**0.807**	0.786	0.786	0.786	0.786
Conscientiousness	0.707	**0.736**	0.729	0.729	0.729	0.707
Openness	0.707	**0.720**	0.714	0.700	0.693	0.707
Neuroticism	**0.800**	0.750	0.729	0.686	0.714	0.800
YOUTUBE						
Extraversion	0.583	**0.593**	**0.593**	0.585	0.583	0.585
Agreeableness	**0.618**	0.556*	0.556	0.572*	0.546*	0.598*
Conscientiousness	**0.774**	**0.774**	**0.774**	0.762	0.772	0.774
Openness	0.605	**0.623***	**0.623**	0.605*	0.595	0.605
Neuroticism	0.556	0.567	0.567	**0.569**	**0.569**	0.556
GENDER	0.656	0.624*	0.591*	0.618*	0.618	0.656*

Tab. 3.4: Classification accuracy for the five personality traits across four datasets and the gender prediction on the fifth dataset, in five different configurations - bag of words (WORD), bag of WordNet words (WN-WORD), bag of senses with different word sense disambiguation algorithms (MFS, S-LESK, S-E-LESK), and bag of words combined with S-LESK bag of senses. The standard error of the 10-fold crossvalidation measurements ranges between 0.017 – 0.036 for the largest (ESSAYS) dataset. A star(*) denotes results which differ significaltly from the results obtained with the setup on their left (we are interested in incremental improvement), using McNemar's two-tailed test on $p < 0.01$. There are no significant differences in the small TWITTER dataset.

- **W+S:** Combining the bag of words (**WORD**) with the bag of senses using the Simplified Lesk algorithm (**WN-S-LESK**), i.e., the classifier has all features from both of the above-mentioned settings available simultaneously at the training time.

On the first glance, the performance of different configurations in Table 3.4 is rather inconclusive. However, we can observe certain patterns. For example, for the *conscientiousness, openness* and *agreeableness* personality traits, adding any WSD technique constantly decreases the performance across all datasets, while the performance on *extraversion* and *neuroticism* improves in three of the four cases. The restriction to WordNet-only words was helpful in 65% of the cases, especially on the ESSAYS dataset. There is no significant difference in personality classification performance between using Extended Simplified Lesk and the plain Simplified Lesk, however, it is noteworthy that on the FACEBOOK and TWITTER data, both WSD algorithms perform worse than the most frequent sense baseline.

Intriguingly, the combination of bag of words and bag of senses mostly did not outperform the better of the two individual setups, suggesting, that the improvements achieved through WSD might be largely due to the vocabulary restrictions.

3.3.5 Error analysis

In the classification results, we observe four phenomena that are worth further exploration:

1. The performance of the most frequent sense WSD (WN-MFS) is sometimes (especially for the TWITTER data) worse than using the original forms of the same words (WN-WORD), although the sense assigned remains constant.
2. The restriction to WordNet words (WN-WORD vs. WORD) helps in 3 out of 4 datasets for predicting *openness* and *agreeableness*.
3. The Lesk algorithms perform worse than the most frequent sense baseline on social media data.
4. The positive effects of WSD are the highest for *neuroticism* (3 out of 4 datasets) and *extraversion* (2 out of 4 datasets).

These effects can be better understood by examining the individual features (words, senses) obtaining the highest scores in the feature selection process. In the table 3.5, we list the highest ranked features for *extraversion* on the ESSAYS dataset, using the χ^2 feature selection.

WORD vs. WN-WORD We observe that using the all-words approach, many of the top-ranked features are pronouns. These are removed when filtering for WordNet words only, as WordNet focuses mainly on nouns, and to a lesser extent adjectives, verbs and adverbs (see Table 2.1). Interestingly, removing these high-ranked features such as pronouns, particles, and punctuation, increases the accuracy on the ESSAYS dataset in all cases, while for other datasets the impact is inconclusive. One possible explanation is that the ESSAYS are written in a more thoughtful manner, focused on the inner processes. They may therefore carry more personality-related information in the content words than the social media data, where the interjection and smileys are more revealing than the topic of the discussion. Filtering for WordNet words thus helps in the essays in a similar way as removing stopwords.

WN-WORD vs. WN-MFS We also observe that the drop in performance between the WN-WORD and the WN-MFS setup is influenced by two factors. The first one is the impact of lemmatization - for example, notice that in table 3.5 the word *thinks* scores among the top WN-WORD features, while the first sense of *think* in the WN-MFS setup is rated much lower

WORD	χ^2	WN-WORD	χ^2	WN-MFS	χ^2	WN-S-LESK	χ^2
love	.012	love	.026	love$_1v$	.016	love$_1v$	.017
boyfriend	.008	music	.010	music$_1n$	.009	assignment$_1n$	.009
'd	.008	sleep	.009	guy$_1n$	.009	sleep$_1v$	.008
me	.007	assignment	.009	good$_1a$	.009	stress$_4n$	.007
so	.006	proud	.008	proud$_1a$	.008	love$_1n$	.006
people	.005	boyfriend	.007	assignment$_1n$	.008	sleep$_1n$	.006
much	.006	worry	.007	boyfriend$_1n$	.008	music$_1n$	.005
we	.005	people	.007	real$_1a$	.006	good$_6a$	.005
thinks	.005	awkward	.007	sleep$_1v$	.006	proud$_3a$	.004
I	.004	stress	.006	view$_1n$	.004	awkward$_5a$	.004
you	.003	thinks	.006	people$_1n$	.004	view$_1n$	.003

Tab. 3.5: The highest ranked features for Extraversion on the ESSAYS dataset, averaged across the 10 cross-validation folds, using the χ^2 feature selection.

(in this case, the predictive feature is likely the usage of 3rd person singular pronouns, while the verb is merely an artifact of it). The second effect is the part-of-speech (POS) tagging. While in the WN-WORD setup, the multi-POS use of a word is rewarded, i.e., different POS of a word form are collapsed, in the WN-MFS setup the first sense is selected after resolving the POS. This separation impacts the feature ranking in many cases. For example, in the WN-WORD setup, the word *worry* is ranked to predict extraversion with χ^2 = .007, while the sense *worry*$_1v$ is ranked to predict introversion with χ^2 = -.004.

Next, we have a closer look at the drop in performance between the WORD and the WN-WORD setup, and further drop in WN-MFS and WN-S-LESK configurations, for the *openness* trait on FACEBOOK. The most predictive features are listed in table 3.6.

WORD vs. WN-WORD We can see that some of the most predictive feature in the WORD setup are pronouns and conjunctions, which are removed in the WN-WORD settings. Those features, which remained in the WN-WORD setup, obtain lower individual scores overall, suggesting that in contrast to the ESSAYS dataset we may have removed an important piece of information.

WN-WORD vs. WN-MFS In the WN-MFS setup, additional errors appear to be introduced by lemmatization, for example transforming the word *stuck* into *stick*, which is likely used in two different contexts (e.g. *I am stuck* vs. *We shall stick together*).

WN-MFS vs. WN-S-LESK Comparing the WN-MFS with the WN-S-LESK, we observe two intriguing phenomena. First, many senses are assigned erroneously in the Facebook context, for example the word *screen* often obtains the sense *screen*$_6n$, defined as *"the personnel of the film industry"*. This is likely due to the word overlap with expressions such as *"watching this film on a large screen"*. Similarly, the word *profile* is assigned the sense *profile*$_2n$, *"an outline*

WORD	χ^2	WN-WORD	χ^2	WN-MFS	χ^2	WN-S-LESK	χ^2
they	.021	music	.019	music$_1 n$	.014	wonderful$_1 a$	.015
as	.020	mood	.018	addicted$_1 a$	.012	perfect$_1 a$	.014
mood	.018	addicted	.015	mood$_1 n$	.011	music$_1 n$	.013
music	.018	wonderful	.015	wonderful$_1 a$	.011	tomorrow$_1 a$	.011
you	.016	perfect	.014	perfect$_1 a$	.010	watch$_3 v$	.009
shall	.016	think	.013	tomorrow$_1 a$	.008	food$_1 n$	.007
if	.015	stuck	.012	admit$_1 v$	.008	love$_1 n$	.005
was	.014	admit	.011	stick$_1 v$	.007	friday$_1 n$	.004
perfect	.014	tomorrow	.011	think$_1 v$	.007	workout$_1 n$	.004
addicted	.013	watch	.011	watch$_1 v$	.006	screen$_6 n$	.003
wonderful	.013	play	.008	love$_1 n$	.006	profile$_2 n$	.003
think	.013	love	.007	time$_1 n$	.005	minute$_1 n$	.003
of	.010	sense	.006	look$_1 v$	.004	love$_1 v$	.003

Tab. 3.6: The highest ranked features for Openness on the FACEBOOK dataset, averaged across the 10 cross-validation folds, using the χ^2 feature selection

of something, especially a human face as seen from one side". This likely originates from the overlap with expressions such as *"your profile on Face"*, meaning Facebook. The second phenomenon is the decrease in ranking of highly polysemous words. While in the WN-MFS setup the words such as *think, watch, time* and *look* are ranked as highly predictive, the very fine granularity of the senses of these words results in lower predictive power of the individual disambiguated features. For example, consider the WordNet examples of these three different senses of the verb *watch*:

- Watch$_1$: "watch a basketball game"
- Watch$_3$: "view a show on television"
- Watch$_6$: "Watch how the dog chases the cats away"

In the social media context, it is often not clear if somebody watches a game physically or on television. Distinguishing between these senses therefore often results to an arbitrary assignment, diminishing the predictive power of the feature.

An additional observation we make is that on TWITTER the drop in performance between the WN-WORD and WN-MFS setup is likely caused by the frequent errors in the part-of-speech tagging. The difference between MFS and Lesk-based strategies is relatively low on Twitter, since most of the senses are not matched to any of the WordNet definitions based on their Twitter context and therefore default to the MFS disambiguation strategy.

Another visual perspective on the WSD behavior is available for selected words in Figure 3.3. The upper part shows a stacked χ^2 score for all detected senses of a word in the WN-S-LESK setup, while the lower part of the figure displays the χ^2 scores of the original words in the WN-WORD setup. We can see that for example for the word *love*, the cumulated score of all

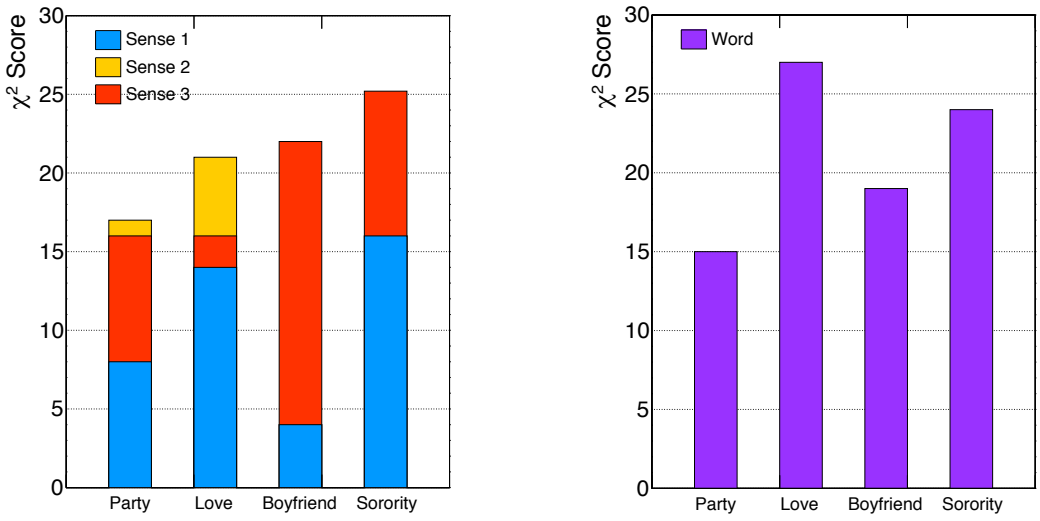

Fig. 3.3: Comparison of cumulated predictive power of disambiguated senses (left) vs predictive power of original words (right) for the Extraversion classification task on the Essays dataset. Vertical axis displays the feature ranking expressed as χ^2 feature ranking times 100. Histogram columns represent word features in the right plot, or sense features for all identified senses of the same word in the left plot.

senses together is lower than the score of the original word. We hypothesize that this is due to the too fine granularity of the senses, introducing unnecessary distinctions. Grouping all such similar senses into one word is then helpful for the classifier in a similar manner as grouping similar words into one topic.

WORD and WN-S-LESK vs. W+S The combination of bag of words and bag of senses does not outperform the individual settings (FACEBOOK, TWITTER). We hypothesize this is due to the fact that naively combining both feature sets neutralizes the vocabulary filtering effect (which we saw for example between WORD and WN-WORD).

3.3.6 Summary of the resource-based WSD experiments

We conducted six binary classification tasks in five different WSD/non-WSD settings applied to five distinct corpora. We found that the sense disambiguation per se does not generally lead to an improvement in classification results except of arbitrary dataset-specific differences, which can be largely attributed to the lemmatization and part-of-speech tagging. However, in contrary to previous beliefs [Sanderson, 1994, Gonzalo et al., 1998], the performance of the WSD algorithms is not the major issue for stagnating performance. Rather, it is the reduction of the representative scope of bag of words (since function words are not present in the lexicon) and the reduction of the impact of multi-POS words (since those are assigned different senses), which leads to a lower ranking of otherwise highly

predictive features. However, while the effect of WSD itself in a bag-of-words setup is marginal, we observe that the WSD quality is rather high. This implies that the assigned senses can be reliably used to query additional information about the word meaning (and relations to other words) from the lexical-semantic resources. We investigate this potential in the following chapters. We also observed that the above tasks are sensitive to non-content words that are predictive of style. In the future, the experiment should be extended to additional text categorization datasets with topic classes rather than demographic or psychological classes for comparison.

3.4 Experiments: Distributional WSD for document classification

We also attempted to employ WSD in the sentiment prediction task in a similar fashion, i.e., disambiguating a sense of a word before assigning it a sentiment score for example from SentiWordNet [Esuli and Sebastiani, 2006]. However, it soon became apparent that the WordNet sense disambiguation is often irrelevant for this purpose [Flekova et al., 2014a]. For example, consider the words *warm* and *cold*, which are often associated with a positive, respectively negative sentiment (*a warm welcome, a cold person*) in sentiment lexicons. The resource-based WSD approach works well for the prototypical cases, mentioned above (*warm welcome* would be a WordNet sense $warm_2a$, *cold person* is $cold_2a$). However, let's consider expressions such as *cold beer* vs. *cold coffee* or *warm cup of tea* vs. *warm beer*. While the WordNet sense is in both cases the same ($cold_1a$ and $warm_1a$, respectively), the polarity of these expressions comes from the external knowledge, i.e. a cup of tea is supposed to be warm, therefore positive, while a beer is not supposed to be warm, therefore negative. Furthermore, there is a mismatch between the formality of many language resources, such as WordNet, and the extremely informal language of social media [Volkova et al., 2013].

Word-based sentiment lexica are very popular in the task of sentiment polarity prediction [Rosenthal et al., 2014, Rosenthal et al., 2015]. However, the ambiguity and meaning shift of the words contained in such lexica can be harmful for the performance on a new domain. We propose a customized approach to disambiguate the meaning of such sentiment-bearing words and quantify their polar ambiguity, not with WordNet senses, but with contextual word pairs (bigrams). We build upon the hypothesis of [Kilgarriff, 1997], who suggests that "a task-independent set of word senses for a language is not a coherent concept", and sense distinctions shall be defined "relative to a set of interest" as determined by the NLP application. Similarly, also [Krovetz, 2002] argues that "different language applications need different sense distinctions". Additionally, [Volkova et al., 2013] specifically point out that "for polarity classification, most errors happen because of relying on either positive or negative polarity scores for a term but not both. However, in the real world, terms may sometimes have both usages."

We therefore propose an approach which helps to quantify the suitability of polar words from a given generic lexicon to a specific target domain by disambiguating these and expressing their polar ambiguity and orientation.

This experiment was conducted together with the PhD student Eugen Ruppert from the Language Technology group at Universität Hamburg. The work was split in the following way: the Language Technology group provided the Twitter corpus and computed the Twitter Bigram Thesaurus in 3.4.2 using the JoBim [Biemann and Riedl, 2013] software, while we have conducted the sentiment classification experiments and the human annotation studies in 3.4.3, performed the analysis of the sentiment bigrams and made the resulting lexicon adjustments.

3.4.1 Background

Sentiment research has tremendously expanded in the past decade. Sentiment prediction is of the utmost interest for researchers as well as commercial organizations, especially in social media. Recently, state-of-the-art sentiment prediction performance has been achieved using supervised neural network models without an additional semantic input [Socher et al., 2013b, Severyn and Moschitti, 2015]. However, recent sentiment prediction challenges show that the vast majority of currently used systems is still based on supervised learning techniques with the most important features derived from pre-existing sentiment lexica [Rosenthal et al., 2014, Rosenthal et al., 2015].

Sentiment lexica were initially developed as general-purpose resources [Pennebaker et al., 2001, Strapparava et al., 2004, Hu and Liu, 2004, Wilson et al., 2005]. After the initial boom of explicit, manually crafted sentiment lexica [Kim and Hovy, 2004, Pang and Lee, 2004, Hu and Liu, 2004], there have been efforts to infer the polarity lexica automatically [Turney and Littman, 2003]. As the sentiment lexica were later shown as unstable across time and domain [Cook and Stevenson, 2010, Mitra et al., 2014, Dragut et al., 2012], an increasing amount of work focused on domain-specific lexica such as Twitter [Mohammad, 2012, Mohammad et al., 2013a, Choi and Cardie, 2009]. However, even customized domain-specific lexica still suffer from ambiguities at a contextual level, such as those in our introductory examples.

Linguistic approaches are used to discover the interaction between words that may switch a sentiment polarity of a sentence [Wilson et al., 2005, Wilson et al., 2009]. Negation and its scope has been studied extensively [Moilanen and Pulman, 2008, Pang and Lee, 2004, Choi and Cardie, 2009]. Lists of contextual polarity shifter words, diminishers and intensifiers, such as *barely* or *hardly*, were often proposed to adjust the sentiment on phrase level [Wilson et al., 2005, Ikeda et al., 2008, Taboada et al., 2011, Polanyi and Zaenen, 2006, Kennedy and Inkpen, 2006, Steinberger et al., 2012, Danescu-Niculescu-Mizil et al.,

2009, Wiegand and Klakow, 2010, Li et al., 2010]. Polarity modifiers, however, do not distinguish cases such as *cannot be bad* from *cannot be worse*.

Polar words can even carry an opposite sentiment in a new domain, e.g., *unpredictable movie plot* vs. *unpredictable dishwasher* [Blitzer et al., 2007, Andreevskaia and Bergler, 2006, Schwartz et al., 2013, Wilson et al., 2005]. Recent experiments revealed that some nouns can carry sentiment per se (e.g. *chocolate*, *injury*). Subsequently, several noun connotation lexicons have been built [Feng et al., 2013, Klenner et al., 2014] based on a set of seed adjectives. One of the biggest disadvantages of such lexicons, however, is that they rely on either positive or negative score of a word, while in reality it can be used in both contexts [Volkova et al., 2013].

3.4.2 Our method

We propose an approach to assessing the ambiguity and semantic orientation of polar words in an established sentiment lexicon, thereby determining their suitability for a new domain. We achieve this by leveraging automatically collected data approximating sentiment labels (silver standard). We present a method for creating switched polarity bigram lists to explicitly reveal and address the issues of a lexicon in question (e.g. the positivity of ***cold*** *beer,* ***dark*** *chocolate* or ***limited*** *edition*, and the ambiguity of ***like*** or ***just***), including words, for which the polarity switch does not necessarily happen on sense level, but within one word sense. We demonstrate that the explicit usage of such inverse polarity bigrams and replacement of the words with high ambiguity improves the performance of the classifier on unseen test data and that this improvement exceeds the performance of simply using all in-domain bigrams. Further, our bigram ranking method is evaluated by human raters, showing correlation with human sentiment judgment.

The key ability of our method is identifying the ambiguous (positive and negative in the new domain at the same time) and incorrect (e.g., positive in lexicon, negative in the new domain) sentiment bearing lexicon unigrams based on the contexts they appear in. The individual steps of our method, detailed in the remainder of this section, are roughly summarized in Algorithm 1:

The input for the method is a generic sentiment polarity lexicon containing positive and negative unigrams. We demonstrate our approach on two polarity lexicons consisting of single words, namely the lexicon of Hu and Liu [Hu and Liu, 2004], further denoted **HL**, and the **MPQA** lexicon [Wilson et al., 2005].

A second ingredient for the disambiguation is a small labeled corpus from the target domain (the *SentiCorpus* in 1). We use a corpus of automatically collected Twitter sentiment dataset of over one million tweets to generate bigrams and compute bigram polarities for the given

Algorithm 1 Modify sentiment unigram lexicon for new domain

```
unigrams ← Read all word unigrams from the sentiment polarity lexicon
SentiCorpus ← an in-domain corpus labeled with sentiment
LargeCorpus ← a large, unlabeled in-domain corpus
for word ∈ unigrams do
    sentiBigrams ← add all bigrams from SentiCorpus containing the word,
Compute bigram frequencies and mutual information (LMI) at LargeCorpus
for bigram ∈ sentiBigrams do
    combine bigrams from SentiCorpus with their LMI score from LargeCorpus
if unigram polarity != ∑ (bigram polarity * LMI) then
    if ∑ (bigram polarity * LMI) < ambiguityThreshold then
        Replace lexicon unigram with all its bigrams (unigram is ambiguous)
    else if ∑ (bigram polarity * LMI) ≤ ambiguityThreshold then
        Assign unigram opposite polarity in the lexicon
        (in the new domain, the unigram is more often used in the other polarity context)
return updatedLexicon
```

seed words from the lexicon and determine contexts which alter the polarity of the original lexicon word.

As a third ingredient, we then use a large unlabeled Twitter corpus (the *LargeCorpus* in Algorithm 1), from which build a large Twitter bigram thesaurus which serves as a background frequency distribution which aids in ranking the bigrams, i.e. correcting for the information obtained from the small supervised sentiment corpus.

For each of the lexicon unigrams, we examine if its in-domain bigrams have mostly the same polarity, mostly the opposite polarity, or are ambiguous in their semantic orientation. We then move the opposite-polarity unigrams to the other polarity list, and we replace the ambiguous unigrams in the lexicon with their bigrams.

In our sentiment prediction experiments, we compare this technique with a straightforward usage of all bigrams generated from the lexicon unigrams on the supervised corpus, and show that our approach helps to achieve a better balance between precision and recall.

Below, we describe the details of our implementation.

Calculating the mutual information of bigrams on a large unlabeled corpus

Twitter Bigram Thesaurus Methods based on word co-occurrence have a long tradition in NLP research, being used in tasks such as collocation extraction or sentiment analysis. [Turney and Littman, 2003] used polarity seed words to measure the semantic orientation of words which co-occur in the same contexts. To determine the probability of a word *w* in a context *c*, they use the Pointwise Mutual Information (PMI), defined by Equation 3.1. The equation reflects the frequency *f(w)* of the word and *f(c)* of the context. However, the PMI

is known to be sensitive to low count words and bigrams, overemphasising them over high frequency words. To account for this, we express the mutual information of a word bigram by means of Lexicographer's Mutual Information (LMI), characterized in the Equation 3.2.[5] The LMI, introduced by [Kilgarriff et al., 2004], offers an advantage to PMI, as the scores are multiplied by the bigram frequency, boosting more frequent combinations of word (w) and context (c).

$$\text{PMI}(w,c) = \log_2\left(\frac{f(w,c)}{f(w)\cdot f(c)}\right) \tag{3.1}$$

$$\text{LMI}(w,c) = \text{PMI}(w,c)\cdot f(w,c) \tag{3.2}$$

Computing bigram sentiment scores

We compute the LMI over a corpus of positive, respectively negative tweets, in order to obtain positive (LMI_{pos}) and negative (LMI_{neg}) bigram scores. We combine the following freely available data, leading to a large corpus of positive and negative tweets:

- 1.6 million automatically labeled tweets from the Sentiment140 dataset [Go et al., 2009], collected by searching for positive and negative emoticons;
- 7,000 manually labeled tweets from University of Michigan;[6]
- 5,500 manually labeled tweets from Niek J. Sanders;[7]
- 2,000 manually labeled tweets from the STS-Gold dataset [Saif et al., 2013].

We filtered out ca. 30,000 fully duplicate messages, as these appear to bring more noise than realistic frequency information. The resulting corpus contains 794,000 positive and 791,000 negative tweets. In pursuance of comparability between the positive and negative LMI scores, we weight the bigrams by their relative frequency in the respective dataset, thus discounting rare or evenly distributed bigrams, as illustrated for the negative score in:

$$\text{LMI}_{neg_{REL}}(w,c) = \text{LMI}_{neg}(w,c)\cdot \frac{f_{neg}(w,c)}{f_{neg}(w,c)+f_{pos}(w,c)}$$

The LMI scores are considered more reliable when coming from a larger corpus, e.g. [Turney and Littman, 2003] uses a corpus of one hundred billion words. The sentiment-annotated tweets account merely for 16 million words. We therefore boost the scores obtained on the annotated dataset by incorporating LMI scores from a background corpus (LMI_{GLOB}) –

[5] An online demo illustrating the score values and distributional term similarities in this Twitter space can be found at the website `http://maggie.lt.informatik.tu-darmstadt.de/jobimviz/`

[6] `http://inclass.kaggle.com/c/si650winter11/data`

[7] `http://www.sananalytics.com/lab/twitter-sentiment/`

described below. This approach emphasizes significant bigrams, even when their score in one polarity dataset is low:

$$\text{LMI}_{neg_{GLOB}}(w,c) = \text{LMI}_{neg_{REL}}(w,c) \cdot \text{LMI}_{GLOB}(w,c)$$

As background data we use a Twitter corpus of 1 % of all tweets from the year 2013, obtained through the Twitter Spritzer API. We filtered this corpus with a language filter,[8] resulting in 460 million English tweets, i.e. around 4-5 billion words. To compute the LMI scores for large volumes of data, we use the JoBimText framework [Biemann and Riedl, 2013]. JoBimText uses a MapReduce model to compute a distributional thesaurus, where word similarities are obtained by establishing the context overlap of two words. While the framework allows to define contexts of arbitrary length, for our experiments we chose a bigram model where the previous and the next word are used as context features.

For each bigram, we then compute its semantic orientation, similarly to the PMI_{SO} introduced in [Turney and Littman, 2003]:

$$\text{LMI}_{SO} = \text{LMI}_{pos_{GLOB}} - \text{LMI}_{neg_{GLOB}}$$

These two large bigram lists, which at this point still contain all bigrams from the Twitter sentiment corpus, are then filtered by sentiment lexica, as we are only interested in bigrams with at least one word from the original sentiment lexicon (containing single words). We chose two sentiment polarity lexica for our experiments:

- the **HL** lexicon [Hu and Liu, 2004] having 4,782 negative and 2,004 positive words (e.g. *happy*, *good*, *bad*);
- the **MPQA** sentiment lexicon [Wilson et al., 2005], with 1,751 positive and 2,693 negative words.[9]

The most interesting candidates for a novel bigram sentiment lexicon are:

- bigrams containing a word from a **negative** lexicon, which has a **positive semantic orientation** LMI_{SO}, i.e. having higher global LMI in the positive dataset than in the negative;
- bigrams containing a word from a **positive** lexicon with **negative semantic orientation** LMI_{SO}

[8] `https://github.com/shuyo/language-detection`

[9] This lexicon also contains neutral words, which might be interesting for some applications. Since the **HL** lexicon does not feature neutral words, we chose to omit those entries for comparable results.

Negative Word to Positive Context				Positive Word to Negative Context			
HL		MPQA		HL		MPQA	
Word	Context	Word	Context	Word	Context	Word	Context
limit	why-	vice	-versa	luck	good-	super	-duper
sneak	-peek	stress	-reliever	wisdom	-tooth	happy	-camper
impossible	mission-	down	calmed-	well	oh-	just	-puked
lazy	-sunday	deep	-breath	work	gotta-	heart	-breaker
desperate	-housewives	long	-awaited	hot	-outside	gold	-digger
cold	-beer	cloud	-computing	better	feels-	light	-bulbs
guilty	-pleasure	dark	-haired	super	-tired	sincere	-condolences
belated	-birthday	bloody	-mary	enough	-money	frank	-iero

Tab. 3.7: The highest scoring bigrams with opposite LMI sentiment orientation than the original lexicon word. Note that the polarity rarely changes on sense level i.e., same sense can have different polar contexts. Some bigrams, detected as used mostly in differently oriented contexts, are rather neutral, e.g. *light bulbs* as opposed to the positive *light*. Some reflect a broader typical context of the tweets, for example the bigram *happy camper* is usually used in the idiomatic phrase *not a happy camper*.

The top ranked bigrams, where local contextualization reverts the original lexicon score, are listed for both lexicons in Table 3.7. We can observe that the polarity shifting occurs in a broad range of situations, e.g. by using polar word as an intensity expression (***super*** *tired*), by using multiword expressions, idioms and collocations (***cloud*** *computing,* ***sincere*** *condolences,* ***light*** *bulbs*), by using polar word in names (e.g., ***desperate*** *housewives* is rated as negative although it is a name of a film series, usually discussed in a positive tweet), but also by adding a polar nominal context to the adjective (***cold*** *beer/person,* ***dark*** *chocolate/thoughts,* ***stress*** *reliever/management,* ***guilty*** *pleasure/feeling*). Note also the bigrams such as *happy camper*, *looking good* or *enough money*, which are explicitly positive, but have learnt their negative connotation from a broader context, as they are typically used with negations.

Quantifying an impact of the polarity switch

We have shown how to identify words which switch to the opposite polarity based on their word context. Our next goal is to identify words which occur in many contexts with both the original and the switched polarity (such as *just* or *right*) and therefore are, without further disambiguation, harmful in either of the polarity lists. With this aim we calculate a polarity score POL_{word} for each word (w) in the polarity lexicon, using the number of its positive and negative contexts determined by their semantic orientation LMI_{SO} as previously computed:

$$\text{POL}(w) = p_{pos}(w) - p_{neg}(w)$$

where we define $p_{pos}(w)$ and $p_{neg}(w)$ as the count of positive and negative bigrams respectively, of a lexicon word, divided by the count of all bigrams of that word:

$$p_{neg}(w) = \frac{\sum(w,c)_{\forall(w,c):\text{LMI}_{SO}<0}}{\sum(w,c)}$$

Lexicon words with the lowest absolute polarity score and the highest number of different contexts (w,c) are listed in Table 3.8. Typically, these are the words which are highly polysemous, e.g. the word forms *right, hot, top, support, back and down* can refer to over 20 WordNet senses each. However, some of the expressions are likely domain specific, e.g. the negativity of *super* (as in *super dumb*) or the positivity of *hell* (as in *hell of a movie*), which both have only five WordNet senses. Also the words *proper* and *enough*, listed as positive in the HL lexicon, are, in the Twitter case, used in both positive and negative contexts despite their low degree of polysemy (4 and 3, respectively).

HL					MPQA				
Word	POL(w)	$\#(w,c)_{pos}$	$\#(w,c)_{neg}$	orig	Word	POL(w)	$\#(w,c)_{pos}$	$\#(w,c)_{neg}$	orig
hot	.022	1151	1101	+	just	-.002	742	738	+
support	.022	517	494	+	less	.009	51	50	-
important	-.023	204	214	+	sound	-.011	43	44	+
super	-.043	734	801	+	real	.027	35	37	+
crazy	-.045	809	886	-	little	.032	354	332	-
right	-.065	3061	3491	+	help	-.037	42	39	+
proper	-.093	242	292	+	back	-.046	191	174	+
worked	-.111	275	344	+	mean	.090	24	20	-
top	.113	516	411	+	down	-.216	154	239	-
enough	-.114	927	1167	+	too	-.239	252	411	-
hell	.115	616	488	-					

Tab. 3.8: Most ambiguous sentiment lexicon words. POL(w) displays the overall semantic orientation of a word weighted by the absolute number of its positive and negative contexts. *orig* shows the original polarity of the word in the examined sentiment lexicon.

3.4.3 Intrinsic and extrinsic evaluation

To evaluate the quality of our bigrams, we perform two studies. First, we rate our inverted polarity bigrams intrinsically using crowdsourced annotations. Second, we assess the performance of the original and adjusted lexicons on a distinct expert-constructed dataset of 1,600 Facebook messages annotated for sentiment. The disambiguated bigram lexicons are available on our website [10].

[10] https://www.ukp.tu-darmstadt.de/data/sentiment-analysis/inverted-polarity-bigrams/

HL	Positive	Neutral	Negative	**MPQA**	Positive	Neutral	Negative
Positive	30	10	9	Positive	21	24	3
Negative	11	10	30	Negative	5	18	25

Tab. 3.9: Confusion matrix for the majority vote of word polarity as labeled by three crowdsourced annotators. For each of the 100 bigrams, annotators could select from the options *positive*, *negative* or *neutral*.

Intrinsic Evaluation

We crowdsource ratings for the inverted polarity bigrams found using both the **HL** and **MPQA** lexicon. The raters were presented a list of 100 bigrams of each lexicon, with 25% having the same positive polarity as in the original lexicon, 25% the same negative polarity, 25% switching polarity from positive unigram to negative bigram and the remaining quarter vice versa. They had to answer the question 'Which polarity does this word pair have?', given *positive*, *negative* and also *neutral* as options. Each bigram is rated by three annotators and the majority vote is selected. The inter-annotator agreement is measured using weighted Cohen's κ [Cohen, 1968], which is especially useful for ordered annotations, as it accounts not only for chance, but also for the level of disagreement between annotators. κ can range from -1 to 1, where the value of 0 represents an agreement equal to chance while 1 equals to a perfect agreement, i.e. identical annotation values. We obtained an agreement of weighted Cohen's $\kappa = 0.55$, which represents a "moderate agreement" [Landis and Koch, 1977]. The confusion matrix of our computed bigram polarity compared to human labels (obtained as a majority vote from the three judgments per bigram) is shown in Table 3.9. Some of the bigrams, especially for the MPQA lexicon, were assessed by human judges as *neutral*, an option which our LMI method unfortunately does not reflect beyond the score value (neutral words are less polar in their score - introducing a minimum threshold for the score could be an option to address this issue). However, as it can be seen from the Table 3.9, the confusion between negatively and positively labeled bigrams was quite low (8% of misjudged words in MPQA, 20% in HL). The errors were mostly originating from the expressions that were considered negative by the system due to the typically negative broader tweet context, such as the (typically *not a*) *happy camper* or the (typically *not*) *enough money*.

Extrinsic Evaluation

We evaluate our method on a dataset of Facebook posts annotated for positive and negative sentiment by two psychologists [Preotiuc-Pietro et al., 2016]. The posts are annotated on a scale from 1 to 9, with 1 indicating strong negative sentiment and 9 indicating strong positive sentiment. An average rating between annotators is considered to be the final message score. Ratings follow a normal distribution, i.e. with more messages having less

polar score. An inter-annotator agreement of weighted Cohen's $\kappa = 0.61$ on exact score was reached, representing a "substantial agreeement" [Landis and Koch, 1977]. Given our task, in which we attempt to improve on misleading bipolar words, we removed the posts annotated as neutral (rating 5.0). This left us with 2,087 posts, of which we use only those containing at least one word from the polarity lexicons of our interest, i.e., 1,601 posts for **MPQA** and 1,526 posts for **HL**. Both of the resulting sets have a mean human sentiment score of 5.3 and a normal distribution of the ratings with a standard deviation of 1.04. We then estimate a sentiment score of a post as a difference of positive and negative word counts present in the post. If a bigram containing the lexicon word is found, its LMI_{SO} score is used instead of the lexicon word polarity score. For the two lexicons and their modifications, we employ two evaluation measures - Pearson correlation of the sentiment score of a post with the affect score, and classification accuracy on binary label, i.e., distinguishing if the affect is negative (score ≤ 4.5) or positive (score ≥ 5.5). Table 3.10 presents the results of our experiments, using the following features in their positive, negative and combined ablations:

- **Unigrams:** using the original unigram lexicon only (configuration 1 in the table);
- **Unigrams + Bigrams:** using original lexicon corrected by polarity score of lexicon bigrams when the bigram is found in the text, i.e. a polarity of $+0.8$ would be used for *lazy sunday*, otherwise a polarity of -1 is used for *lazy* (configuration 2–4 in the table);
- **Pruned:** using pruned unigram lexicon, removing words that exceed weighted ambiguity threshold of 0.99, i.e., appear in many positive and negative contexts with similar frequency (e.g. *just, hot*), and removing the words which appear in more contexts of the opposite polarity than of the one assumed in the lexicon, (e.g. *guilty, impossible*) (configuraton 5 in the table);
- **Pruned + Bigrams:** using pruned unigram lexicon corrected by polarity score of (unpruned) lexicon bigrams when they appear in the assessed text (configurations 6–8 in the table);
- **All in-domain bigrams** learnt from a Twitter corpus (configuration 9 in the table).

Using only the positive or the negative part of a lexicon is denoted in the Table 3.10 with the $+$ and $-$ symbols.

Table 3.10 shows that adding contextual bigrams brings a consistent improvement (Config. 1 vs. 2 and 5 vs. 6). Especially the negative part of the bigram lexica, including bigrams of negative words which have positive orientation, consistently improves results (Config. 1 vs. 4 and 5 vs. 8). Likewise, pruning of the lexicon for ambiguous words (1 vs. 5) enhances the sentiment prediction performance. For both polarity lexicons, the best performance is achieved by combining the two effects (Config. 8). In case of the HL lexicon, the performance is even higher than in case of applying, to the same data, a fully in-domain bigram lexicon (Config. 9), generated from the same large public Twitter corpus

		HL		MPQA	
Id	Features	Acc.	Corr.	Acc.	Corr.
1	Unigrams	0.707	0.582	0.661	0.447
2	Unigrams + Bigrams	0.722	0.596	0.663	0.448
3	Unigrams + Bigrams_+	0.712	0.593	0.662	0.447
4	Unigrams + Bigrams_-	0.716	0.597	0.662	0.447
5	Pruned	0.723	0.613	0.663	0.482
6	Pruned + Bigrams	**0.733**	0.594	**0.665**	0.492
7	Pruned + Bigrams_+	0.715	0.626	0.663	0.491
8	Pruned + Bigrams_-	0.729	0.633	0.664	0.493
9	All in-domain Bigrams	0.691	0.184	**0.701**	0.181

Tab. 3.10: Predictive performance using lexicon based methods, displaying the classification accuracy and the linear (Pearson) correlation of the gold-label sentiment score to the estimated score based on the unigrams and bigrams. For comparison, the agreement correlation between the two human annotators was $r = .768$. The standard error of the accuracy is below 0.001 in all cases. Using McNemar's two-tailed test, there is a significant difference on p<0.05 level between the runs 1 and 2, 5 and 6 and 1 and 5 for HL, and between the runs 1 and 6 for MPQA.

[Mohammad et al., 2013a]. Possibly, the enforced usage of bigrams is introducing noise through the less frequent and thus less reliable expressions, while our methods encourages the usage of unigrams in cases where the bigram-based polarity disambiguation is not necessary for a lexicon word.

Based on the correlation values in Table 3.10, the lexicon words appear to be better suited for predicting the exact sentiment score. Our interpretation is that while the in-domain bigrams have better coverage for distinguishing the positive from negative messages, their presence or absence does not distinguish between a score 6 and score 9, for example. The lexicon words might be more precise in the sense that they occur more frequently in highly sentimental messages, e.g. scoring 1 or 9.

The correction of negative unigrams to positive bigrams does not improve the prediction as much as its counterpart (see the error analysis discussion below). The main cause appears to be the fact that those expressions with shifted polarity shall be rather neutral (e.g., *dark hair, cloud computing*) - as discussed in our intrinsic evaluation experiments and by some recent research [Zhu et al., 2014].

Error analysis

Using bigrams does not only bring improvement, but sometimes also introduces new errors. One of the frequent sources of errors appears to be the remaining ambiguity of the bigrams due to more complex syntactic constructions. While the bigrams are tremendously helpful in a negative text such as '*holy shit, tech support... help!*', where the *holy* (+1) and *support* (+1) are replaced by its appropriately polar contexts (-0.35, -0.85), the same replacement

is harmful in a post '*holy shit monday night was amazing*'. The same applies for bigrams such as *work ahead* (-0.89) in '*new house....yeah!! lots of work ahead of us!!!*' or *nice outside* (-0.65) in '*it's nice outside today!*'.

Additionally, the performance suffers when a longer negation window is applied, such as *feeling sick* in the post '*Isn't feeling sick woohoo!*'. In our setup, we did not employ explicit polarity switchers commonly used with word lexicons [Wilson et al., 2005, Pang and Lee, 2008, Steinberger et al., 2012] since the negation is often incorporated in the bigrams themselves. This, however, makes it challenging to combine the bigrams with an additional negation heuristics.

Another interesting issue are the bigrams which are explicitly positive but have learnt their negative connotation from a broader context, such as *happy camper* or *looking good*, which are more often used jointly with negations. Posts that use these bigrams without negation ('*someone is a happy camper!*') then lead to errors, and similarly a manual human assessment without a longer context fails. This issue concerns distributional approaches in general.

Lastly, several errors arise from the non-standard, slang and misspelled words which are not present often enough in our training corpus. For example, while *love you* is clearly positive, *love ya* has a negative score, probably due to a small number of negative sentences that happened to contain it. One solution could be further word normalization and optimization of word frequency thresholds so that stopwords are still not prominent but rare and misspelled expressions are penalized.

3.4.4 Summary of the distributional WSD experiments

Lexicon-based methods currently remain, due to their simplicity, the most prevalent sentiment analysis approaches. We propose a method to adapt a general-purpose sentiment lexicon to a target specific target domain, and suggest that using in-domain data selectively for the cases requiring disambiguation in their semantic orientation is sometimes more beneficial than using all in-domain data. Using our method, we (i) identify frequent bigrams where a word switches polarity, and (ii) find out which words are bipolar to the extent that it is better to have them removed from the polarity lexica. We validate our computed bigram sentiment scores by crowdsourced human ratings, and we demonstrate that the modified sentiment lexicons bring improvement in the classification results. Our method helps to gain qualitative insights into the shortcomings of a general-purpose lexicon in a new domain, quantifying ambiguous lexicon words and contexts with an inverse semantic orientation, and address those by an appropriate action on the lexicon. Note that beside the actual domain adaptation, we are able to first determine if there is an actual need for the adaptation and in which way the lexicon is impacted. This can be particularly beneficial for

adapting older resources built on "traditional" corpora, such as newspapers or books, for their usage in modern communication channels, such as social media and online fora, as these are usually the first to capture a semantic change in the meaning of words [Rohrdantz et al., 2011, Kulkarni et al., 2015, Hamilton et al., 2016, Eger and Mehler, 2016], and the character of these changes may be non-trivial to anticipate.

3.5 Chapter summary

In this chapter, we introduces several resource-based and distributional approaches to word sense disambiguation in text classification. While the resource-based approaches are designed to be more general, the distributional approaches can be mainly useful with a specific task and corpus in mind.

In the section 3.3, we have implemented three popular resource-based WSD algorithms, namely the most frequent sense selection, the Simplified Lesk, and the Simplified Extended Lesk disambiguation algorithms. We conducted six binary classification tasks in five different WSD/non-WSD settings applied at five distinct corpora from varied domains. We found that the sense disambiguation per se does not generally lead to an improvement in classification results. However, in contrary to many previous beliefs [Sanderson, 1994, Gonzalo et al., 1998], we argue that the performance of the WSD algorithms is not a major cause of the stagnating performance. Rather, it is the "one sense per discourse" paradigm [Gale et al., 1992], according to which the sense of a word remains the same when repeated within the same document or set of documents. While this means that the WSD itself is largely redundant in the bag-of-words setup, it also implies that the assigned senses can be used rather reliably to query additional information about the word meaning (and relations to other words) from the lexical semantic resources, which is positive news. We investigate this potential in the following chapters.

In the section 3.4, we point out that the resource-based WSD is not a panacea, and identify counter-examples where the traditional sentiment-bearing words can have different sentiment polarity even within one WordNet sense. We then proposed a method to address it. This method enables to identify frequent bigrams where a sentiment lexicon word switches polarity, and to find out which words are bipolar to the extent that it is better to have them removed from the sentiment lexica. We identified four types of semantic orientation switching situations, and demonstrated that our bigram sentiment scores match human perception of polarity and bring improvement in the classification results using our context-aware method. Our method enhances the assessment of lexicon based sentiment detection algorithms and can be further used to quantify ambiguous words. Our results have been published in [Flekova et al., 2014a] and [Flekova et al., 2015b].

Overall, we have found that the word sense disambiguation is mostly beneficial when customized relative to the task, e.g., disambiguating positive and negative meanings of a word in a sentiment classification task, rather than approached in a task-independent, generic manner. This is in line with previous suggestions of [Kilgarriff, 1997] and [Krovetz, 2002]. However, the generic resource-based WSD algorithms show an acceptably high performance to be exploited further in the classification pipeline for querying additional high-level information about the meaning of a given word and its relations.

4 Lexical-semantic Features for Concept Generalization

> *An ocean traveler has even more vividly the impression that the ocean is made of waves than that it is made of water.*
>
> **— Arthur S. Eddington**

In the previous chapter, we have seen that the impact of disambiguating document words, and using them in a bag-of-senses approach instead of bag-of-words, is marginal. However, the information that lexical-semantic resources provide, doesn't stop with the word sense. In contrary, WSD is merely the first step to obtain a point of entry for accessing the information structure in the resource. In this chapter, we propose to use lexical-semantic resources to abstract from individual words to higher-level semantic concepts, as illustrated on figure 4.1. This addresses a different problem than in the previous chapter. In Chapter 3, our main concern was that the words can be polysemous, i.e., have more than one meaning. This turned out to be only a minor factor in document classification tasks. In this chapter, we address the issue that even when the words themselves are monosemous, the same information can be expressed differently in the test data than it was in the training data, using synonymous expressions. Consider for example the sentences *We bought the company* and *We acquired the enterprise*. While the semantic relatedness of the two is apparent to a human, note that the system trained in a bag-of-words manner on the data containing the first sentence has no way to understand the second one, and is likely to fail on it in the classification task. An obvious insight is that we need to provide the system with some kind of information, quantifying that *bought* is related to *acquired* and *company* to *enterprise*, i.e. capturing the semantic relatedness between word pairs.

4.1 Approaches to acquiring abstraction over words

There are several key approaches used in text classification to group individual word features[1] into higher-level concepts based on their meaning. We introduce the main ideas behind them below.

[1] by "features" we understand the text classification features defined in Section 2.3.1

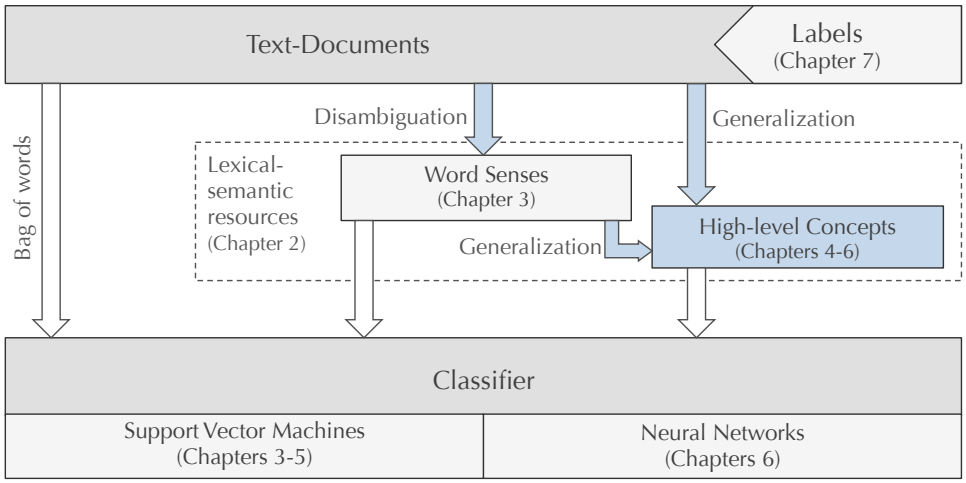

Fig. 4.1: The concepts and workflows of this thesis explored in this chapter (highlighted blue).

Manually curated lexicons

The basic, yet powerful approach is to manually create word lists, i.e., an expert lists all the words related to a certain topic. All occurrences of the words in a list are then counted towards the same lexical-semantic feature, which can be any concept describing the words it contains, for example *Anger*, *School* or *Motion*. Such word lists usually do not make any word sense distinctions, focusing on capturing the word surface form or lemma.

The most prominent examples of this approach are sentiment lexicons, emotion lexicons and lexicons describing psychological phenomena. Sentiment lexicons typically list positive and negative words - such as those we discussed in detail in Section 3.4 [Hu and Liu, 2004, Wilson et al., 2005]. Emotion lexicons expand these to a larger range of emotions beyond the bipolar distinction, separating for example sadness from anger or joy from trust. NRC Emotion Lexicon [Mohammad and Turney, 2013] is one of the most commonly used ones.

Psychological lexicons usually focus on capturing cognitive states. Probably the most popular one is the Linguistic Inquiry and Word Count (LIWC) [Pennebaker et al., 2001], which was developed by researchers with interests in social, clinical, health, and cognitive psychology. The language categories were created to capture people's social and psychological states, and range from cognition word lists such as *Tentativeness, Certainty* or *Insight*, through perpetual processes such as *Hearing, Feeling* or *Ingesting*, to time orientation (*Focus on past, Focus on present*) and topical interests (*Work, Leisure, Money*).

Numerous additional lists of manually curated topics have proven useful for example in author gender prediction [Argamon et al., 2007, Flekova and Gurevych, 2013].

Automatically created topics (word clusters)

Another approach to creating more robust classification systems is to derive latent topics from existing large corpus, which are used as features to enrich the representation of short text. This approach has been particularly useful for expanding short texts in social media [Chen et al., 2011, Li et al., 2008b]. To provide insight into how the distributional topic models are created, we first need to explain the idea of distributional semantic relatedness.

Distributional semantic relatedness. The idea of generating topics automatically originates in the distributional hypothesis [Harris, 1954], according to which the word is characterized by its context. Therefore, similar words can be identified by the fact they are appearing in similar contexts. Methods for using automatically generated contextual features were developed around 1990 mainly in the context of information retrieval. The word co-occurrence matrix computed on a document corpus is used to map words or phrases from a vocabulary to a corresponding vector of real numbers. One of the most influential early models was Latent Semantic Analysis (LSA) [Deerwester et al., 1990], the precursor of today's topic models.[2] Representing words in this way, as vectors in a multidimensional conceptual space, allows to maintain the intuitive concept of semantic relatedness[3] of two words (i.e., a *dog* is semantically closer to a *cat* than to *taxes*) as a distance between the two vectors. In the previous chapter, we have also introduced the PMI and LMI scores, which can be used in a similar manner (the higher the score, the more related the words).

Distributional topic models. Computational topic models developed from LSA as its probabilistic refinements [Hofmann, 1999]. While LSA was mostly intended for assessing the document similarity in information retrieval, topic modeling is typically used with a more exploratory focus. Among the most popular topic models is the Latent Dirichlet Allocation (LDA) [Blei et al., 2003], which assumes dirichlet priors for the document-topic and topic-word distributions. In contrast to the original LSA, the topics here are not assumed to be orthogonal. Since the generated clusters are not named, one of the challenges in this approach is the human interpretability of such topics [Chang et al., 2009]. Another

[2] In parallel, several different models using contextual representations were developed in other research areas, for example Self Organizing Maps [Kohonen and Somervuo, 1998].

[3] Zesch (2010) argues that semantic relatedness is a broader term than semantic similarity, pointing out that while semantic similarity is typically defined via synonymy (*automobile – car*) and hypernymy (*vehicle – car*), and semantic relatedness covers a broader range of relations, e.g. *night – dark* or *high – low*. Whether we refer to semantic similarity or relatedness in this thesis, we have the latter, broader definition in mind.

challenge is defining the appropriate granularity of the topics generated, which usually requires manual parameter tuning.

Word embeddings

The early approaches to distributional vectors posed computational obstacles – the vocabulary size is typically very high (hundred thousands or millions of words). Storing each of these N words in a N-dimensional vector results in a N x N matrix and performing operations on it is computationally heavy. Addressing this issue, [Bengio et al., 2003] propose to reduce the high dimensionality of word representations in contexts by learning a distributed representation for words. This "word embedding" approach became popular only recently thanks to the advances in vector quality and training speed. Word embeddings learned with neural-network based language models have contributed to state-of-the-art results on various linguistic tasks [Collobert and Weston, 2008, Bordes et al., 2011, Mikolov et al., 2013c, Pennington et al., 2014, Levy et al., 2015].

One of the core benefits of word embeddings, beside the low dimensionality, is that they don't require expensive annotation, but only a large unannotated corpus, from which they can be derived in an unsupervised way. Such embeddings can then be used in downstream document classification tasks, using the word vector similarity information implicitly, or they can be used to build topic clusters based on this similarity (as in [Preoţiuc-Pietro et al., 2015], for example), and then treated similarly to the traditional distributional topic models, as we do in Chapter 7 and in [Flekova et al., 2016a].

Beside this chapter, we discuss the recent developments in the area of word embeddings and neural network approaches further in Chapter 6, where we show how these can benefit from the explicit lexical-semantic knowledge and conduct corresponding experiments.

Resource-based semantic relatedness

Another approach to quantifying the semantic relatedness of two words is to use a lexical-semantic resource. One of the advantages of these methods is that they work directly on the sense level, in contrast to the word-level-based ones introduced above. Among the most popular resource-based methods are the path-based measures, which determine the length of the path between nodes representing concepts in a semantic resource (e.g. in the WordNet graph or the Wikipedia category graph). The shorter the path, the higher the relatedness between concepts.

The simple path-length-based methods described above do not take into account that concepts higher in the taxonomy are more abstract, i.e. that a path with a length of 1 between abstract concepts near the top of the taxonomy should yield a lower similarity value than a path of the same length between specific concepts on the leaf level of the taxonomy. To overcome this limitation, [Wu and Palmer, 1994] introduce a measure that uses the notion of a lowest common subsumer of two concepts (LCS). An LCS is the first shared concept on the paths from the concepts to the root concept of the hierarchy. Further enhancements of both of these approaches – the shortest path and the LCS – have been published since, still with the core principle of penalizing concepts which are distant from each other in the hierarchy.

An alternative resource-based approach, introduced by [Lesk, 1986] , is using the gloss word overlap. We explained this strategy in the WSD algorithms in the previous chapter, with the difference that here we are interested not only in the most similar instance, but in an actual score of all senses of interest. This gloss overlap later developed into a vector-based approach, when [Patwardhan and Pedersen, 2006] took all words from the glosses and created a term frequency matrix, processed similarly as in the distributional semantic relatedness described above.

Resource-based abstraction annotation For the text classification applications, the main question is how to express the semantic relatedness of the words as a property of the document, which we can use as a feature for our classifier. We explained in the introduction of this chapter, that we want to bridge the lexical gap between the concepts such as *bought* and *acquired* or *company* and *enterprise*. But how do we do it when we do not have both of these documents available at training time? Resource-based similarity of what to what do we compute? In this thesis, we propose to focus on the level of abstraction defined by supersenses. These are described in more detail in the following section, yet to briefly outline the idea behind our approach, consider the following example. We have a classification task in which we want to label short descriptions of activities with the actions required. For simplicity let's assume we have only two labels: *pet_needs_attention* and *no_action_needed*. The training data contains, among others, this sentence:

- *Cat is sitting in the bathroom, miaowing loud. (pet_needs_attention)*

And the test data contains these two sentences:

- *Dog is lying in the living room, howling loud. (pet_needs_attention)*
- *Anthony is dancing at the disco, singing loud. (no_action_needed)*

Given the first sentence in the training data, the word-based classifier cannot distinguish between the second and the third one in the test data. However, if we add the supersense

information (in this case combining WordNet and VerbNet resources), our data will look as follows:

- *Cat*$_{ANIMAL}$ *is sitting*$_{SPATIAL-CONFIGURATION}$ *in the bathroom*$_{LOCATION}$, *miaowing*$_{ANIMAL-SOUNDS}$ *loud. (pet_needs_attention)*
- *Dog*$_{ANIMAL}$ *is lying*$_{SPATIAL-CONFIGURATION}$ *in the living room*$_{LOCATION}$, *howling*$_{ANIMAL-SOUNDS}$ *loud. (pet_needs_attention)*
- *Anthony*$_{PERSON}$ *is dancing*$_{PERFORMANCE}$ *at the disco*$_{LOCATION}$, *singing*$_{PERFORMANCE}$ *loud. (no_action_needed)*

This additional semantic annotation helps the classifier to learn generalizable features, and interpret the sentence 1 as similar to sentence 2, but not sentence 3. We detail on the concept of supersenses in the following section.

4.2 Supersenses

Lexical-semantic information is beneficial in many natural language processing and information retrieval applications. The information we are relying on in this thesis builds upon the semantic labels, which lexicographers historically used with the aim to organize the word senses in lexical-semantic resources into several domains, based on syntactic category and semantic coherence [Fellbaum, 1990]. [Ciaramita and Johnson, 2003] first coin the term **supersenses** for these labels.

In WordNet, these are available for nouns and verbs on a synset level. Each noun synset is assigned one out of 26 (each verb synset one out of 15) broad categories, which include labels such as person, location, event, quantity, etc. These categories (supersenses) and their WordNet definitions are listed in table 4.1.

Rather than defining such categories ourselves, we adopted those used in WordNet. However, our approach is generalizable to other definitions of supersenses, as we show in the following chapter, where we extend our approach to the semantic labels in VerbNet. These provide a finer granularity of verb meanings, convenient for particular classification tasks.

The WordNet set of supersenses has a number of attractive features for the purposes of text classification. It is relatively small and therefore easy to process. At the same time, the supersenses are not too vague – most of them seem natural and easily recognizable.

The supersense information was proven to be beneficial in several natural language processing tasks, such as dependency parsing [Agirre et al., 2011], question answering [Surdeanu

NOUNS	nouns denoting	VERBS	verbs of
ACT	acts or actions	BODY	grooming, dressing and bodily care
ANIMAL	animals	CHANGE	size, temperature change, intensifying, etc.
ARTIFACT	man-made objects		
ATTRIBUTE	attributes of people and objects	COGNITION	thinking, judging, analyzing doubting
BODY	body parts		
COGNITION	cognitive processes and contents	COMMUNICATION	telling, asking, ordering singing
COMMUNICATION	communicative processes and contents	COMPETITION	fighting, athletic activities
EVENT	natural events	CONSUMPTION	eating and drinking
FEELING	feelings and emotions	CONTACT	touching, hitting, tying, digging
FOOD	foods and drinks	CREATION	sewing, baking, painting, performing
GROUP	groupings of people or objects		
LOCATION	spatial position	EMOTION	feeling
MOTIVE	goals	MOTION	walking, flying, swimming
OBJECT	natural objects (not man-made)	PERCEPTION	seeing, hearing, feeling
PERSON	people	POSSESSION	buying, selling, owning
PHENOMENON	natural phenomena	SOCIAL	political and social activities and events
PLANT	plants		
POSSESSION	possession and transfer of possession	STATIVE	being, having, spatial relations
		WEATHER	raining, snowing, thawing, thundering
PROCESS	natural processes		
QUANTITY	quantities and units of measure		
RELATION	relations between people or things or ideas		
SHAPE	two and three dimensional shapes		
STATE	stable states of affairs		
SUBSTANCE	substances		
TIME	time and temporal relations		
TOPS	unique beginner for nouns		

Tab. 4.1: Definitions of WordNet supersenses (in WordNet called lexicographer files) for verbs and nouns

et al., 2011] and semantic role labeling [Laparra and Rigau, 2013]. In this thesis, we propose (and demonstrate) the utility of supersenses for document classification tasks, namely personality profiling, subjectivity and sentiment prediction, and metaphor detection.

4.2.1 Annotating supersenses

To explore the impact of supersense features on text classification, we first need to annotate the words in a document with their supersenses. This is possible through two strategies. The first strategy is to apply one of the word sense disambiguation algorithms, e.g. those introduced in the previous chapter, and then access the supersense information in the lexical-semantic resource, such as WordNet, by providing the assigned sense ID and querying the corresponding supersense. This approach has an advantage of being readily usable with existing WSD frameworks, such as our setup in the previous chapter, with only a minimal adaptation (the final mapping of a given sense to its supersense). On the other hand, by forcing the WSD algorithm to identify the fine-grained sense first, and in the next step abstracting from it again, we may introduce unnecessary errors. For example, the word *library* has five WordNet senses, but only two supersenses - an ARTIFACT (a building, a room or a piece of furniture) and a GROUP (a collection of documents, or of programs).

It is therefore convenient to use the strategy of applying a supersense annotation model directly rather than performing the intermediary WSD step.

The supersense tagging (i.e., supersense annotating) task was introduced by [Ciaramita and Johnson, 2003] for nouns and later expanded for verbs [Ciaramita and Altun, 2006]. They trained and evaluated a supervised system on the SemCor data [Miller et al., 1994], a manually sense-annotated corpus of news and other genres, with an F-score of 77.18%, using a hidden Markov model. Their system still holds a state-of-the-art performance in supersense tagging, as evaluated on this corpus.

Direct supersense taggers have then been built also for Italian [Picca et al., 2008], Chinese [Qiu et al., 2011] and Arabic [Schneider et al., 2013], after annotating the corpora with supersenses mostly manually.

Recently, [Johannsen et al., 2014] introduced a task of multiword supersense tagging on Twitter. On their newly constructed dataset, they show poor domain adaptation performance of previous systems, achieving a maximum performance with a search-based structured prediction model [Daumé III et al., 2009] trained on both Twitter and SemCor data. In parallel, [Schneider and Smith, 2015] expanded a multiword expression (MWE) annotated corpus of online reviews with supersense information, following an alternative fine-grained annotation scheme (focused on MWE). Similarly to [Johannsen et al., 2014], they find that SemCor may not be a sufficient resource for supersense tagging adaption to different domains.

In this thesis, we develop our own supersense tagging model using a neural network approach (multilayer, multichannel perceptron) and compare its performance to previous works. The features in our model make use of our concept of supersense embeddings, which we developed as a novel contribution to the emerging word embedding vectors.

4.3 Supersense embeddings

4.3.1 Word embeddings

Recently, word vector representations learned with neural-network based language models have contributed to state-of-the-art results in various linguistic tasks [Bordes et al., 2011, Mikolov et al., 2013c, Pennington et al., 2014, Levy et al., 2015].

One of the most widely used word embedding models is the word2vec [Mikolov et al., 2013c]. Word2vec can utilize either of two model architectures to produce a distributed representation of words: continuous bag-of-words (CBOW) or continuous skip-gram. In the

CBOW architecture, the model predicts the current word from a window of surrounding context words. The order of context words does not influence the prediction, similarly as in a bag-of-words approach for a traditional classifier. In the continuous skip-gram architecture, the model uses the current word to predict the surrounding window of context words. The skip-gram architecture weights nearby context words more heavily than more distant context words.

There are several parameters to be defined for building the word embeddings with word2vec. These include the size of the desired output vectors (usually between 100-300) and the size of the context window used in training. Larger context windows capture a broader semantic relatedness (e.g., *dog* will be close to *bark*) while smaller context windows lead to more syntactically similar vectors (e.g., *dog* will be close to *poodle* and *pitbull*, and *bark* to *yelp*) [Goldberg, 2016]. Other parameters to adjust are the sub-sampling of frequent words (stopwords), and enabling the negative sampling method, which approaches the maximization problem by minimizing the log-likelihood of sampled negative instances. Negative sampling is usually suitable for smaller vector sizes.

In this thesis, we present a novel approach for incorporating the supersense information into the word embedding space. We further propose a new methodology for utilizing these embeddings to label the text with supersenses (Section 4.4) and to exploit the supersenses (Chapter 5) and supersense embeddings (Chapter 6) for multiple different text classification tasks - classifying personality and gender of a text author, predicting sentiment and subjectivity of a text, and distinguishing metaphoric from literal expressions.

4.3.2 Semantically enhanced word embeddings

An idea of combining the distributional information with the expert knowledge is attractive and has been newly pursued in multiple directions. One of them is creating the word sense or synset embeddings [Iacobacci et al., 2015, Chen et al., 2014, Rothe and Schütze, 2015, Bovi et al., 2015]. While the authors demonstrate the utility of these embeddings in tasks such as WSD, knowledge base unification or measuring semantic similarity, the contribution of such vectors to downstream document classification problems can be challenging [Navigli, 2009, Ciaramita and Altun, 2006] due to the fine granularity of the WordNet senses. As discussed above, supersenses have been shown to be better suited for carrying the relevant amount of semantic information. An alternative approach focuses on altering the objective of the learning mechanism to capture relational and similarity information from knowledge bases [Bordes et al., 2011, Bordes et al., 2012, Yu and Dredze, 2014, Bian et al., 2014, Faruqui and Dyer, 2014, Goikoetxea et al., 2015]. While, in principle, supersenses could be seen as a relation between a word and its hypernym, to our knowledge they have not been explicitly employed in these works. Moreover, an important advantage of our explicit supersense embeddings compared to the retrained word

embeddings with altered distances is the direct interpretability of the supersense positions, for example, examining which vectors are the closest to the supersense NOUN.SHAPE and considering enriching the knowledge resource with those, if they are not yet in it.

While supersenses have not been, to our knowledge, used in an embedding fashion, they were recently shown to be beneficial in interpreting (evaluating) word embedding vectors. Specifically, [Tsvetkov et al., 2015] proposed the usage of SemCor [Miller et al., 1994] supersense frequencies as a way to evaluate word embedding models, aligning the word embedding dimensions to a matrix of word-supersense frequencies for several hundreds of common words and assessing how the distances in the embedding vector space align with the grouping of words by their supersense distribution in SemCor. They show that their evaluation score correlates with the performance of the embeddings in word similarity and text classification tasks.

4.3.3 Supersense embeddings

We propose an approach for incorporating explicit lexical-semantic knowledge into the word embedding space. We are the first to provide a joint word- and supersense-embedding model in the same vector space, publicly available at our website[4] for the research community. This joint model provides an insight into the word and supersense positions in the vector space through similarity queries and visualizations, and can be readily used in any word embedding application, as we demonstrate in Chapter 6.

To learn our embeddings, we adapt a freely available sample of 500k articles of Babelfied English Wikipedia [Scozzafava et al., 2015]. To our knowledge, this is one of the largest published and evaluated sense-annotated corpora, containing over 500 million words, of which over 100 million are annotated with BabelNet synsets, with an estimated synset annotation accuracy of 77.8%. Few other automatically sense-annotated Wikipedia corpora are available [Atserias et al., 2008, Reese et al., 2010]. However, as Atserias et al. state (p.2316): *"Wikipedia text differs significantly from the corpora used to train the taggers ... Therefore the quality of these NLP processors is considerably lower."*

We map the BabelNet synsets to WordNet 3.0 synsets [Miller, 1995] using the BabelNet API [Navigli and Ponzetto, 2012], and map these synsets to their corresponding WordNet's supersense categories. For the nested named entities, only the largest BabelNet span is considered, hence there are no nested supersense labels in our data. In this manner we obtain an alternative Wikipedia corpus, where each word is **replaced** by its corresponding supersense (see Table 4.2, second row) and another alternative corpus where each word has its supersense **appended** (Table 4.2, third row). Using the Gensim [Řehůřek and Sojka, 2010] implementation of Word2vec [Mikolov et al., 2013b], we applied the skip-

[4] `https://github.com/UKPLab/acl2016-supersense-embeddings`

Plain Wikipedia	Generalized Wikipedia	Disambiguated Wikipedia
About 10.9% of families were below the poverty line, including 13.6% of those under age 18.	*About 10.9% of* N.GROUP *were below the* N.POSSESSION V.CHANGE *13.6% of those under* N.ATTRIBUTE *18.*	*About 10.9% of* FAMILIES_N.GROUP *were below the* POVERTY_LINE_N .POSSESSION INCLUDING_V.CHANGE *13.6% of those under* AGE_N.ATTRIBUTE *18.*

Tab. 4.2: Example of plain (1), generalized (2) and disambiguated (3) Wikipedia

gram model with negative sampling on these three Wikipedia corpora jointly (i.e., on the columns *Plain Wikipedia*, *Generalized Wikipedia* and *Disambiguated Wikipedia* in Table 4.2) to produce continuous representations of words, supersense-disambiguated words and standalone supersenses in one vector space based on the distributional information obtained from the data. The embeddings are learned using skip-gram as the training algorithm with downsampling of 0.001 higher-frequency words, negative sampling of 5 noise words (i.e., randomly sampled contexts unrelated to the target word), minimal word frequency of 100, window of size 2 and alpha of 0.025, using 10 epochs to produce 300-dimensional vectors. Our experiments with fewer embedding dimensions and with the CBOW model performed worse in the initial embedding quality assessment (tasks described in Section 4.3.5).

The benefits of learning this information jointly are threefold:

1. Vectorial representations of the original words are altered (compared to training on text only), taking into account the similarity to supersenses in the vector space.

2. Standalone supersenses are positioned in the vector space, enabling insightful similarity queries between words and supersenses, esp. for words without a previously known supersense.

3. Disambiguated word+supersense vectors of annotated words can be employed similarly to sense embeddings [Iacobacci et al., 2015, Chen et al., 2014] to improve downstream tasks and serve as the input for supersense disambiguation or contextual similarity systems.

In the following, the designation *Word Embeddings* denotes the experiments with the word embeddings learned on plain Wikipedia text (as in the first column of Table 4.2) while the designation *Supersense Embeddings* denotes the experiments with the word and supersense embeddings learned jointly on the *Plain Wikipedia*, *Generalized Wikipedia* and *Disambiguated Wikipedia*. (i.e., columns 1, 2 and 3 in Table 4.2 together).

4.3.4 Qualitative analysis

Verb supersenses Table 4.3 shows the most similar word vectors to each of the verb supersense vectors using cosine similarity. Note that while no explicit part-of-speech information is specified, the most similar words hold both the semantic and syntactic information - most of the assigned words are verbs. Furthermore, using a large corpus such as Wikipedia conveniently reduces the current need of lemmatization for supersense tagging, as the words are sufficiently represented in all their forms. The most frequent error originates from assigning the adverbs to their related verb categories, e.g. *jokingly* to COMMUNICATION and *drastically* to CHANGE. Figure 4.2 displays the verb supersenses using the t-distributed Stochastic Neighbor Embedding [Van der Maaten and Hinton, 2008], a technique designed to visualize structures in high-dimensional data. While many of the distances are probable to be dataset-agnostic, such as the proximity of BODY, CONSUMPTION and EMOTION, other appear emphasized by the nature of the Wikipedia corpus, e.g. the proximity of supersenses COMMUNICATION and CREATION or SOCIAL and MOTION, as can be explained by table 4.3 (see *led, followed*).

Noun supersenses Table 4.4 displays the most similar word embeddings for noun supersenses. In accordance with previous work on suppersense tagging [Ciaramita and Altun, 2006, Schneider et al., 2012, Johannsen et al., 2014], the assignments of more specific supersenses such as FOOD, PLANT, TIME or PERSON are in general more plausible than those for abstract concepts such as ACT, ARTIFACT or COGNITION.

VERBS	
BODY	wearing, injured, worn, wear, wounded, bitten, soaked, healed, cuffed, dressed
CHANGE	changed, started, added, dramatically, expanded, drastically, begun, altered, shifted transformed
COGNITION	known, thought, consider, regarded, remembered, attributed, considers, accepted, believed, read
COMMUNICATION	stated, said, argued, jokingly, called, noted, suggested, described, claimed, referred
COMPETITION	won, played, lost, beat, scored, defeated, win, competed, winning, playing
CONSUMPTION	feed, fed, employed, based, hosted, feeds, utilized, applied, provided, consumed
CONTACT	thrown, set, carried, opened, laid, pulled, placed, cut, dragged, broken
CREATION	produced, written, created, designed, developed, directed, built, published, penned, constructed
EMOTION	want, felt, loved, wanted, delighted, disappointed, feel, like, saddened, thrilled
MOTION	brought, led, headed, returned, followed, left, turned, sent, travelled, entered
PERCEPTION	seen, shown, revealed, appeared, appears, shows, noticed, see, showing, presented
POSSESSION	received, obtained, awarded, acquired, provided, donated, gained, bought, found, sold
SOCIAL	appointed, established, elected, joined, assisted, led, succeeded, encouraged, initiated, organized
STATIVE	included, held, includes, featured, served, represented, referred, holds, continued, related
WEATHER	glow, emitted, ignited, flare, emitting, smoke, fumes, sunlight, lit, darkened

Tab. 4.3: Top 10 word embeddings with the highest cosine similarity to each of the verb supersense vectors

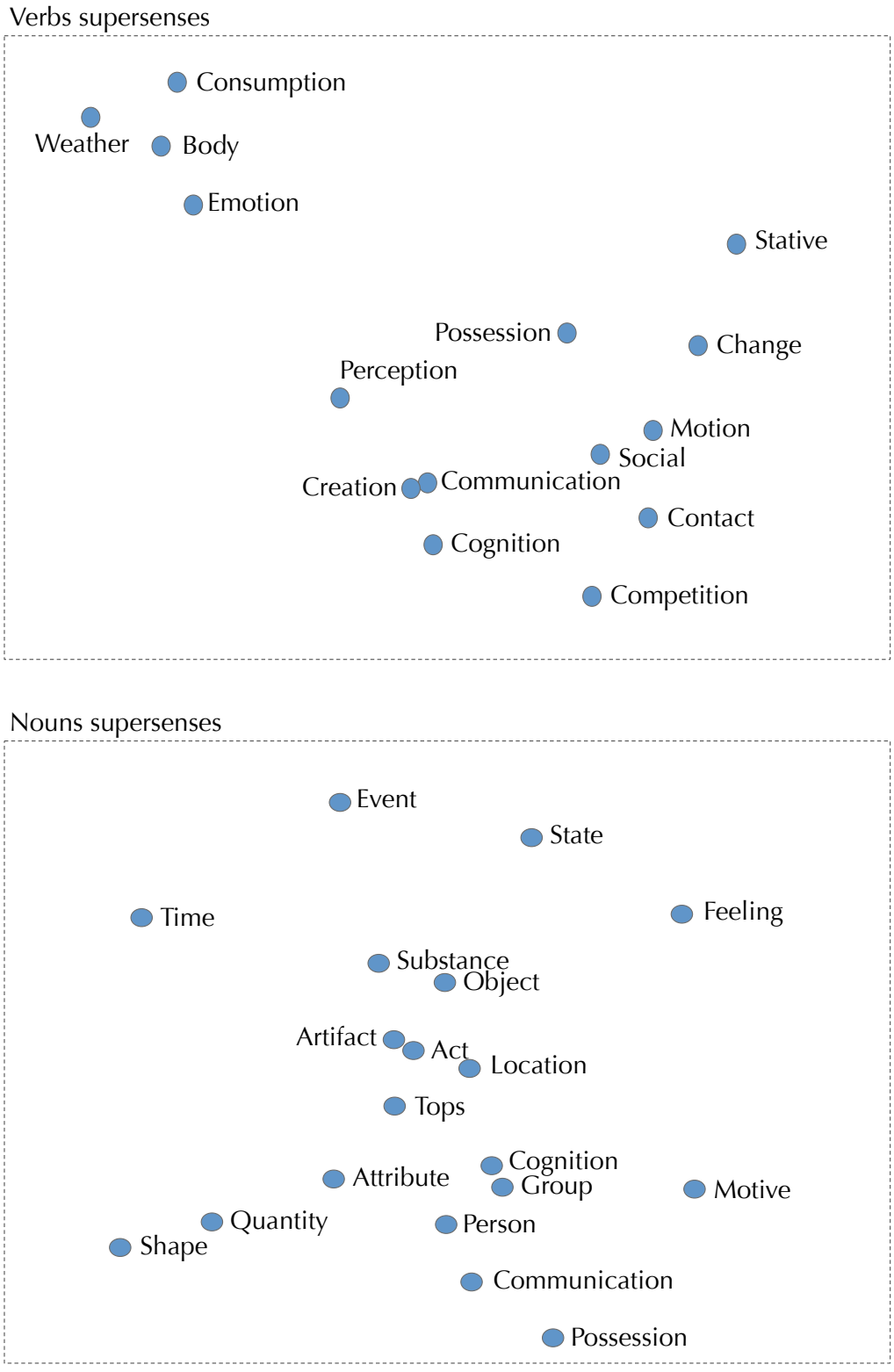

Fig. 4.2: Projection of 300-dimensional verb and noun supersense embeddings into a 2D space using the t-SNE visualization method [Van der Maaten and Hinton, 2008], which preserves semantically meaningful distances between concepts. We can see that the abstract concepts are more central, with many close neighbors, while the concrete concepts are more distinct.

The same is visible in Figure 4.2, where these supersense embeddings are more central, with closer neighbors. In contrast to the observations by [Schneider et al., 2012] and [Johannsen et al., 2014], the COMMUNICATION supersense appears well defined, likely due to the character of the Wikipedia corpus.

4.3.5 Word analogy and word similarity tasks

With the aim to assess the difference between the individual word embeddings learned on plain Wikipedia text (referred to as **Word embeddings**) and jointly with the supersense-enriched Wikipedia (referred to as **Supersense embeddings**), we perform two standard embedding evaluation tasks: word similarity and word analogy.

NOUNS	
ACT	participation, activities, involvement, undertaken, ongoing, conduct, efforts, large-scale, success
ANIMAL	peccaries, capybaras, frogs, echidnas, birds, marmosets, rabits, hatchling, ciconiidae, species
ARTIFACT	wooden, two-floor, purpose-built, installed, wall, fittings, turntable, racks, wrought-iron, ceramic, stone
ATTRIBUTE	height, strength, age, versatility, hardness, power, fluidity, mastery, brilliance, inherent
BODY	abdomen, bone, femur, anterior, forearm, femoral, skin, neck, muscles, thigh
COGNITION	ideas, concepts, empirical, philosophy, knowledge, epistemology, analysis, atomistic, principles
COMMUNICATION	written, excerpts, text, music, excerpted, translation, lyrics, subtitle, transcription, words
EVENT	sudden, death, occurred, event, catastrophic, unexpected, accident, victory, final, race
FEELING	sadness, love, sorrow, frustration, disgust, anger, affection, feelings, grief, fear
FOOD	cheese, butter, coffee, milk, yogurt, dessert, meat, bread, vegetables, sauce
GROUP	members, school, phtheochroa, ypsolophidae, pitcairnia, cryptanthus, group, division, schools
LOCATION	northern, southern, northeastern, area, south, capital, town, west, region, city
MOTIVE	motivation, reasons, rationale, justification, motive, justifications, motives, incentive, desire, why
OBJECT	river, valley, lake, hills, floodplain, lakes, rivers, mountain, estuary, ocean
PERSON	greatgrandfather, son, nephew, son-in-law, father, halfbrother, brother, who, mentor, fellow
PHENOMENON	wind, forces, self-focusing, radiation, ionizing, result, intensity, gravitational, dissipation, energy
PLANT	fruit, fruits, magnifera, sativum, flowers, caesalpinia, shrubs, trifoliate, vines, berries
POSSESSION	property, payment, money, payments, taxes, tax, cash, fund, pay, $100
PROCESS	growth, decomposition, oxidative, mechanism,rapid, reaction, hydrolysis, inhibition, development
QUANTITY	miles, square, meters, kilometer, cubic, ton, number, megabits, volume, kilowatthours
RELATION	southeast, southwest, northeast, northwest, east, portion, link, correlation, south, west
SHAPE	semicircles, right-angled, concave, parabola, ellipse, angle, circumcircle, semicircle, lines
STATE	chronic, condition, debilitating, problems, health, worsening, illness, illnesses, exacerbation, disease
SUBSTANCE	magnesium, zinc, silica, manganese, sulfur, oxide, sulphate, phosphate, salts, phosphorus
TIME	september, december, november, july, april, january, august, february, year, days
TOPS	time, group, event, person, groups, individuals, events, animals, individual, plant

Tab. 4.4: Top 10 word embeddings with the highest cosine similarity to each of the noun supersense vectors

[Mikolov et al., 2013c] introduce a word analogy dataset containing 19544 analogy questions that can be answered with word vector operations (*Paris is to France as Athens is to...?*). The questions are grouped into 14 categories such as State - Capital pairs or antonymy pairs. Table 4.5 presents our results.[5] Using McNemar's test [McNemar, 1947], we observe no statistically significant difference ($p < 0.05$) in performance on the individual analogy groups with the exception of the group *Capitals - common*, where the NOUN.LOCATION supersense likely contributed to the improvement, and the syntactic analogy group *Plurals*, where the performance with supersenses drops. In general, word embeddings trained jointly with supersenses (column **Supersense embeddings** in the table) achieve better results than **Word embeddings** on the analogy groups related to entities, e.g. *Family Relations* and *Citizen to State* questions, where the PERSON and LOCATION supersenses can provide additional information to reduce noise. At the same time, performance on questions such as *Opposites, Plurals* or *Gerund to past* drops. We hypothesize that this information is pushed to the background by nouns with the same supersense being closer together. Enriching our data with the recently proposed adjective supersenses [Tsvetkov et al., 2014] could be of interest for these categories.

Analogy group	Example	Pairs	**Word embeddings**	**Supersense embeddings**
Capitals - common	Athens-Greece	506	91.1	94.7
Capitals - world	Abuja-Nigeria	4,524	87.6	89.5
City in state	Chicago-Illinois	2,467	65.2	65.7
Nationality to state	Albania-Albanian	980	94.5	95.2
Family relations	uncle-aunt	506	93.0	94.4
Opposites	happy-unhappy	812	56.7	54.6
Plurals	banana-bananas	1,332	89.4	86.4
Comparatives	bad-worse	1,332	90.6	90.4
Superlatives	bad-worst	1,332	79.4	79.6
Adjective to adverb	amazing-amazingly	992	20.2	22.2
Present to gerund	dance-dancing	1,056	64.2	64.6
Gerund to past	coding-coded	1,560	60.0	59.2
3rd person verbs	decrease-decreases	870	84.3	82.1
Total	18,888		**75.0**	**76.0**

Tab. 4.5: Accuracy and standard error on analogy tasks from Mikolov [2013]. Tasks related to noun supersense distinctions show the tendency to improve, while syntax-related information is pushed to the background. Statistically significant differences (McNemar's two-tailed test, $p < 0.05$) are underlined.

Without explicitly exploiting the sense infromation, we compare the performance of our text-only-trained to our jointly trained word embeddings on the following word similarity datasets: WordSim353-Similarity (353-S) and WordSim353-Relatedness (353-R) [Agirre et al., 2009a], MEN dataset [Bruni et al., 2014], RG-65 dataset [Rubenstein and Goodenough, 1965] and MC-30 [Miller and Charles, 1991].

[5] We omit the category Currency as it would require maintaining special characters in the training of word embeddings, which is not relevant for our task.

Dataset	Bruni 2014 (MEN)	WordSim353 Similarity (353-S)	WordSim353 Relatedness (353-R)	Rubenstein 1965 (RG-65)	Miller and Charles (MC-30)
Word embeddings	73.18	76.93	62.11	79.13	79.49
Supersense embeddings	74.26	78.63	61.22	79.75	80.94

Tab. 4.6: Performance of our vectors (Spearman's ρ to human judgments) on five similarity datasets. Results indicate a trend of better performance of embeddings trained jointly with supersenses than the original word embeddings, although not statistically significant ($p > 0.05$)[Rastogi et al., 2015].

The word embeddings for words trained jointly with supersenses achieve higher performance than those trained solely on the same text without supersenses on 4 out of 5 tasks (Table 4.6). The differences, however, are small, and following the thresholds reported in previous work [Rastogi et al., 2015, Batchkarov et al., 2016], not statistically significant at $p > 0.05$. To optimize for these tasks, the explicit supersense information could be further exploited in the spirit of previous sense embedding works [Iacobacci et al., 2015, Rothe and Schütze, 2015, Chen et al., 2014]. However, we leave it to future work, as our focus is on downstream applications.

Note that while we report the performance of our embeddings on the word similarity tasks for historical reasons (comparability to previous publications evaluating word embeddings), there has been a substantial discussion on seeking alternative ways of word embedding evaluation with the focus on their purpose in downstream applications [Li and Jurafsky, 2015, Faruqui et al., 2016]. Therefore, in Chapter 6, we evaluate the usefulness of supersense embeddings in text classification tasks rather than by their intrinsic vector space properties.

4.4 Supersense tagging experiments

The task of predicting supersenses has recently regained its popularity [Johannsen et al., 2014, Schneider and Smith, 2015], since supersenses provide disambiguating information, useful for numerous downstream NLP tasks, without the need of tedious fine-grained WSD. Exploiting our joint word and supersense embeddings, we build a deep neural network model to predict supersenses on the Twitter supersense corpus created by [Johannsen et al., 2014],using the same training data as the authors. [6][7] The datasets follow the token-level annotation which combines the B-I-O flags [Ramshaw and Marcus, 1995] with the supersense class labels to represent the multiword expression segmentation and supersense labeling in a sentence. For example, the expression *tomorrow morning* is labeled as: *tomorrow* `B-noun.time`, *morning* `I-noun.time`.

[6] `https://github.com/kutschkem/SmithHeilmann_fork/tree/master/MIRATagger/data`
[7] `https://github.com/coastalcph/supersense-data-twitter`

4.4.1 Experimental setup

We implement a sliding window approach with a multi-layer perceptron model in a multi-channel architecture using the Theano framework [Bastien et al., 2012]. Theano offers flexibility in model architecture and operates on multidimensional arrays that can be easily parallelized, allowing more efficient utilization of GPUs. We use a sliding window of size 5 for the sequence learning setup, and extract for each word the following seven "channels" (feature vectors) which become an input of the network:

1. 300-dimensional word embedding,
2. 41 cosine similarities of the target word embedding to each standalone supersense embedding (Generalized Wikipedia),
3. 41 cosine similarities of the target word embedding to each of its *word*_SUPERSENSE embeddings (Disambiguated Wikipedia),
4. fixed vector of frequencies of each supersense in Generalized Wikipedia, in order to simulate the MFS backoff strategy,
5. for the given word, the frequency of each *word*_SUPERSENSE in our Disambiguated Wikipedia,
6. part-of-speech information for the target word, supplied as a unit vector,
7. casing information for the target word as a 3-dimensional (upper/lower/mixed) unit vector

After a Dropout regularization, the embedding sets are flattened, concatenated and fed into fully connected dense layers with a rectified linear unit (ReLU) activation function and a final softmax. For a more detailed understanding of the above-mentioned settings, we refer the interested reader to the neural network Background section in Chapter 6.

4.4.2 Supersense prediction

We evaluate our system on the same Twitter dataset with provided training and development (`Twitter-Ritter Development`) set and two test sets: `Twitter-Ritter Evaluation`, reported by Johannsen et al. as *RITTER*, and `Twiter-Johannsen Evaluation`, reported by Johannsen et al. as *IN-HOUSE*. Our results are shown in Table 4.7 and compared to the results reported in previous work by [Johannsen et al., 2014], with two additional

System/Data:	Twitter -Ritter Development	Twitter -Ritter Evaluation	Twitter -Johannsen Evaluation
Baseline and upper bound			
Most frequent sense	47.54	44.98	38.65
Inter-annotator agreement	-	69.15	61.15
SemCor-trained systems			
[Ciaramita and Altun, 2006]†	48.96	45.03	39.65
Searn [Johannsen et al., 2014]	56.59	50.89	40.50
HMM [Johannsen et al., 2014]	57.14	50.98	41.84
Ours Semcor	54.47	50.30	35.61
Twitter-trained systems			
Searn [Johannsen et al., 2014]	67.72	**57.14**	42.42
HMM [Johannsen et al., 2014]	60.66	51.40	41.60
Ours Twitter (all features)	61.12	**57.16**	41.97
Ours Twitter no casing	61.06	56.20	41.13
Ours Twitter no similarities	63.47	56.78	39.44
Ours Twitter no frequencies	61.10	**57.32**	39.02
Ours Twitter no part-of-speech	57.08	54.45	36.50
Ours Twitter no word embed.	57.57	53.43	34.91

Tab. 4.7: Weighted F-score performance on supersense prediction for the development set and two test sets provided by Johannsen et al. [2004]. Our configurations perform comparably to state-of-the-art system, highlighted in bold.
† For the system of Ciaramita et al, the publicly avaliable reimplementation of Heilman was used – `https://github.com/kutschkem/SmithHeilmann_fork/tree/master/`

baselines: The SemCor system of [Ciaramita and Altun, 2006] and the most frequent sense. Remarkably, our system achieves comparable performance to the best previously used supervised systems, without using any explicit gazetteers.

Feature ablation and error analysis. To get an intuition[8] of how the individual feature vectors contribute to the prediction, we perform an ablation test by removing one feature group at a time. The biggest performance drop in the F-score occurs when removing the part of speech information (second-to-last row in Table 4.7). The casing information, typically important in Named Entity Recognition tasks, has a minimal contribution to Twitter supersense tagging. The largest part of the errors comes from omitting to label a supersense (labeling it as `O-other`). In accordance with previous work [Ciaramita and Altun, 2006], the easiest class to label is `noun.person`, while more abstract supersenses such as `noun.act` are more error-prone.

[8] Intuition, since there are many additional aspects that may affect the performance. For example, we keep the network parameters fixed for the ablation, although the feature vectors are of different lengths. Furthermore, our model performs a concatenation of the feature vectors, hence only the ablation extended to all possible permutations would verify the feature order effect, which is however beyond the scope of these experiments.

4.5 Constructing supersense embeddings on other corpora

In this section, we extend our supersense embedding construction work beyond Wikipedia and explore the effect that the choice of an underlying corpus has on the properties of the resulting embeddings. We cannot perform the McNemar's test to evaluate the significance of performance difference to the state of the art, as we do not have the item-level results of the previous systems available.

4.5.1 Corpora used

Wikipedia: We compare the three other corpora to the Babelfied Wikipedia [Scozzafava et al., 2015] supersense embeddings which we constructed in the previous section.

SemCor: To our knowledge, the largest manually sense-disambiguated dataset is the SemCor corpus [Miller et al., 1994]. SemCor is a subset of the English Brown Corpus containing 360,000 words from 15 genres, including press, religion and fiction, with more than 200,000 sense annotations. We link the assigned WordNet synsets to their lexicographer files, similarly to the previous work of [Heilman, 2011] and [Ciaramita and Altun, 2006].

Twitter: [Johannsen et al., 2014] published a small corpus of Twitter posts (ca. 20,000 words) annotated for supersenses directly. This approach overcomes some WordNet limitations - e.g. only 40% of the nouns and verbs annotated in the evaluation set of [Johannsen et al., 2014] are covered by WordNet.

STREUSLE: Recently, [Schneider and Smith, 2015] introduced the STREUSLE corpus of online reviews with 55,000 words - a subset of the English Web Treebank [Bies et al., 2012], annotated for multiword expressions and supersenses.

All four systems use the same WordNet supersense inventory, howeverm the annotation instructions differ for the case of STREUSLE [Schneider and Smith, 2015] and Twitter [Johannsen et al., 2014]. STREUSLE authors suggest a hierarchical preference of annotations from more concrete to more abstract supersenses, and in the Twitter case, verbs are typically assigned a different supersense than in WordNet (e.g. *tweet* or *follow*), even when using the same inventory.

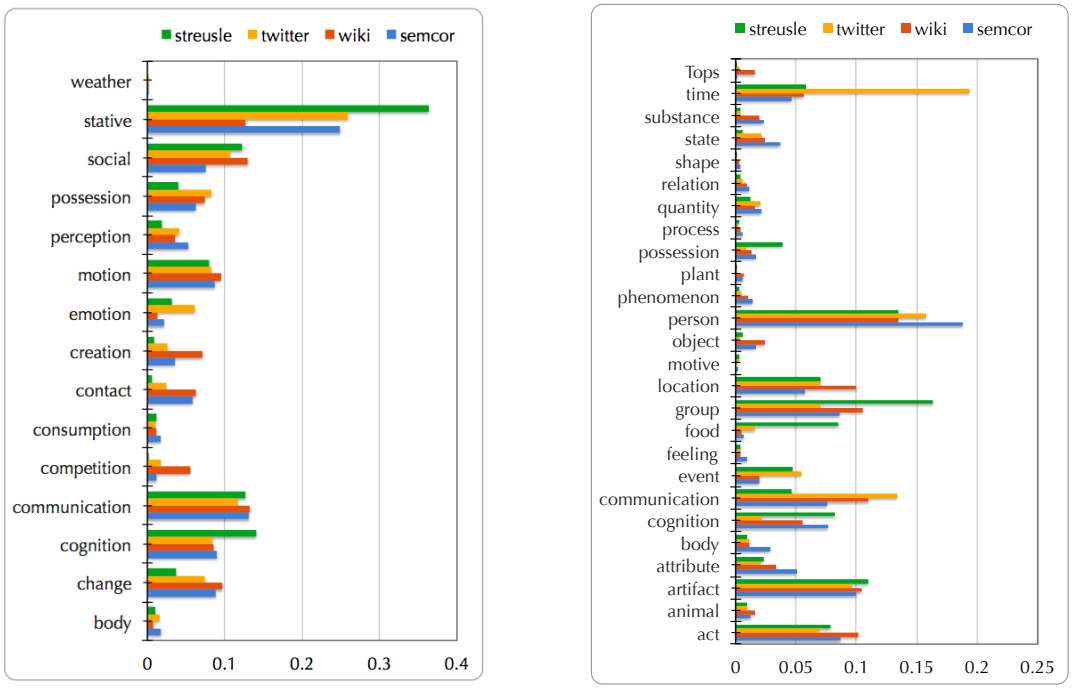

Fig. 4.3: Relative occurrences of the verb (left) and noun (right) supersenses in the four examined corpora (i.e., the occurrence counts for all supersenses within one corpus are normalized to 1).

4.5.2 Differences between corpora

Before proceeding to the supersense embeddings, we are interested in the properties of annotated corpora and differences between them with respect to the supersense annotations. Figure 4.3 displays the proportion of verb and noun supersenses for each corpus.

We observe that Twitter, in comparison to other corpora, has a higher proportion of supersenses NOUN.TIME and NOUN.COMMUNICATION. This originates from the purpose of this medium - people often share current events *(today)* or plans *(tomorrow)*. NOUN.COMMUNICATION category is frequent due to the names of social networks and mentions of other messages. It is also the most personal corpus, with a frequent VERB.EMOTION supersense.

Wikipedia contains proportionally more supersenses NOUN.ANIMAL, NOUN.PLANT, NOUN.OBJECT, NOUN.LOCATION, NOUN.ACT and NOUN.TOPS, which corresponds to its purpose as well. The verb supersenses VERB.COMPETITION and VERB.CREATION are more frequent, as Wikipedia entries are often triggered by inventions or achievements.

The STREUSLE review dataset shows the highest proportion of the supersenses NOUN.FOOD, NOUN.ARTIFACT, NOUN.GROUP (e.g. *store, restaurant*) and NOUN.COGNITION (e.g. *problem*), which unmistakeably represents its content type. High frequency of VERB.COGNITION and

VERB.STATIVE suggests that users describe the properties of the subject of the review and their impressions.

SemCor has proportionally the highest representation of NOUN.PERSON, VERB.BODY (e.g. *smile*), VERB.CONTACT (e.g. *kill*) and VERB.PERCEPTION (e.g. *see, watch*), originating mainly from the fiction genre that it contains. Ratio of NOUN.COGNITION supersenses (e.g. *belief, feeling*) is also high, mostly due to the religion genre.

Supersense domain differences Table 4.8 displays the verbs with the largest differences across corpora in terms of their majority supersense in each corpus. These differences are mostly caused by the purpose of the corpus. For example the prevalent sense of the verb *discover* in Wikipedia is VERB.PERCEPTION, i.e., to determine the existence, while in other corpora it is VERB.COGNITION, i.e. to learn about a fact. Similarly, the verb *follow* appears as a VERB.COMMUNICATION in Twitter and Streusle (following a user), as a VERB.MOTION in Wikipedia, and as a VERB.STATIVE (e.g. *an example follows*) in SemCor. The verb *spend* is used as VERB.POSSESSION in the reviews and as VERB.STATIVE (as in *spend some time*) otherwise.

For nouns, the situation is similar, with e.g. *star* being in 95% of the cases a NOUN.OBJECT in Wikipedia and mostly NOUN.PERSON elsewhere.

Verb / Supersense:	Wikipedia	SemCor	Twitter	Streusle
born	stative	change	stative	stative
caught	cognition	motion	social	social
charged	competition	communication	competition	possession
confirmed	social	cognition	communication	communication
count	communication	communication	communication	cognition
discovered	perception	cognition	cognition	cognition
follow	motion	stative	communication	communication
lost	possession	possession	possession	cognition
mean	communication	communication	cognition	cognition
meet	motion	social	social	social
need	stative	stative	stative	cognition
note	communication	perception	communication	perception
order	communication	communication	possession	communication
read	cognition	cognition	cognition	communication
reserved	possession	cognition	possession	communication
responded	cogn,	cognition	communication	communication
spent	stative	stative	stative	possession
stop	motion	social	motion	motion
want	emotion	emotion	emotion	cognition
write	creation	creation	creation	communication

Tab. 4.8: Verbs with the largest supersense differences between corpora. Table shows the most frequent supersense annotation for the given verb in each dataset.

4.5.3 New senses of existing words

The direct supersense annotation of the Twitter and STREUSLE corpora enabled to get beyond WordNet supersenses even for words which already exist in WordNet. For example an *account* as a NOUN.ARTIFACT, to *tweet* and to *follow* as a VERB.COMMUNICATION or to *chill* as a VERB.STATIVE. If an automated system used WordNet as a backup annotation solution, problems would arise also with named entities already present in WordNet. This is for example the case of *Amazon* (NOUN.GROUP), *Amber* (NOUN.PERSON) or *Face* (NOUN.COMMUNICATION, an abbreviation for Facebook) in the Twitter data.

4.5.4 Exploring supersenses via embedding properties

Methodology

Since the words and their supersenses are embedded into the same vector space, we can examine the relation of each word to all of its possible supersense labels, which allows us to study the nature of the semantic ambiguities, even for words that were not labeled with supersenses in the original corpus. Furthermore, we can explore similarities between the supersenses themselves.

For each of the corpora, we use two textual inputs - the original, unannotated text, and an altered text in which the annotated nouns and verbs were replaced by their supersenses. We then train a word embedding model on both texts (the plain and the supersense-based ones) jointly. We merge the small Twitter corpus with the STREUSLE corpus to obtain more robust vectors, as both corpora contain user-generated, informal online text. We thus obtain three vector models: 1) Wikipedia, 2) Twitter+STREUSLE, 3) SemCor.

For each embedding model, we compute the cosine similarity of each word vector to each supersense vector, and the similarities between supersenses themselves. The resulting 26-dimensional similarity vectors for each noun (or noun supersense) and 15-dimensional similarity vectors for each verb (or verb supersense) are fed into an online visualization tools which he have built.[9]

Relations between supersenses across corpora

Our tools allow for visualization of similarities between noun, resp. verb supersenses. This enables researchers to investigate the tendencies to regular polysemy [Lakoff and Johnson,

[9] `http://mydemo.czweb.org/`

2008, Copestake and Briscoe, 1995, Boleda et al., 2012], such as the sense alterations between NOUN.ANIMAL and NOUN.FOOD, or to explore ambiguous supersenses.

The overall surface of the graph provides a first notion of distinctiveness of a supersense. For clearly defined concepts, such as NOUN.PLANT, the similarity to other supersenses is in general lower (Figure 4.4) than for more abstract concepts such as NOUN.ACT (Figure 4.5). The supersense NOUN.PLANT also confirms the regular polysemy findings through its similarity to NOUN.FOOD, NOUN.SUBSTANCE and NOUN.OBJECT.[10]

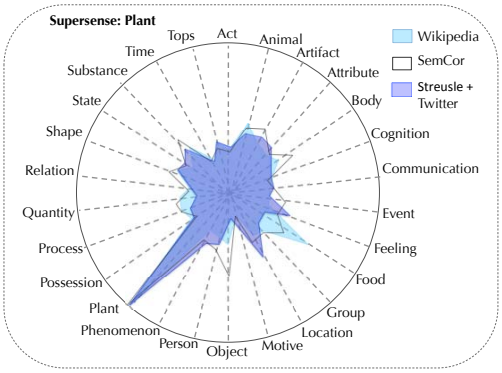

Fig. 4.4: Cosine similarity of the well-defined supersense PLANT to other noun supersenses. The overall surface of the plot is smaller.

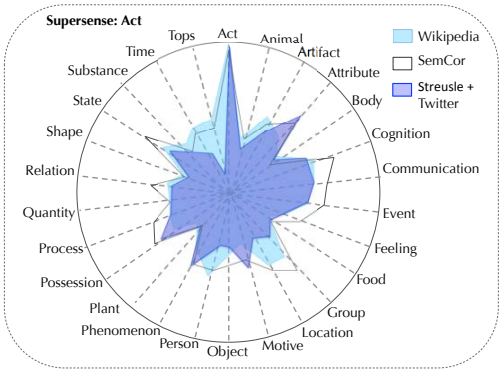

Fig. 4.5: Cosine similarity of the more abstract supersense ACT to other noun supersenses. The overall surface of the plot is larger.

Many supersense relations correspond to the annotation and classification errors fragmentarily described in previous work. E.g., the most similar supersenses for NOUN.COMMUNICATION are NOUN.ACT, NOUN.ARTIFACT and NOUN.PERSON, in accordance with the error analysis of [Schneider et al., 2012] and [Ciaramita and Johnson, 2003].

The discrepancies between corpora are revealed as well. E.g., the NOUN.EVENT supersense is more similar to NOUN.TIME in the Twitter+Streusle corpus, while in SemCor the similarity is higher to NOUN.ACT. These differences can be relevant e.g. for NLP tasks of event extraction.

Regarding verbs, we consistently observe a similarity of VERB.COMMUNICATION to VERB.COGNITION and VERB.SOCIAL, as illustrated on Figure 4.6. Indeed, according to [Ciaramita and Johnson, 2003] "*abstract classes such as communication or cognition are more confusable*". Many findings are intuitive, such as the similarity of VERB.MOTION, VERB.BODY and VERB.CONTACT or the distinctiveness of VERB.WEATHER. Intriguing is the similarity of the supersense VERB.POSSESSION to VERB.SOCIAL across corpora, possibly triggered by the verbs such as *give* and *donate* in the former, and *help* and *offer* in the latter.

[10] `http://mydemo.czweb.org/nounsupersenses.php?noun=nounplant`

4.5.5 Comparison between words, across corpora and to annotations

Another view in our demonstration allows for comparison of one word across different corpora, which can be beneficial for domain adaptation tasks. Figure 4.7 shows the comparison for the verb *to know*. According to WordNet, *know* shall be assigned the VERB.COGNITION supersense, and the figure confirms a high similarity to it in all three corpora. However, in Wikipedia and SemCor, the verb *know* manifests also a high similarity to the VERB.EMOTION supersense. This is not the case for the Streusle+Twitter corpus, in which the supersenses were not deducted from fine-grained WSD, but annotated directly. Indeed, we found that in the annotation guidelines for the STREUSLE corpus, a precedence relation between the more abstract VERB.EMOTION and more concrete VERB.COGNITION was specified.

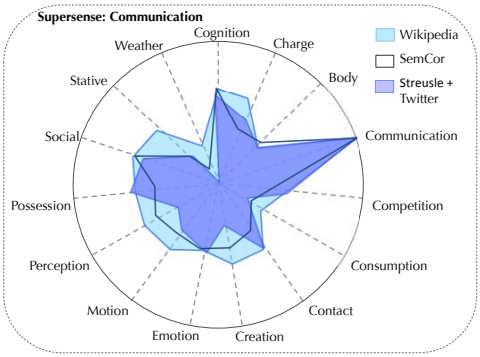

Fig. 4.6: Similarity of the verb supersense COMMUNICATION to WordNet verb supersense embeddings in Wikipedia, SemCor and Streusle+Twitter

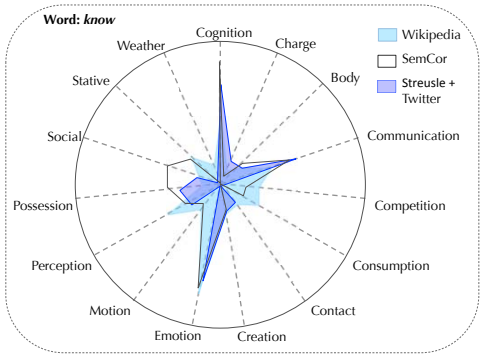

Fig. 4.7: Similarity of the verb *know* to WordNet supersense embeddings in Wikipedia, SemCor and Streusle+Twitter

4.6 Chapter summary

In this chapter, we presented approaches to acquiring abstraction over individual words, thereby **addressing the lexical gap issue**, and propose to focus on the lexical semantic information called supersenses. We developed a competitive method to annotate supersenses for words in a document, and trained and evaluated a **supersense tagging model**. Our solution is available open-source on the group website. [11]

We proposed a novel concept of joint dense vectors of words and supersenses (**supersense embeddings**), built those and evaluated their semantic properties. We investigated the impact of the choice of an underlying corpus for building the supersense embeddings and

[11] `https://github.com/UKPLab/acl2016-supersenses`

illustrated that the semantic properties may differ and shall be considered with regards to the end task. We provided a visual interface for comparing the properties of different supersense embeddings. The major contributions in this chapter have been published in [Flekova and Gurevych, 2016].

In the next chapter, we evaluate the usefulness of supersense annotations for the downstream text classification tasks, and compare the direct supersense tagging method with the option of obtaining the supersenses through fine-grained WordNet sense annotations.

5 Concept Generalization Experiments

> *I do not carry such information in my mind since it is readily available in text books.*
>
> **— Albert Einstein**

In the previous chapter (Chapter 4), we introduced the idea of supersenses as a way to provide concept generalization over individual words. In this chapter, as illustrated in Figure 5.1, we empirically investigate the impact of using supersense annotations as features in the traditional supervised document classification settings, using support vector machines. Roughly, we look up the supersense information in a lexical-semantic resource and supply it to the classifier so that it does not have to learn it implicitly and can rather build upon this information. In the following Chapter 6, we then proceed towards using the dense supersense vectors (supersense embeddings) instead of plain supersense labels in text classification tasks, and evaluating those in deep learning experiments.

We conduct our experiments on four different tasks - (1) extraversion prediction (for real and fictional characters), (2) gender classification, (3) subjectivity classification and (4) sentiment classification. We perform the evaluation mostly on existing datasets (introduced in Chapter 3) to enable comparison to previous work. An exception is the extraversion prediction for fictional characters, where we created our own dataset.

Experimental settings

For the classification we use the same experimental settings as in Chapter 3, i.e., the SVM classifier with the χ^2 feature selection, performing 10-fold cross-validation over our entire data.

For each of the experiments, we compare the following document representations:

- **WORD:** bag-of-words features
- **LIWC:** bag of words and LIWC features (details below)
- **SENSE-SUPER:** bag of words and supersense features annotated through WordNet senses determined with the Simplified Lesk WSD algorithm

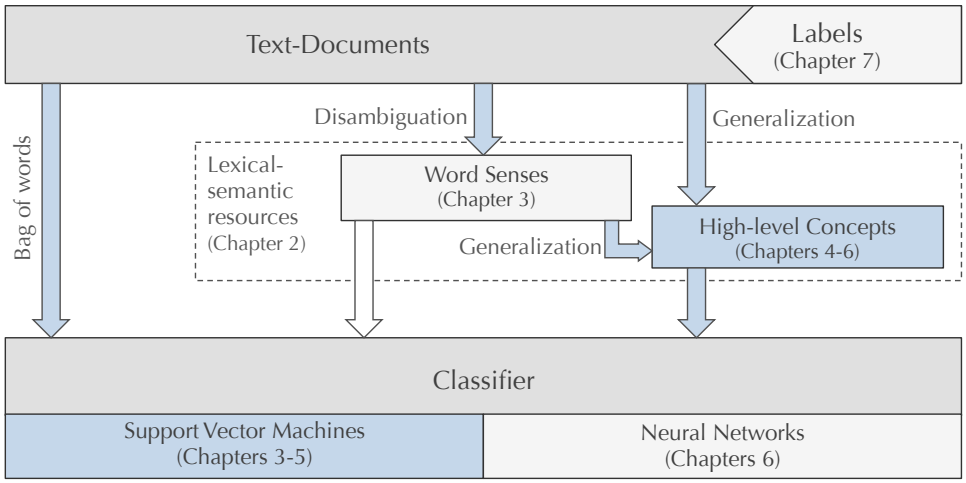

Fig. 5.1: The concepts and workflows of this thesis explored in this chapter (highlighted blue).

- **SUPER:** bag of words and supersense features annotated directly with a supersense tagger
- **ALL:** bag of words, LIWC, supersenses and additional task-specific features (details below) to compare against the performance in other papers
- **SoA:** state-of-the-art results, i.e., best reported performance in other papers known to us

LIWC features The Linguistic Inquiry and Word Count (LIWC) tools [Pennebaker et al., 2001], introduced in the previous chapter, was widely used in personality profiling experiments. It consists of word lists related to psychological processes (cognitive, perceptual, social, biological, affective) and personal concerns (achievement, religion, death...) and other categories such as fillers, disfluencies or swear words[1]. We extract 81 additional features corresponding to these word lists. Note that the LIWC operates on word level rather than sense level, therefore for example an *isolated cable* would be an expression contributing to the emotional category Sadness.

Additional task-specific features Since emotion features have been found predictive in previous personality work [Mohammad and Kiritchenko, 2013], we measure overall positive and negative sentiment expressed using SentiWordNet [Esuli and Sebastiani, 2006] and NRC Emotion Lexicon [Mohammad, 2012]. Additionally, following up on previous work, we capture the syntactic and stylistic properties of each document, disregarding the semantics. Starting from the surface properties, we measure the sentence, utterance and word length, including the proportion of words shorter than 4 or longer than 6 letters, frequency of each punctuation mark, and endings of each adjective as per [Corney et al., 2002]. On the

[1]For a complete overview refer to `www.liwc.net`

syntactic level we measure the frequency of each part of speech as well as the 500 most frequent part-of-speech bi-, tri- and quadrigrams, and the frequency of each dependency obtained from the Stanford Parser. We additionally capture the frequency of superlatives, comparatives and modal verbs, the proportion of verbs in present, past and future tense, and the formality of the language as per the part-of-speech-based formality coefficient [Heylighen and Dewaele, 2002], and measure the average depth of the parse trees.

5.1 Extraversion experiments

Extraversion has been shown to affect many aspects of daily life, for example, job performance [Tett et al., 1991] and inter-personal relationship satisfaction [White et al., 2004], or as [McCrae et al., 2012] put it, "anything from sex [Trobst et al., 2002] to drugs [Terracciano et al., 2008] to rock-and-roll [Rentfrow et al., 2011]." Predicting extraversion automatically can be therefore beneficial for numerous applications, including automated dialog agents, customer satisfaction analysis or computational forensics.

In this section, we first present our experiments on existing datasets of human personality (Subsection 5.2.1). Our goal in this subsection is to analyze if the supersense features contribute to the extraversion prediction on different types of data, therefore we focus on the performance differences rather than discussing the contributions of individual state-of-the-art features and the psychological reasoning behind them. For a detailed discussion on correlations between lexical and stylistic aspects of text and extraversion of the author, we refer the interested reader to numerous previous experiments [Pennebaker and King, 1999, Dewaele and Furnham, 1999, Gill and Oberlander, 2002, Mehl et al., 2006, Aran and Gatica-Perez, 2013, Lepri et al., 2010], including those on the data we use here [Celli et al., 2013, Celli et al., 2014]. As the previous experiments used very similar stylistic features as we do, we mostly confirm those previous observations, and focus on the role of supersenses in these experiments, which they did not explicitly explore.

We however pursue the feature analysis on a deeper level in the Subsection 5.2.2, where we introduce the novel task of personality prediction for fictional characters. Besides demonstrating the usefulness of supersenses, we compare the most predictive features for fictional characters to those found in the task of extraversion prediction of human individuals. Since this experiment was performed before the development of our direct supersense tagging model, we only apply the SENSE-SUPER setup instead of the direct SUPER one. However, in this task we additionally explore the potential of densely linked lexical-semantic resources (here, exploiting the sense-level links between WordNet and VerbNet), to demonstrate that our method generalizes to additional knowledge bases, which can be customized based on the task at hand. This is presented in our results as the SENSE-SUPER-VN setup.

5.1.1 Extraversion of human individuals

The corpora we use in this work were previously utilized in the personality challenges at the Workshop of Computational Personality Recognition (WCPR) 2013 and 2014 [Celli et al., 2013, Celli et al., 2014]. The corpora consistz of the three datasets we described in Chapter 3, namely the stream-of-consciousness personality essays [Pennebaker and King, 1999], the Facebook personality data [Kosinski et al., 2013] and the Youtube personality-annotated video transcripts [Biel and Gatica-Perez, 2013]. We list the results of our experiments in Table 5.1 and compare them to the best reported results from these workshops, which are, to our knowledge, the most recent experiments on these datasets to date.

Previous work

Personality essays: The dataset of personality essays was first used in the supervised classification experiments of [Mairesse et al., 2007]. They experiment with several classifiers, using features based on the word lists of the LIWC lexicon [Pennebaker et al., 2001] and the MRC Psycholinguistic Database [Wilson, 1988]. They achieve a binary classification accuracy around 54% and confirm the feature correlations previously reported by [Pennebaker and King, 1999], i.e., extraverts talk significantly more about their family and friends, use more pronouns and positive emotions, and refer more often to themselves. Introverts use more hedges, negations, longer words and articles, and talk more about music. The classification setup of [Mairesse et al., 2007] was extended by [Mohammad and Kiritchenko, 2013], who added fine-grained emotion features and further improved the performance, using SVM classifiers. They extract the fine-grained emotions from the NRC Hashtag Emotion Lexicon [Mohammad et al., 2013b], which contains around 10,000 words with associations to 585 emotion-word hashtags. Emotions such as *happy, admiring* and *jealous* are listed among the most predictive. We use their results as a state-of-the-art (SoA) benchmark for this dataset in our results table 5.1. Other researchers performed experiments on this dataset as a part of the 2013 WCPR workshop [Celli et al., 2013], however, they focused on using the essays dataset for improving the performance on the Facebook one. Therefore they either do not report their results on the essays data, or their results are lower.

Facebook personality: A subset of the Facebook myPersonality dataset, collected by [Kosinski et al., 2013], was the central corpus for the WCPR 2013 workshop [Celli et al., 2013]. The status update collection and the user and personality information is enriched with the metadata about the user's social network structure, which many participants focused on. For example [Farnadi et al., 2013] obtain the highest classification performance of $F_1 = 0.62$ by ignoring the textual (LIWC) features fully, and using only the network-based features. The best binary classification results in extraversion, listed as SoA in our results

table, are reported by [Verhoeven et al., 2013], who simply use 2000 character trigrams as features, and rather experiment with composing classifiers on different personality traits and datasets into an ensemble. Their single component classifier (i.e., not an ensemble of classifiers; a single component classifiers predicts the expraversion only based on the features, not using the information from other traits) achieves the score of $F_1 = 0.66$ on Facebook extraversion. However, their scores are reported on a held-out test set, while our experiments perform 10-fold cross-validation over the full dataset, therefore the results are not directly comparable.

YouTube personality: The dataset of [Biel and Gatica-Perez, 2013] was central to the following WCPR workshop in 2014 [Celli et al., 2014], which focused on multimodal personality detection. Provided feature vectors extracted from videos included values such as pitch, eye gaze direction, camera proximity, energy, time speaking, voice rate etc. However, in one of the workshop tracks participants were allowed to use only the transcribed text. The best overall performance of $F_1 = 0.710$ classifying extraversion in the multimodal track was obtained by [Alam and Riccardi, 2014], using a cascaded model which combines classifiers for all five personality traits, learning customized combinations of audio and textual features for each of those. In the text-only track, the best score of $F_1 = 0.596$ is reported by [Verhoeven et al., 2014], using only bag of words features. Both character-based trigrams and LIWC features performed worse in their setup. For a fair comparison, we use this result as a state of the art (SoA) in our result table.

Experimental results

In the table 5.1 we can see that enriching the data with supersense annotations, whether directly (SUPER) or through WSD (SENSE-SUPER), outperforms the bag of words (WORD) settings in all cases, i.e., this information is never redundant or harmful. On the ESSAYS and FACEBOOK dataset, supersenses also outperform LIWC features. On the YOUTUBE dataset, we observe that LIWC performs better mainly due to the importance of interjections, that are not included in supersenses, but are part of the LIWC word categories. The following is an example of a YouTube transcript:

- *Hi, this is just a quick little update to let you know how I'm doing. Um, I went to a retreat with the church, um, last weekend and, um, when I got up there, I couldn't eat anything. Um, my –*

The disfluencies can be highly predictive for extraversion, as reported by [Mairesse et al., 2007]. Those are taken into account by the LIWC lexicon, but for supersenses, focused on nouns and verbs, these are irrelevant. Using both, the supersenses and LIWC, jointly (feature setup ALL) does not improve the performance though.

Dataset/setup	Personality essays		Facebook		YouTube	
	Accuracy	F-score	Accuracy	F-score	Accuracy	F-score
WORD	0.546	0.548	0.567	0.576	0.583	0.604
LIWC	0.557*	0.558	0.579	0.608	0.623	0.622
SENSE-SUPER	0.619*	0.584	0.613*	0.615	0.583	0.604
SUPER	**0.649***	**0.587**	0.621*	0.617	**0.585**	**0.605**
ALL	0.585*	0.582	0.629*	0.623	0.583	0.604
MAJ. BASELINE	0.517	0.498	0.616*	0.472	0.597	0.411
SoA	-	0.563	-	**0.700**	-	0.596
	[Mohammad et al., 2013b]		[Verhoeven et al., 2013]		[Verhoeven et al., 2014]	

Tab. 5.1: Classification performance on the task of predicting authors' extraversion on three datasets. We can see that the usage of supersenses outperforms the plain bag-of-words settings in all cases. Configurations with a statistically significant difference (McNemar's test, $p < 0.05$) from the ngram (WORD) setup are denoted with a star(*) on the accuracy column. There is no human annotator upper bound for the ESSAYS and FACEBOOK dataset, as the labels are based on the results of a questionnaire taken by each user. For the YouTube dataset labeled by external judges, the authors report an annotation intra-class correlation ICC = 0.76 and Cronbach's α = 0.63 for the extraversion trait.

Overall, the most predictive features for extroverts were related to words and concepts referring to friends, family, love, social life and positive emotions. For introverts, the highest ranked feature was the pronoun *I*, suggesting focus on the speakers themselves. This is in accordance with the previous work in this area [Pennebaker and King, 1999].

We also observe that the classification performance when annotating the supersenses directly with our supersense tagger (SUPER) is slightly higher than when linking the disambiguated senses to supersenses subsequently (SENSE-SUPER). This can be attributed to the error in the Lesk WSD algorithm compared to a trained supervised model, but judging from the higher performance difference on the FACEBOOK dataset, there is likely an effect of the supersense training data choice - our supervised tagging model is trained on a combination of SemCor [Miller et al., 1994] news data, sense-disambiguated Wikipedia and annotated social media posts (Twitter), which seems to provide a more robust word-supersense mapping than a direct WordNet lookup.

This hypothesis is further supported by the WordNet coverage analysis which we conducted - while for the ESSAYS dataset WordNet covers 90% of all occurring nouns and 92% of all verb instances, for the FACEBOOK dataset it contains only 67% of the nouns and 76% of the verbs used.

Intriguingly, the performance of all features together (ALL) on the ESSAYS and YOUTUBE dataset is not higher than when using only a subset of those. With a more detailed look, we hypothesize that this is due to overly emphasizing part-of-speech n-grams, which obtain high feature ranking in training, but do not perform well on the test sets. On the FACEBOOK dataset, the increase in performance when using all features (ALL) can be mainly attributed to the sentiment lexicons, as the emotions are more pronounced in the social media data.

5.1.2 Extraversion of fictional characters

It has been shown that the personality traits of book readers impact their literature preferences [Tirre and Dixit, 1995, Mar et al., 2009]. Psychology researchers also found that perceived similarity is predictive of interpersonal attraction [Montoya et al., 2008, Byrne, 1961, Chartrand and Bargh, 1999]. More explicitly, recent research [Kaufman and Libby, 2012] shows that readers of a narrative develop more favorable attitudes and less stereotype application towards a character, if his difference (e.g. racial) is revealed only later in the story. We therefore hypothesize that readers might have a preference for reading novels depicting fictional characters that are similar to themselves. Finding a direct link between reader's and protagonist's personality traits would advance the development of content-based book recommendation systems. As a first step to explore this hypothesis further, it needs to be determined if we are able to construct a personality profile of a fictional character in a similar way as it is done for humans. We construct our own dataset for this purpose.

Labeling the data

Traditionally, the gold standard for this supervised classification task is obtained by means of personality questionnaires, used for the Five-Factor Model, taken by each of the individuals assessed. This poses a challenge for fictional characters. However, strong correlations have been found between the self-reported and perceived personality traits [Mehl et al., 2006]. Our gold standard benefits from the fact that readers enjoy discussing the personality of their favourite book character online. A popular layman instrument for personality classification is the Myers-Briggs Type Indicator [Myers et al., 1985], shortly MBTI, which sorts personal preferences into four opposite pairs, or dichotomies, such as Thinking vs. Feeling, or Judging vs. Perceiving. While the MBTI validity has been questioned by the research community [Pittenger, 2005], the Extraversion scale is showing rather strong validity and correlation to the extraversion trait in the Five-Factor Model [McCrae and Costa, 1989, MacDonald et al., 1994]. Our study hence focuses on the Extraversion scale.

Our data was collected from the collaboratively constructed Personality Databank[2] where the readers can vote if a book character is, among other aspects, introverted or extraverted. While the readers used codes based on the MBTI typology, they did not apply the MBTI assessment strategies. There was no explicit annotation guideline, and the interpretation was left to the readers' intuition and knowledge.[3]

[2] `http://www.mbti-databank.com/`

[3] MBTI defines extraversion as "getting energy from active involvement in events, having a lot of different activities, enjoying being around people." In the NEO Five-Factor Inventory [Costa and McCrae, 2008], underlying facets of extraversion are warmth, gregariousness, assertiveness, activity, excitement seeking, and positive emotion.

Character	Book	E	I	Character	Book	E	I
Tyrion Lannister	Game of Thrones	52	1	Harry Potter	Harry Potter series	1	71
Cersei Lannister	Game of Thrones	48	7	Severus Snape	Harry Potter series	1	65
Joffrey Baratheon	Game of Thrones	41	1	Gandalf	Lord of the Rings	1	59
Ron Weasley	Harry Potter series	37	4	Yoda	Star Wars series	0	58
Jamie Lannister	Game of Thrones	38	9	Jon Snow	Game of Thrones	1	47
Draco Malfoy	Harry Potter series	33	4	A. Dumbledore	Harry Potter series	4	46
Anakin Skywalker	Star Wars series	30	6	Ned Stark	Game of Thrones	0	41
Robert Baratheon	Game of Thrones	28	2	Aragorn	Lord of the Rings	1	41
Gimli	Lord of the Rings	19	2	Frodo	Lord of the Rings	1	40
Jar Jar Binks	Star Wars series	12	2	Bran Stark	Game of Thrones	1	36

Tab. 5.2: Extraverts (E) and introverts (I) with the highest number of user votes.

We have collected extraversion ratings for 298 book characters, of which 129 (43%) are rather extraverted and 166 (56%) rather introverted. Rated characters come from a wide range of novels that the online users are familiar with, often covering classical literature which is part of the high school syllabus, as well as the most popular modern fiction, such as the Harry Potter series, Twilight, Star Wars or A Game of Thrones. A sample of the most rated introverts and extraverts is given in Table 5.2. The rating distribution in our data is strongly U-shaped. The percentage agreement of voters in our data is 84.9%, calculated as:

$$P = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{k}\frac{n_{ij}(n_{ij}-1)}{n(n-1)}$$

where $k = 2$ (introvert, extravert), N is the number of book characters and n the number of votes per character. Voters on the website were anonymous and cannot be uniquely identified. There is no correlation between the extraversion and the gender of the character.

Our set of English e-books, where we had the full text available for processing, covered 220 of the characters from our gold standard. We have built two systems to assess the following:

1. **Direct speech**: Does the style and content of character's utterances predict his extraversion in a similar way as it was shown for living individuals?
2. **Actions**: Is the behavior, of which a character is an agent, predictive for extraversion?

In the following, we present the experimental settings and results for each of the systems.

Direct speech of fictional characters

The system for the direct speech resembles the most the previous systems developed for author personality profiling, e.g. the systems developed for the ESSAYS [Mairesse et al.,

2007] and FACEBOOK [Celli et al., 2013] datasets we used in the previous chapters, and therefore provides the best opportunity for comparison between human individuals and fictional characters. On top of the comparison to previous research, we exploit the sense links between WordNet and VerbNet to extract additional features - an approach which is novel for this type of task.

Extraction and assignment of speech We process the book text using freely available components of the DKPro framework [Gurevych et al., 2007]. The most challenging task in building the direct speech dataset is assigning the correct speaker to the direct speech utterance. We benefit from the `epub` format of the e-books which defines a paragraph structure in such a way, that only the indirect speech chunk immediately surrounding the direct speech can be considered:

```
<p> John turned to Harry.
"Let's go," he said.</p>
```

Given a large amount of text available in the books, we aim at high precision of our speaker labels, and we therefore discard all utterances with no explicit speaker (i.e., 30-70% of the utterances, depending on the book), as the performance of state-of-the-art systems on such utterance types is still fairly low [O'Keefe et al., 2012, He et al., 2013, Iosif and Mishra, 2014]. We also ignore the coreferences, as the conventional coreference resolution systems did not perform well on this type of data in our initial experiments. Fortunately, for representing a character in a book, the number of utterances with an explicitly mentioned name is sufficient in most cases. However, even with the explicitly mentioned named entity, the assignment of a speaker is not trivial, as detailed below. We adapt the Stanford Named Entity Recognizer [Finkel et al., 2005] to consider titles (Mr., Mrs., Sir...) as a part of the name and to treat the first person I as a named entity. Yet, identifying only the named entity PERSON in this way is not sufficient. On our evaluation sample consisting of *A Game of Thrones* and *Pride and Prejudice* books (the former annotated by us, the latter by [He et al., 2013]), 20% of utterances with an explicitly named speaker were not recognized. Of those correctly identified as a Person in the adjacent indirect speech, 17% were not the speakers. Therefore we implemented a custom heuristics (Algorithm 1), which additionally benefits from the WordNet semantic classes of verbs, enriching the speaker detection by grabbing the nouns. With this method we retrieve 89% of known speakers (the entity being successfully identified), of which 92% are assigned correctly (in the remaining cases, a wrong selection from the range of mentioned entities was made). Retrieved names are grouped based on string overlap (e.g. *Ser Jaime* and *Jaime Lannister*), excluding the match of the last name, and manually corrected for non-obvious groupings (such as *Margaret* and *Peggy*).

Algorithm 2 Assign speaker

1: *nsubj* ← subjects in adjacent indirect speech
2: **if** $count(nsubj(i) = \text{PERSON}) = 1$ **then** $speaker \leftarrow nsubj$
3: **else if** $count(nsubj(i) = \text{PERSON}) \geq 1$ **then** $speaker \leftarrow$ the nearest one to $directSpeech$
4: **else if** $directSpeech$ preceded by VERB.COMMUNICATION **then** $speaker \leftarrow$ the preceding noun(s)
5: **else if** $directSpeech$ followed by VERB.COMMUNICATION **then** $speaker \leftarrow$ the following noun(s)
6: **else if** $directSpeech$ followed by gap & VERB.COMMUNICATION **then** $speaker \leftarrow$ the noun(s) in gap
7: **else if** $directSpeech$ preceded by gap & VERB.COMMUNICATION **then** $speaker \leftarrow$ the noun(s) in gap
return $speaker$

Our experimental data consists of usable direct speech sets of 175 characters - 80 extraverts (E) and 95 introverts (I) - containing 289 274 words in 21 857 utterances (on average 111 utterances for E and 136 for I, as I are often central in books).[4]

Classification approach for direct speech All speech utterances of one book character are represented as one document (one classification instance) in our system. Due to the relatively small dataset size we use the leave-one-out classification setup, using the support vector machines (SVM-SMO) classifier, which performs well on comparable tasks [Celli et al., 2013].

Since the top-down approach, i.e., not focusing on individual words, has been found more suitable for the personality profiling task on smaller datasets [Mohammad and Kiritchenko, 2013], we aim on capturing additional phenomena on a higher level of abstraction. The main part of our features is extracted on sense level. We use the most frequent sense of WordNet [Miller, 1995] to annotate all verbs in the direct speech (a simple but well performing approach for books). We then label the disambiguated verbs with their WordNet supersenses and measure the frequency and occurence of each of the supersenses in the document. Additionally, we use the lexical-semantic resource UBY [Gurevych et al., 2012] to access the WordNet and VerbNet information, and to exploit the VerbNet sense-level links which connect WordNet senses with the corresponding 273 main VerbNet classes [Kipper-Schuler, 2005]. These are more fine-grained (e.g. pay, conspire, neglect, discover) than the WordNet supersenses (e.g. cognition, communication, motion, perception). WordNet covered 90% and VerbNet 86% of all the verb occurences.

Table 5.3 shows the precision, recall, F_1-score and accuracy for extraversion and introversion as a weighted average of the two class values.

We can see that the bottom-up word based approach is outperformed by top-down semantic approaches which employ a more abstract feature representation. As in previous work, LIWC features exhibit good performance. However, the highest performance is achieved

[4]The number of classification units (characters) in the dataset is comparable to ongoing personality profiling challenges - see `http://pan.webis.de`

Feature set	Precision	Recall	F-score	Accuracy
WORD	.519	.514	.515	.514
LIWC	.555	.560	**.552**	.560*
SENSE-SUPER	.527	.548	.528	.548*
SENSE-SUPER-VN	.649	.617	**.572**	.617*
ALL	.550	.632	**.588**	.632*
BASELINE	.295	.543	.382	.543
Percentage human agreement:				.849

Tab. 5.3: Weighted precision (P), recall (R), F-score (F) and accuracy (A) for a direct speech system, in each line using only the given group of features. Configurations with a statistically significant difference (McNemar's test, $p < 0.05$) from the ngram (WORD) setup are denoted with a star(*) on the accuracy column. The McNemar's test values are estimated from a subset of classification folds.

Introvert Feat.group	Features	Merit
unigrams	*reason, trouble, strange, indeed*	0.24-0.19
bigrams	*this time, tell me, I hope*	0.19-0.16
LIWC	Negate, Discrepancy,Insight, Exclusion	0.18-0.13
WordNet	stative, creation, cognition	0.15-0.09
VerbNet	lodge, hunt, defend	0.23-0.19
Style	modal verbs, neg, sbar, articles	0.19-0.14
Extravert Feat.group	Features	Merit
ngrams	*we, hurry, fat, dirty*	0.24-0.19
LIWC	We, Inclusion, Pronoun, Body	0.18-0.09
WordNet	motion, contact, communication, body, perception, change	0.14-0.07
VerbNet	get, talk, substance emission	0.18-0.15
Style	pronoun We, whadjp, type-token ratio., interjections	0.20-0.14

Tab. 5.4: The most predictive features for each group for speaker's extraversion and introversion.

employing the VerbNet verb classes with WordNet word-sense disambiguation. Also stylistic features contribute substantially to the classification despite the mixture of genres in our book corpus - especially frequencies of modal verbs and part-of-speech ratios were particularly informative. The most predictive features from each group are listed in Table 5.4 together with their correlation merit, and compared with previous work in Table 5.5.

Correlation merit [Hall, 1999] is a score used in the Correlation Feature Selection (CFS) algorithm. CFS is a simple filter algorithm that ranks feature subsets according to a correlation-based heuristic evaluation function. The bias of the evaluation function is toward subsets that contain features that are highly correlated with the class and uncorrelated with each other. In WEKA, the correlation merit is calculated in the following way:

$$M_S = \frac{kr_{cf}}{\sqrt{k + k(k-1)r_{ff}}}$$

where M_S is the heuristic merit of a feature subset S containing k features, r_{cf} is the mean feature-class correlation ($f \in S$), and r_{ff} is the average feature-feature intercorrelation. The numerator can be thought of as providing an indication of how predictive of the class a set of features are; the denominator of how much redundancy there is among the features.

In accordance with the experiments of [Pennebaker and King, 1999], we observe more frequent exclusions (e.g. *without, but*), hedging and negation expressed by introverts, and inclusion (e.g. *with, and*) by extraverts. Extraverts talk more in first person plural (using *we, us* etc.), use more back-channels and interjections, and talk more about aspects related to their body. Introverts show more rationalization through insight words and more factual speech using fewer pronouns.

Additionally, the predictive features in Table 5.4 confirm the broad psychological characteristics of both types in general, i.e., for introverts the rationalization, uncertainty and preference for individual or rather static activities, and for extraverts their spontaneity, talkativeness and preference for motion. Furthermore, we observe certain directness in extraverts' speech - note the predictive words *fat* and *dirty* and frequent descriptions of body functions.

Exploiting the links between lexical-semantic resources (performing WordNet word-sense disambiguation and using VerbNet verb classes linked to the disambiguated senses) was particularly beneficial for this task. WordNet supersenses for verbs alone are too coarse-grained to capture the nuances in direct speech, and experiments with fine-grained VerbNet classes without WSD resulted in noisy labels. We did not confirm the previously reported findings on emotional polarity - we observe that the genre of the books (e.g. love romance vs horror story) have blurred the subtle differences between individual characters, unfortunately the dataset size did not allow for genre distinctions. Furthermore, a perceived extravert in our case can be a pure villain (Draco Malfoy, Joffrey Baratheon...) as well as a friendly

Feature	I/E	Reference	Feature	I/E	Reference
Predictive also in our data:			No effect in our data:		
Pronoun 'we'	-/+	[Mairesse et al., 2007]	Neg. emot.	+/-	[Pennebaker and King, 1999]
Tentative, unsure	+/-	[Pennebaker and King, 1999]	Pos. emot.	-/+	[Pennebaker and King, 1999]
Exclusive	+/-	[Pennebaker and King, 1999]	Self-ref.	-/+	[Pennebaker and King, 1999]
Inclusive	-/+	[Pennebaker and King, 1999]	Formality	+/-	[Dewaele and Furnham, 1999]
Insight	+/-	[Pennebaker and King, 1999]	Elaborated	+/-	[Mairesse et al., 2007]
Nouns, articles	+/-	[Dewaele and Furnham, 1999]	Long sent.	+/-	[Mairesse et al., 2007]
Lexical richness	+/-	[Dewaele and Furnham, 1999]	Social	-/+	[Mairesse et al., 2007]
Negations	+/-	[Dewaele and Furnham, 1999]			
Body functions	-/+	[Dewaele and Furnham, 1999]			
Interjections	-/+	[Mairesse et al., 2007]			

Tab. 5.5: Comparison of our results to previously reported predictive features for speaker's extraversion (E), resp. introversion (I). We list publications where these features were, to our knowledge, reported as novel.

companion (Gimli, Ron Weasley...), while the evil extravert types are possibly rarer in the experiments using first-person writing of real individuals (such as essays or social media) than in fiction, or are more likely to fit into the MBTI definition of extraversion than into the definition used in the Five Factor Model of personality (which the previous studies used). Another potential cause, based on the error analysis, is the different target of the same sentiment for extraverts and introverts. For example, the n-gram "I fear" is highly predictive for an introvert in our data while extraverts would rather use formulations to imply that others should fear. Similarly to [Nowson et al., 2005], we did not find any difference in the formality measure of [Heylighen and Dewaele, 2002]. Neither we did in the complexity of sentences as per the parse tree depth and sentence length. It is probable that these aspects were also impacted by our broad variety of author styles (F. Dostoyevsky vs J. K. Rowling).

Actions of fictional characters

While psycholinguists and consequenlty NLP researchers analyzed the relation between speech, resp. writing, and personality of an individual, psychologists often evaluate extraversion through behavioral personality questionnaries [Costa and McCrae, 2008, Goldberg et al., 2006], with questions such as *What would you do if...?*. We hypothesize that similar behavior shall be predictive for extraversion of fictional characters as perceived by the readers.

Action extraction For our purpose, we define actions as the subject, verb and context of a sentence, where the subject is a named entity Person and the context is either a direct object in relation *dobj* to the verb or a first child of the adjacent verb phrase in a parse tree. After grouping the actions per character, the subject name is removed. For example, a sample of actions of the character Eddard Stark of Game of Thrones would be: *X paused a moment, X studied his face, X changed his mind, X unrolled the paper, X said* etc., visualized in Figure 5.2. We obtained 22 030 actions for 205 characters (102 E, 116 I), with on average 100 actions for E and 101 for I. Note that also actions for those characters who do not talk enough in the books (often first-person perspectives) could be used.

Action classification setup In the system based on actions we use only a subset of the features described in 5.1.2. From the lexical features, we focus on the 500 most frequent verbs and dependency word pairs. Semantic features are used the same way as in 5.1.2, profiting from LIWC, WordNet, VerbNet, and the sentiment lexicons. From the stylistic features, we use the part-of-speech bigrams and trigrams, verb modality and verb tense.

Classification results on actions Table 5.6 shows the performance of the classification models based on the protagonists' actions, using different feature groups.

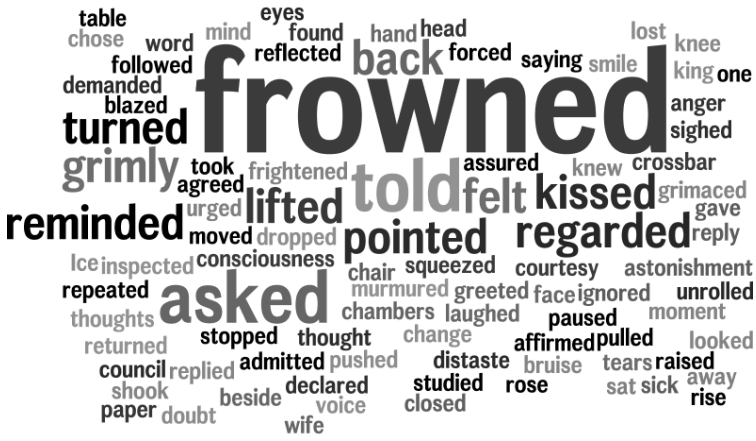

Fig. 5.2: A revealing word cloud of the most frequent words from the actions of which Eddard Stark (Game of Thrones) is a subject. Size is proportional to the frequency of a word.

Feature set	Precision	Recall	F-score	Accuracy
WORD	.539	.506	.505	.507
LIWC	.600	.577	**.567**	.577*
SENSE-SUPER	.517	.518	.517	.518
SENSE-SUPER-VN	.599	.583	**.578**	.583*
ALL	.600	.623	.598	.623*
BASELINE	.267	.517	.352	.517
Percentage human agreement:				.849

Tab. 5.6: Weighted precision (P), recall (R), F-score (F) and accuracy (A) for actions - in each line for a system using only the given group of features. WordNet stands for WordNet semantic labels, VerbNet setup uses the WordNet-VerbNet links. Configurations with a statistically significant difference (McNemar's test, $p < 0.05$) from the ngram (WORD) setup are denoted with a star(*) on the accuracy column.

Due to the lack of previous NLP experiments on this task, we compare our features to the actions measured in the International Personality Item Pool [Goldberg et al., 2006], frequently used personality assessment questionnaire (Table 5.7).

The most predictive features of this model capture the activity and excitement seeking facets of extraversion. Stylistic features reflect the complexity difference of the verb phrases (*John jumped vs. John thought about it*), extraverts being characterized by simpler constructs in the form *subject + verb*, as opposed to introverts (described more often by *subject + verb + preposition + object*). Semantic features lead to a higher precision than stylistic ones. Sense-linked semantic classes of VerbNet demonstrate the preference of extraverts for being active and expressing themselves - they *jump*, *fight, shout, run in* and *run out*, *eat* and *drink*, *see* and *hear* and get easily *bored*. Extraverts in books also often *bring* or *hold* something. Introverts, on the other hand, seem to favor slow movements - while they are *thinking, reflecting, creating*, looking for *explanations* and *find out* solutions, they tend to *lie down, sit* or *walk*, eventually even *sleep* or *snooze*. The uncertainty typical for introverts is

Extravert	
International Personality Item Pool:	likes to party, feels comfortable around people, starts conversations, talks to many people, enjoys being a center of attention, makes friends easily, takes charge, captivates people, feels at ease with a company, is skilled in handling social situations
Our experiment:	bring (VN), consume (VN), contiguous location(VN), holding (VN), social (WN), motion (WN), emotion (WN) Leisure (LIWC), Home (LIWC), Family (LIWC), fight, march, care, take, jump, shriek, clear throat, bore, get to, come in, agree, hold, hear, inform, sell, come forward
Introvert	
International Personality Item Pool:	Doesn't talk much, stays in the background, has little to say, does not draw attention, has difficulties to approach others, is quiet around strangers, feels uncomfortable around others,does not show feelings, is a private person, waits to be led
Our experiment:	snooze (VN), conceal (VN), wish (VN), stative (WN), creation (WN), walk, sleep, lay, know, maintain, expect, hope, find out, might, help, explain

Tab. 5.7: Characteristic actions for extraverts and introverts as assessed in the IPIP personality questionaire, compared to our most informative features

also notable in their actions, as they often *hope* or *wish* for something they *might* like to do. Additionally, semantic classes Social and Family, reported as correlated to extraversion by [Pennebaker and King, 1999] and not confirmed in our first model, became predictive in protagonists' actions.

Discussion Also in this task, the VerbNet semantic classes brought significant improvement in performance. The classification model based on actions performed slightly better than the direct speech model, achieving better precision but lower recall. Previous work predicting authors' extraversion from the stream of consciousness essays [Mairesse et al., 2007, Celli et al., 2013, Neuman and Cohen, 2014] reached an accuracy of up to 60% on a balanced dataset, our models on fiction achieved 63% and 62% with the majority baseline of 54% and 52% respectively. While surely not directly comparable, this result is promising for using fiction as an additional source of training data. The findings of [Mairesse et al., 2007, Biel and Gatica-Perez, 2013] and [Aran and Gatica-Perez, 2013] on multimodal datasets suggested that the personality traits are easier to detect from behavior than from person's verbal expression. Our results are not conclusive in this respect, however, the predictive features based on the character's behavior are seemingly easier to interpret for a human analyst.

Combination of the direct speech and action systems

In a follow-up experiment, we have combined both systems together. However, since the characters used for training each of the systems are only partly overlapping (for example, main characters in the *Stranger* of Camus or the *Lolita* of Nabokov do not provide enough direct speech training data, yet they are relatively rich on actions, including thinking processes), we could only use a subset of 115 characters - 68 introverted, 47 extraverted, providing a high majority baseline of 59% accuracy. Using all features, we achieved an accuracy of 76%, i.e., higher than when using each of the systems separately. However, the dataset is too small for drawing generalizable conclusions, as the most prominent features are rather theme-specific (e.g., Star Wars jedi and Tolkien's elves being introverts).

5.1.3 Extraversion conclusions

We have presented extraversion prediction experiments for both real subjects and fictional characters on multiple datasets. We have shown that in all cases the supersense information outperforms the bag of words classification setup and contributes to the classification model with a useful additional piece of information. In the following sections, we examine if this contribution generalizes beyond personality profiling tasks.

5.2 Gender prediction experiments

Studying gender differences has been a popular psychological interest over the past decades [Gleser et al., 1959, McMillan et al., 1977]. Traditional studies worked on small datasets, which often led to contradictory results – [Mulac et al., 1990] cf. [Pennebaker et al., 2003]. The first detailed gender study on a larger scale was performed by [Newman et al., 2008] on 14,324 samples from 70 different studies (conversation, exams, fiction etc.). According to them, women are more likely to include pronouns, verbs, references to home, family, friends and to various emotions. Men tend to use longer words, more articles, prepositions and numbers. Men also swear more often and discuss current concerns (e.g. money, leisure or sports). [Schler et al., 2006] apply machine learning techniques to a corpus of 37,478 blogs from blogger.com. They found differences in topics which men and women discuss. More recent author profiling experiments [Rangel et al., 2014, Rangel et al., 2015] revealed that gender can be well predicted from a large spectrum of features, ranging from emotions, grammar and abbreviation usage to social network metadata, web traffic [Culotta et al., 2015] and apps installed [Seneviratne et al., 2015]. Most of these experiments were based on self-reported gender in blogs and social media profiles.

Setup	WORD	LIWC	SENSE-SUPER	SUPER	ALL	BASELINE	SoA
Accuracy	0.656	0.701*	0.673*	0.668	0.743*	0.512	0.885

Tab. 5.8: Gender classification accuracy. Configurations with a statistically significant difference (McNemar's test, $p < 0.05$) from the ngram (WORD) setup are denoted with a star(*) on the accuracy column.

5.2.1 Dataset

For our experiments on gender prediction, we use the balanced dataset of 3,100 blogs, collected by [Mukherjee and Liu, 2010] and introduced in Chapter 3 of this thesis.

5.2.2 Gender results and conclusions

Since our experimental data is balanced, we follow the practice of previous publications on it and report only the classification accuracy rather than F-score. Our results are displayed in the table 5.8. There is no human upper bound, since humans are typically worse than machines in this task - e.g., in a study we performed in [Flekova and Gurevych, 2013] with 20 annotators, the human accuracy on a sample of internet blogs from [Rangel et al., 2013] was 55%. This is partly because humans are relying only on topical stereotypes in their decision, while classifiers can learn fine-grained stylistic differences [Flekova et al., 2016a]. The best previous results (SoA) of 0.88 on this dataset were achieved by [Mukherjee and Liu, 2010], who introduced an advanced ensemble feature selection algorithm, combined with dynamic part-of-speech patterns as features. Other experiments [Schler et al., 2006, Yan and Yan, 2006, Argamon et al., 2007] using simpler classification settings, such as those described above, achieve an accuracy between 0.61 and 0.79.

For the gender classification, supersenses show only mild improvement in performance. A more detailed examination of our results shows, in accordance with previous work [Pennebaker et al., 2003, Newman et al., 2008, Schler et al., 2006, Mukherjee and Liu, 2010], that the stylistic features are more predictive for gender than the content-based ones. For this reason, features based on LIWC, which contains also syntax-oriented word lists (e.g., personal pronouns, 1st person pronouns, articles...) and word length measures, perform better than the meaning-focused supersenses operating on nouns and verbs. Even higher increase in performance is then triggered by adding the part-of-speech n-grams and frequencies, and the length-based and sentiment features (as detailed in 5.1). From supersenses themselves, the most predictive are features capturing feelings, emotions, persons and food.

5.3 Sentiment experiments

Sentiment classification has been a widely explored task which received a lot of attention. [Hu and Liu, 2004] proposed a lexicon-based algorithm, based on a sentiment lexicon generated using a bootstrapping strategy with some given positive and negative sentiment word seeds and the synonym and antonym relations in WordNet. In [Kim and Hovy, 2004], a similar approach was also used. Later approaches use supervised learning, with state of the art performance achieved using neural network algorithms with word embedding features [Kim, 2014, Zhao et al., 2015, Zhang and Wallace, 2015].

The goal of subjectivity classification is to label sentences as either subjective and objective [Wiebe et al., 1999]. An objective sentence expresses some factual information, while a subjective sentence usually gives personal views and opinions. Most existing approaches to subjectivity classification are based on supervised learning. For example, the early work reported in [Wiebe et al., 1999] performed subjectivity classification using the naïve Bayes classifier with a set of binary features, e.g., the presence in the sentence of a pronoun, an adjective, a cardinal number, a modal other than *will* and an adverb other than *not*. Subsequent research also used other learning algorithms and more sophisticated features. [Pang and Lee, 2004] demonstrate that subjectivity detection can be a useful input for a sentiment classifier. Supersenses are a natural candidate for subjectivity prediction, as we hypothesize that nouns and verbs in the subjective and objective sentences often come from different semantic classes (e.g. VERB.FEELING vs. VERB.COGNITION).

5.3.1 Datasets

Sentiment data: The Movie Review dataset, published by [Pang and Lee, 2005][5], has become a standard machine learning benchmark task for binary sentence classification. It contains 5331 positive and 5331 negative sentences processed from high-ranking and low-ranking movie reviews.

Subjectivity data: [Pang and Lee, 2004] compose a publicly available dataset[6] of 5000 subjective and 5000 objective sentences, classifying them with a reported accuracy of 90-92% and further show that predicting this information improves the end-level sentiment classification on a movie review dataset. [Kim, 2014] and [Wang and Manning, 2013] further improve the performance through different machine learning methods.

[5] http://www.cs.cornell.edu/people/pabo/movie-review-data/rt-polaritydata.tar.gz
[6] https://www.cs.cornell.edu/people/pabo/movie-review-data/rotten_imdb.tar.gz

Setup	WORD	LIWC	SENSE-SUPER	SUPER	ALL	BASELINE	SoA [Kim, 2014]
Accuracy	0.744	0.758*	0.751*	0.761*	0.764*	0.500	0.815

Tab. 5.9: 10-fold cross-validation accuracy of our system for the sentiment classification task on movie review data. We can see that any conceptual abstraction over words is helpful in the classification. Configurations with a statistically significant difference (McNemar's test, $p < 0.05$) from the ngram (WORD) setup are denoted with a star(*).

Setup	WORD	LIWC	SENSE-SUPER	SUPER	ALL	BASELINE	SoA [Zhang et al., 2016]
Accuracy	0.910	0.923*	0.925*	0.926*	0.929*	0.500	0.939

Tab. 5.10: 10-fold cross-validation accuracy of our system for the subjectivity classification task Configurations with a statistically significant difference (McNemar's test, $p < 0.05$) from the ngram (WORD) setup are denoted with a star(*).

5.3.2 Results and error analysis

Our classification results for the sentiment dataset are displayed in Table 5.9. We can see that any conceptual abstraction over words is helpful in the classification. Supersenses tagged directly lead to better classification results than those annotated via WordNet-based sense disambiguation. This is mainly caused by the limited coverage of WordNet, which results in annotating mostly the very frequent words (such as the verb *to be*), contributing little to distinguishing the two datasets. The supervised supersense tagger, which is able to annotate words beyond WordNet, deals better with the variety of expressions in online reviews. Adding sentiment lexicons (NRC, SentiWordNet) and POS frequencies further improves the performance of the classifier (line ALL).

A detailed analysis of the supersense-tagged data and the classification output revealed that supersenses help to generalize over rare terms. Noun concepts such as GROUP, LOCATION, TIME and PERSON appear somewhat more frequently in positive reviews, while certain verb supersenses such as PERCEPTION, SOCIAL and COMMUNICATION are more frequent in the negative ones. On the other hand, the supersense tagging introduces additional errors too - for example the director's *cut* is persistently classified into FOOD. None of our SVM-based classification settings outperforms the state of the art (SoA) of [Kim, 2014], which relies on a convolutional neural network algorithm using only generic word embeddings (word2vec, [Mikolov et al., 2013c]) as features. We attempt to rival this algorithm with our neural network system architecture in the next chapter.

Table 5.10 displays our results on the subjectivity dataset. Also in this case, any concept abstraction features (LIWC, SENSE-SUPER, SUPER, ALL) help to improve the classification performance - in this case, both supersense annotation approaches are even more informative than the LIWC features (however, the standard error of the accuracy ranges between 0.002 – 0.003).

Based on a manual feature analysis, subjective sentences contain more verbs of supersense PERCEPTION, while objective ones more frequently feature the supersenses POSSESSION and SOCIAL. Nouns in the subjective category are characterized by supersenses COMMUNICATION and ATTRIBUTE, while in objective ones the PERSON, ARTIFACT, ACT, COGNITION and POSSESSION are more frequent. The state of the art classification performance of [Zhang et al., 2016] (SoA) is in this case achieved with a convolutional neural network architecture, too, and therefore discussed in the following chapter, which focuses on deep learning.

5.3.3 Sentiment and subjectivity tasks conclusions

We have shown that also for the task of sentiment and subjectivity classification, the classifier benefits from the concept generalization achieved through supersenses. While the most predictive sentiment classification features are still the emotion-related adjectives (e.g., *bad, boring* and *pointless* for negative reviews vs. *enjoyable, moving* and *thoughtful* for positive reviews), supersenses help to group unique nouns and verbs into meaningful high-level concepts and benefit from the differences between concepts discussed in the positive and negative comments (e.g., positive reviews discussing actor's performance or storyline complexity, while negative ones often mention the movie in general, and author's feelings rather than movie attributes). Subjective messages differ from objective ones in a similar manner. While we did not manage to outperform the state of the art in the explored tasks, we demonstrate that the usage of supersenses is a promising approach, which is worth incorporating into more advanced machine learning configurations.

5.4 Chapter summary

In this chapter, we empirically evaluated the utility of concept generalization in a range of document classification tasks - extraversion prediction, gender classification, sentiment polarity prediction and subjectivity classification. This generalization was achieved through supersense tagging, proposed in the previous chapter. We have introduced a new task of personality prediction of fictional characters, proposed a methodology to perform this task, and shown that the achieved results are comparable to previous findings of psychologists for human personality. The main difference in predicting personality trait on fictional data was the presence of villain characters labeled by readers as extraverts, while in the previous studies with real humans the extraverts are typically nice and friendly. Additionally, in contrast to the real-world spoken dialogue [Mairesse et al., 2007], we found that introverts in the books speak as often and (at least) as long as their extraverted counterparts.

We demonstrated that supersenses contribute towards an increased classification performance on a majority of the downstream classification tasks we examined. We illustrated how our approach can be extended to additional lexical-semantic resources suitable for the

task at hand, presenting a method to extend our approach beyond WordNet supersenses, using the sense-level links between WordNet and VerbNet. We have shown that direct supersense annotation with a pre-trained model leads to better results than accessing supersense labels through fine-grained word senses. We analyzed the most informative features for these tasks and proposed explanations for the way supersenses contribute to the classification. In the next chapter, we analyze how supersense features can be used as dense vectors within the deep learning architectures, and if the performance of such approaches exceeds the traditional supervised text classification methods such as support vector machines used with supersense features in this chapter.

6 Concept Generalization Deep Learning Experiments

> *Shall I refuse my dinner because I do not fully understand the process of digestion?*
>
> **— Oliver Heaviside**

In Chapter 3, we have introduced techniques to determine a sense of a word. In Chapter 4, we have shown, how this sense information can be used to access high-level semantic labels of individual words, called supersenses. We also proposed that the supersense information can be annotated directly, and introduce the concept of supersense embeddings, which we used to build a neural supersense tagging model. In Chapter 5, we have then shown that supersenses improve classification results, and moreover, that tagging supersenses directly with our model is more efficient than tracing them through a fine-grained WSD algorithm. In this chapter, as illustrated in Figure 6.1, we show how the tagged supersenses can be combined with our supersense embedding vectors from Chapter 4, to leverage the additional semantic information in neural-network classification approaches. We show that combining the state-of-the-art machine learning methods with the lexical-semantic knowledge, accessed as dense vectors, enables to outperform either of the approaches alone (neural networks with word embeddings, and support vector machines with supersense tags as features).

6.1 Background

The document classification area of NLP was for a long period dominated by machine-learning techniques that used linear models, such as support vector machines or logistic regression, trained over very sparse, high-dimensional feature spaces. Neural networks, which emerged only recently in the NLP applications, have two main advantages to these approaches - moving from linear to non-linear machine-learning models (i.e., capturing more complex feature relations), and moving from sparse to dense feature spaces (i.e., enabling more efficient computations over the semantic space of words).

In linear models, combining the features is very important to transform the input space, and the expert insight in the engineered features can make the data more linearly separable.

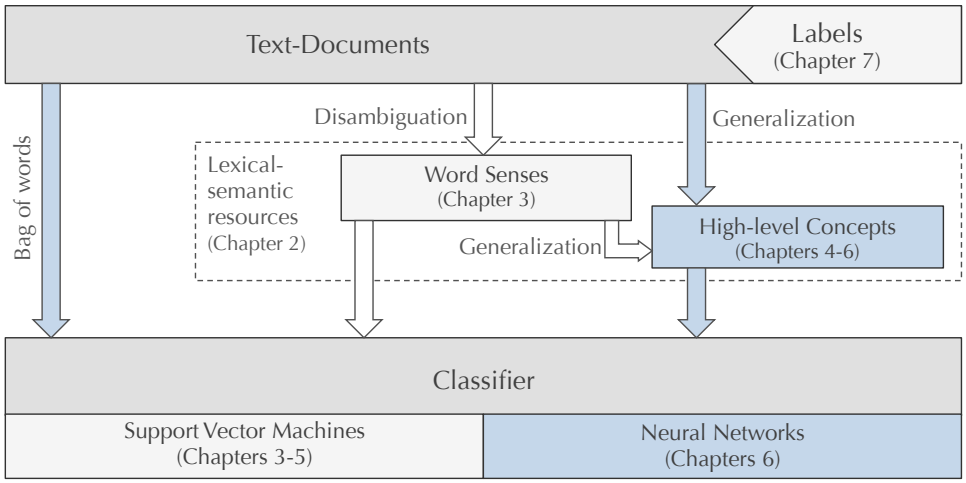

Fig. 6.1: The concepts and workflows of this thesis explored in this chapter (highlighted blue).

At the same time, considering the huge space of possible feature combinations, the expert work is very tedious and time consuming [Stoffel et al., 2015]. In neural network, the non-linearity defined by the architecture alleviates the need for extensive feature engineering, as we aim to find the optimal non-linear combinations of core features (usually words or characters) automatically.

There have been also other approaches to deal with non-linearity in machine learning, most famously using kernel machines [Schölkopf and Smola, 2002]. This approach led to good results at the time, yet since the hardware capabilities developed, neural networks became more popular for dealing with large-scale data.

6.1.1 Historical context

The popularity of neural networks has oscillated over time, and dramatically increased over the past ten years, as the computing infrastructure rapidly improved, enabling to process unprecedented amounts of training data. The name neural network is inspired by a human brain metaphor, rather than being realistic models of biological brain function. The metaphor depicts the brain as a computation mechanism consisting of a network of neurons, i.e. computational units that have scalar inputs and outputs. The neuron multiplies each input by its associated weight, and then sums them up, applies a non-linear function to the result, and passes it to its output. The output of a neuron may feed into the inputs of one or more neurons. Networks with more than one inner ("hidden") layer between the data inputs and the final outputs are called deep networks, hence the name deep learning.

The early theories of biological learning appeared in 1940s [McCulloch and Pitts, 1943], followed by the first implemented models of a single neuron, such as perceptron [Rosenblatt, 1958]. In 1980s, researchers put an emphasis on the way the neurons are connected, introducing the backpropagation algorithm [Rumelhart et al., 1986] and training networks with 1-2 hidden layers. An important concept from this period is the distributed representation [Hinton, 1986], saying that each input to a system should be represented by many features, and each feature should be involved in the representation of many possible inputs. After experiencing harsh criticism in the 90s, related to the rise of kernel-based and bayesian classification methods, deep learning witnesses a new boom for the past several years, starting with their remarkable success in image classification [Krizhevsky et al., 2012] and speech recognition [Dahl et al., 2012, Hinton et al., 2012a, Seide et al., 2011] and followed by the notable improvements they brought to the text-based NLP tasks in the past 2-3 years.

The idea of representing words as dense vectors for input to a neural network was introduced by [Bengio et al., 2003] when focusing on language modeling. For NLP tasks it was first applied in the work of [Collobert and Weston, 2008]. Using embeddings for representing not only words but arbitrary features such as part-of-speech tags was popularized only very recently by [Chen and Manning, 2014].

6.1.2 Types of neural network architectures

Fully connected feed-forward neural networks A basic type of a neural network is a fully connected feed-forward one, illustrated in figure 6.2. Neurons (the circles) are arranged in layers, with the arrows depicting the flow of information. In practice, each connection (arrow) has an assigned weight, expressing its importance. Between the input and the output can be an arbitrary number of hidden layers capturing the relations. As the name suggests, the neurons between two subsequent layers are fully connected (i.e., each one from layer 1 with each one from layer 2). Inside each neuron is a function, typically non-linear, which is applied to the input value. Common choices for such a function are the sigmoid, tanh, and the simple, yet powerful rectified linear unit ReLU [Glorot et al., 2011]. The result of the modified input is then passed forward in the network.

The simplest neural network is the perceptron. Perceptron consists of a linear function of its inputs multiplied by the weights, called the *potential*, and an activation function, step-like or continuous [Rosenblatt, 1958], which transforms it to the output value. Adding hidden layers results in the Multi Layer Perceptron, enabling to distinguish the data which are not linearly separable.

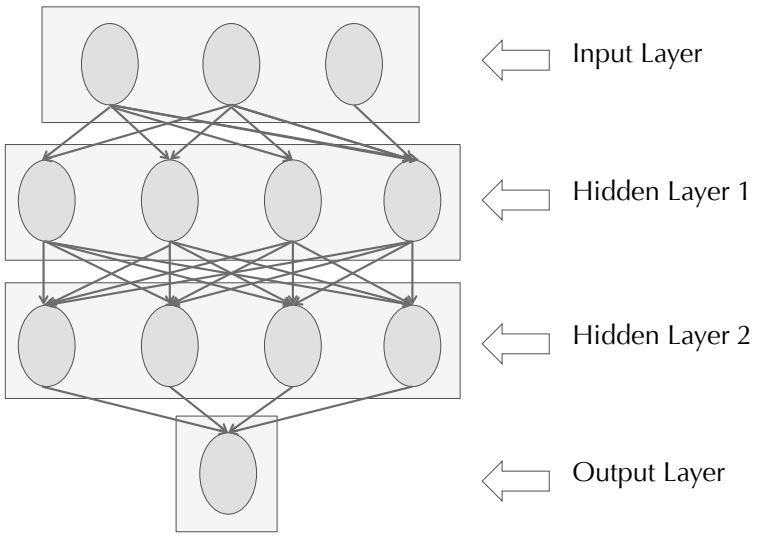

Fig. 6.2: Example of a fully connected feed-forward neural network.

A neural network is trained by minimizing a loss function over a training set, for example using a stochastic gradient descent [LeCun et al., 1998] or its variation,[1] i.e., we repeatedly compute an estimate of the error over the dataset, determine the "direction of the steepest descent" to be taken to reduce the error determined, and then moving the parameters in that direction.

Convolutional neural networks Another type of a feed-forward network is a convolutional neural network (CNN), i.e., a network which contains one or more convolutional layers. These networks, which were developed mainly for image recognition problems [LeCun and Bengio, 1995], are useful for classification tasks in which we expect to find strong local clues to class membership, yet at the same time we assume that these clues can appear in different places of the classified document. In an image classification, for example, the network can learn that a certain set of points and curves is a human face, regardless where on an image it appears [Krizhevsky et al., 2012]. Applied to text, a short phrase (i.e. word pattern) can help in classifying the topic of a document [Johnson and Zhang, 2014].

Another example is sentiment classification. We could use a continuous bag of words approach to represent the words in a sentence, and feed into a fully connected network, but in that case we will ignore the ordering of words completely, and assign the sentences "it was not good, it was actually quite bad" and "it was not bad, it was actually quite good" the exact same representation [Goldberg, 2016], which is suboptimal. We want to learn that certain sequences of words are more predictive, but not necessarily care about the

[1] A more recent alternative to the first-order methods such as SGD are the second-order optimization methods such as in [Martens, 2010].

position of the sequences. With the convolutional layers, such local sequences can be taken into account by the model, regardless where they appear in the document. In contrast to images, text applications usually operate on 1D convolution, i.e., operating with the patterns over an ordered sequence of words "in a row".

CNNs have shown promising results in several NLP tasks - after semantic role labeling [Collobert et al., 2011] also sentiment classification [Kim, 2014, Kalchbrenner et al., 2014], document topic classification [Johnson and Zhang, 2014], question answering [Dong et al., 2015b], paraphrase identification [Yin and Schütze, 2015] and event detection [Nguyen and Grishman, 2015].

In language-oriented classification tasks, the convolutional filter (i.e., a non-linear function that the CNN learns) is applied on a 1D sliding window of k words over the sentence. Each such window is transformed by the convolutional filters into a vector (each filter producing one scalar in it), capturing various properties of the words. A pooling operation is then applied to combine the vectors resulting from the different windows, usually by taking the maximum or the average of each vector dimension over different windows. This strategy enables to focus on the most important features in the sentence, no matter where in the sentence those are located. The resulting vector is then passed into the deeper layers of the network and used for prediction. Parameters of the convolutional filter functions are updated during the training, learning to emphasize the input properties important for the task. Note that both the fully connected and convolutional networks assume a fixed dimensional input - a common practice is to use the size of the longest document and complete the shorter ones with zeros.

Recurrent and recursive neural networks In contrast to images, language data typically comprise some notion of a sequence, e.g., of words, characters, or even sentences in a dialog. We mentioned above that CNNs are sensitive to word order in local patterns, however, the order of patterns that are not in a near proximity is not preserved. Recurrent neural networks (RNNs) are designed to represent input sequences of varying length, which they store in a vector of a fixed size, attempting to preserve useful information about the input structure. The simplest RNN was formulated by [Elman, 1990], but suffers from what is known as the vanishing gradient problem. In later steps in the sequence, the error gradients diminish during the training, and do not reach earlier input signals. Long-distance dependendies are therefore again not captured. The Long Short-Term Memory (LSTM) networks [Hochreiter and Schmidhuber, 1997] are designed with this issue in mind, addressing it by introducing so-called *memory cells* in a form of a vector with controlled access.[2] Such memory cells are capable of preserving the gradients over time. The controlled access is implemented by *memory gates*, conducting the operations of (wholly or partially) forgetting the content of the cells, or writing in (wholly or partially)

[2]There are also other recent approaches addressing vanishing gradients, such as the Gated Recurrent Units (GRU) [Cho et al., 2014], however, these are beyond the scope of this thesis.

the latest input. Recurrent models, especially LSTMs, produced competitive results e.g. for dependency parsing [Dyer et al., 2015], part of speech tagging [Huang et al., 2015], dialog response generation [Sordoni et al., 2015] and machine translation [Sutskever et al., 2014].

Recursive neural networks [Goller and Kuchler, 1996, Socher et al., 2010] are generalizations of recurrent networks to tree structures. Recursive models produced outstanding results for example for discourse parsing, semantic relation classification, and question answering. However, we do not deal with tree structures in this thesis due to the computational complexity on a document level.

6.1.3 Regularization of neural networks

Neural network models have many parameters, and overfitting can easily occur. Overfitting can be to some extent remedied by regularization. One of the efficient recent regularization methods, which we use in this thesis, is Dropout [Hinton et al., 2012b]. The idea of Dropout is to randomly remove units from the network in each training example, along with all their incoming and outgoing connections, preventing co-adaptation of learned weights. We use Dropout regularization in all our neural models, both here and in the MLP model in Chapter 4.

6.1.4 Word embeddings

When processing natural language, we need to represent the features of a text, such as words, part-of-speech tags, and other linguistic information, to provide a numerical input to the classification model. In the traditional supervised text classification approaches, such as those we used in the Chapter 3 and 5, each feature is represented as a unique dimension. In a neural network framework, in contrary, we represent each feature as a dense vector. This relates to the distributed representation theory of [Hinton, 1986]. By expressing the features as vectors (embeddings) which can be trained in the same way as any other parameters of the network during learning, we are able to capture different types of similarities between features by minimizing or maximizing the distances between each component of the multidimensional vectors.

Figure 6.3, produced by [Goldberg, 2016], demonstrates the difference between the traditional and the vectorial (embedding) approach to feature representation. In a sparse one-hot representation, features are independent from each other, i.e., the word dog is as dissimilar to a cat as it is to taxes. In dense vector representations, however, the information is shared between similar features by having similar vectors.

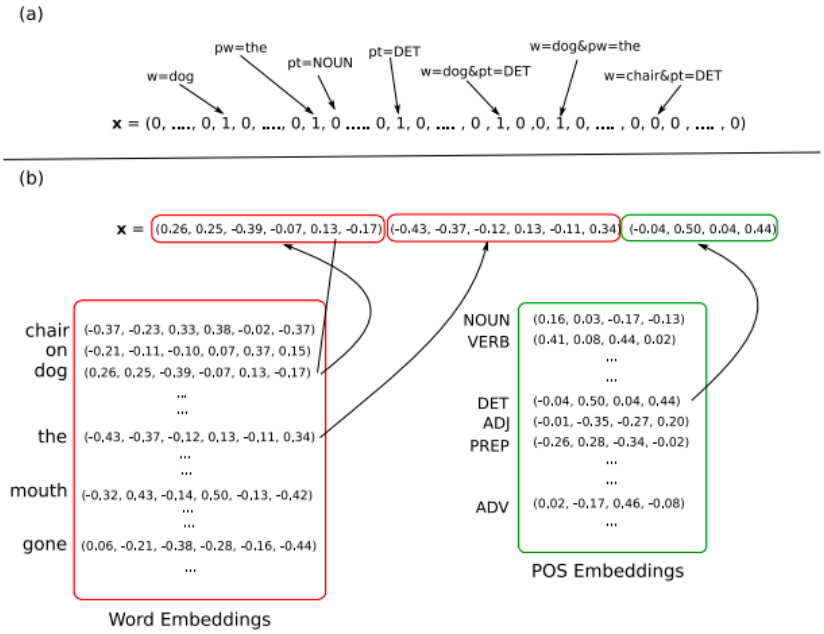

Fig. 6.3: Sparse and dense feature representations, encoding the following information: current word is “dog”; previous word is “the”; previous pos-tag is “DET”. Image taken from [Goldberg, 2016]

In theory, we could train feature representations (such as word embeddings) directly on our labeled classification dataset. However, such datasets are usually rather small, as obtaining the class labels is often expensive. Therefore it is common to train word embeddings on large amounts of unannotated data and then plug them into the neural network architecture. This enables to represent also words which may not appear in our training set. How the similarities between the vector dimensions for two different words are captured is defined by the training objective. Roughly speaking, we usually minimize the distance between words that appear frequently in the same contexts. However, the captured similarity should be useful for performing the intended classification task of the network, hence in some cases it may be useful to add specific objectives. Otherwise it can happen that for example the words *good* and *bad* are contextually very near, which can be an undesirable property for a sentiment classification task for example.

Common unsupervised word-embedding algorithms include word2vec [Mikolov et al., 2013a], GloVe [Pennington et al., 2014] and the Collobert and Weston [Collobert et al., 2011] embedding algorithm. These models are inspired by neural networks.

Approaches such as word2vec and GloVe are posed in a probabilistic setup, trying to model the conditional probability $P(w|c)$ of each word given its context. The most common approach is a sliding window approach, looking at a sequence of 2k + 1 words. Either a single task is created in which the goal is to predict the focus word based on all of the context

words (CBOW - continuous bag of words), or 2k distinct tasks are created, each pairing the focus word with a different context word (skip-gram model). In our experiments in Chapter 4, the skip-gram model outperformed the CBOW, which is consistent with previous literature [Pennington et al., 2014]. In current research, the dimensionality of word embedding vectors ranges between about 50 to a few hundreds, and, in some extreme cases, thousands.

6.1.5 Semantically enhanced word embeddings

The idea of combining distributional information with the expert knowledge is attractive and has been newly pursued in multiple directions. One of them is creating the word sense or synset embeddings rather than word embeddings [Iacobacci et al., 2015, Chen et al., 2014, Rothe and Schütze, 2015, Bovi et al., 2015]. While the authors demonstrate the utility of these embeddings in tasks such as word sense disambiguation and semantic similarity evaluation, the contribution of such vectors to downstream document classification problems can be challenging, given the fine granularity of the WordNet senses (cf. the discussion in [Navigli, 2009]). Although it, to our knowledge, has not been verified, we can expect similar issues as in our WSD experiments in Chapter 3. Supersenses have been shown to be better suited for carrying the relevant amount of semantic information, as we demonstrated in Chapter 5.

An alternative approach focuses on altering the objective of the learning mechanism for word embeddings to capture relational and similarity information from knowledge bases [Bordes et al., 2011, Bordes et al., 2012, Yu and Dredze, 2014, Bian et al., 2014, Faruqui and Dyer, 2014, Goikoetxea et al., 2015]. For example, [Faruqui et al., 2014] and [Jauhar et al., 2015] propose a method called *retrofitting* for refining vector space representations as a post-processing step, by encouraging words linked in WordNet to have similar vector representations. They show that such vectors achieve better results than the original word-based ones in tasks such as semantic similarity scoring, finding synonymy pairs and predicting sentiment polarity. [Ettinger et al., 2016] later point out that this approach does not require an ontology, and can be generalized to any graph defining word senses and relations between them, for example created using translations learned from parallel corpora. They show that it performs similarly to the WordNet-based retrofitting.

While, in principle, supersenses could be seen as a relation between a word and its hypernym, to our knowledge they have not been explicitly employed in these works. We are also not aware of any evaluation of these semantically enhanced embeddings in document classification tasks. Moreover, an important advantage of our explicit supersense embeddings compared to the retrained vectors is their direct interpretability. The fact that we have the resulting supersense vectors created in the same vector space as the word embeddings enables us to perform a straightforward qualitative analysis of our embedding

model, such as the visual exploration we demonstrated in Chapter 4. Subtle nuances, such as the meaning shift for the verbs *follow* and *spend* when changing the knowledge source, would be harder to notice if the learnt properties cannot be assessed explicitly.

6.2 Our experiments

In this section, we propose a deep learning approach, in which we process the original text in parallel to the supersense information. The model can then flexibly learn the usefulness of provided input. We demonstrate that the model extended with supersense embeddings outperforms the same model using only word embeddings, or using supersenses in a non-neural classification setting, in a range of classification tasks.

To summarize what we have learnt in the Background section and apply it to our experiment - the general steps for building an NLP classification system using neural networks are:

1. From the training and test documents, extract a set of core linguistic features (e.g. words) that are relevant for predicting the output class (e.g. sentiment).
2. Assign to each feature (e.g. word) its corresponding vector (e.g. word embedding)
3. Combine the vectors (usually by concatenation, but alternatively summation or multiplication) into one large input sequence of numeric values.
4. Feed this sequence into a neural network architecture.

The first step is usually addressed just by extracting words from a document, corresponding to a bag-of-words classification setup we discussed in previous chapters. However, while not widely used, even in a neural network setup the features can be in principle any linguistic annotations, e.g. part-of-speech tags, which we used in Chapter 4 for building our supersense tagging model with multi-layer perceptron. In this chapter, we use this model to annotate our documents with supersenses and use these supersense annotations as features in addition to words.

In the second step, we therefore assign embedding vectors to both the words and the supersenses. Since we have intentionally built in Chapter 4 the supersense embeddings in the same vector space as the original words, it provides the network with ideal conditions to learn the importance of the specific and generalized meaning of the document words in the semantic space. We additionally project the words into two more vector spaces capturing the relations between a word and its supersenses, as detailed in the following subsection.

In the third step, we concatenate each of these vector mappings of the document.

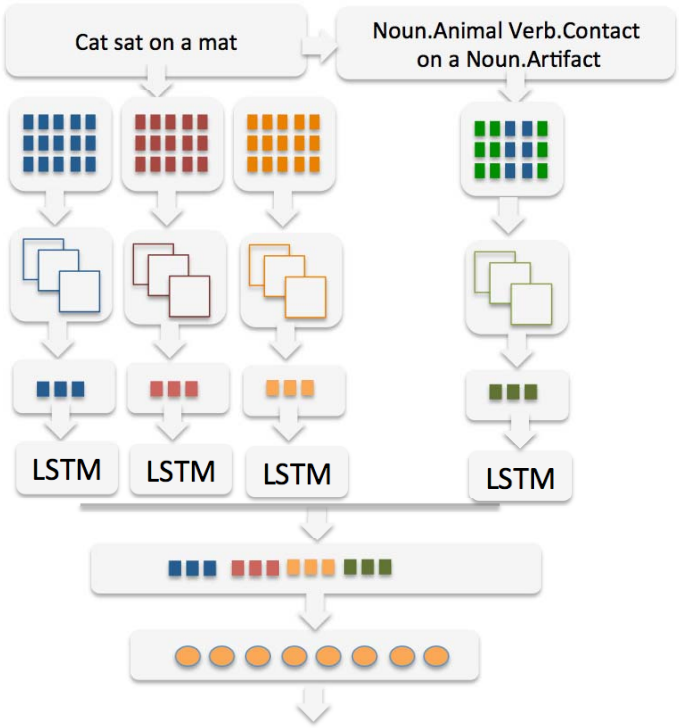

Fig. 6.4: Network architecture. Each of the four different embedding channels serves as input to its CNN layer, followed by an LSTM layer. Afterwards, the outputs are concatenated and fed into a dense layer.

In the fourth step, each of these mappings is fed into its corresponding subnetwork, which is merged with the other subnetworks deeper in the architecture, resulting in one class label output for the documents. The details of our architecture are explained below.

6.2.1 Network architecture

Both Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) [Hochreiter and Schmidhuber, 1997] are state-of-the-art machine learning models for a variety of text classification tasks [Kim, 2014, Li et al., 2015, Johnson and Zhang, 2014]. Recently, their combinations have been proposed, achieving an unprecedented performance [Sainath et al., 2015]. We extend the CNN-LSTM approach from the publicly available Keras framework[3], into which we incorporate the supersense information, based on the supersense embeddings we built in Chapter 4.

Figure 6.4 displays our network architecture. First, we use three channels of word embeddings on the plain textual input. The first channel are the 300-dimensional word embeddings obtained from our enriched Wikipedia corpus. The second embedding channel

[3] `https://github.com/fchollet/keras/blob/master/examples/imdb_cnn_lstm.py`

consists of 41-dimensional vectors capturing the cosine similarity of the word to each supersense embedding. The third channel contains the vector of relative frequencies of the word occurring in the enriched Wikipedia together with its supersense, i.e. providing the background supersense distribution for the word. Each of the document embeddings is then convoluted with the filter size of 3, followed by a pooling layer of size 2 and fed into a long-short-term-memory (LSTM) layer.

In parallel, we feed as input a processed document text, where the words are replaced by their predicted supersenses. Given that we have the Wikipedia-based supersense embeddings in the same vector space as the word embeddings, we can now proceed to creating the 300-dimensional embedding channel also for the supersense text. As in the plain text channels, we feed also these embeddings into the convolutional and LSTM layers in a similar fashion. Afterwards, we concatenate all LSTM outputs and feed them into a standard fully connected neural network layer, followed by the sigmoid for the binary output.

The following sections discuss our results on a range of classification tasks: extraversion prediction, subjectivity prediction, sentiment polarity classification and metaphor detection.

For each of the experiments, we compare the following settings:

- **WORD:** bag-of-words features with an SVM classifier (results from the previous section)
- **SUPER:** bag of words and supersense features annotated directly with a supersense tagger, with an SVM classifier (results from the previous section)
- **W-EMBED:** neural network architecture with word embeddings only (i.e., using only the first channel from figure 6.4)
- **SUPER-ONLY:** neural network architecture with word embeddings substituted by supersense embeddings for tagged nouns and verbs (i.e., using only the rightmost part of figure 6.4)
- **SUPER-EMBED:** The full neural network architecture with word embeddings and additional supersense embeddings on tagged supersenses (i.e., the entire architecture displayed on figure 6.4)
- **BASELINE:** majority class baseline
- **SoA:** state-of-the art results, i.e., best reported performance in other papers known to us

6.2.2 Datasets

We conduct experiments across 4 of the datasets from the previous section: Pennebaker's personality essays (ESSAYS), Kosinski's Facebook myPersonality data (FACEBOOK), Pang's Movie Review sentiment dataset (SENTIMENT) and Pang's Subjectivity dataset (SUBJECTIVITY). In addition, we use a dataset of 985 literal and 985 methaphorical adjective-noun pairs[4], published by [Tsvetkov et al., 2013]. For example, the phrase *broken promise* is an adjective-noun metaphor, where attributes from a concrete domain (associated with the concrete word *broken*) are transferred to a more abstract domain, which is represented by the relatively abstract word *promise*. Our motivation for including this dataset is that super-senses have recently been shown to provide improvements in metaphor prediction tasks [Gershman et al., 2014], as they hold the information of coarse semantic concepts.

6.2.3 Related work

We have discussed the related work on the personality datasets in the previous chapter, and we are not aware of any deep learning experiments on this data.

For sentiment classification, the Movie Review dataset is a standard machine learning benchmark task. [Socher et al., 2011] address this task with recursive autoencoders and Wikipedia-based word embeddings, later improving their score using recursive neural network with parse trees [Socher et al., 2012]. Competitive results were achieved also by a sentiment-analysis-specific parser [Dong et al., 2015a], with a fast dropout logistic regression [Wang and Manning, 2013], and with convolutional neural networks, with which [Kim, 2014] achieved an accuracy of 81.4, using Google-pretrained word2vec word embeddings. [5] We use this result as our state-of-the-art (SoA) benchmark.

In subjectivity classification, [Wang and Manning, 2013] recently improve the original accuracy of [Pang and Lee, 2004] to 93.6 by using fast approximation to dropout regularization, and [Kim, 2014] reaches 93.4 using convolutional neural networks, similarly to the sentiment dataset. [Zhang et al., 2016] achieves 93.9% accuracy (SoA) using an enhancement of a convolutional neural network architecture with word embeddings.

The task of discriminating literal and metaphoric adjective-noun expressions has been previously explored by [Turney et al., 2011]. They report an accuracy of 79% on a small dataset rated by five annotators. [Tsvetkov et al., 2013] pursue this work further by constructing and publishing a dataset of 985 literal and 985 methaphorical adjective-noun pairs and classifying them. [Gershman et al., 2014] further expand on this work using 64-dimensional vector-space word representations constructed by [Faruqui and Dyer,

[4] http://www.cs.cmu.edu/~ytsvetko/metaphor/datasets.zip
[5] https://code.google.com/archive/p/word2vec/

2014] for classification. They report a state-of-the-art F-score of 85% (which we use as SoA in our result table) with random decision forests, including also abstractness and imageability features [Wilson, 1988] and supersenses from WordNet, assigned without sense disambiguation (using a latent supersense assigned as an average of the supersenses of all WordNet senses of a word).

6.2.4 Results and error analysis

Below, we present our results for the five binary classification tasks: extraversion prediction (on two datasets), sentiment polarity classification, subjectivity classification and metaphor identification. We exclude the gender prediction from our tasks, since the semantic features turned out to be less predictive than the stylistic ones in our experiments in the previous chapter, hence the task is less relevant for the analysis of the supersense embeddings.

Extraversion classification

Table 6.1 displays our experimental results on the ESSAYS and FACEBOOK datasets, compared to our results in the previous chapter. On the ESSAYS data, our deep learning architecture did not outperform the results of the support vector machines (WORD, SUPER) in any of the settings. We hypothesize that this is caused by the largely varying length of the input documents. Since the convolutional neural networks require a fixed size vector input, the length of our feature vector (concatenation of all the word, resp. supersense embeddings) is given by the longest document in the data. Input vectors from documents containing fewer words are completed by zeros (an approach known as padding), a simplification which may be suboptimal. Furthermore, as the convolutional networks are suitable for finding local relations between features, an advantage of this method can be more beneficial for short documents than the very long ones, such as the streams of consciousness. However, we observe that the usage of word and supersense embeddings together (SUPER-EMBED) yields superior results compared to using word embeddings only.

Our hypothesis about the long document issue is supported by the results on the FACEBOOK dataset, where the improvement achieved by using the deep learning architecture is much higher. In this case, we classify the extraversion on the level of individual Facebook status update (i.e., usually one or several sentences), and determine the final class for the user by the majority class of her status labels (i.e., over 50% extravert- or introvert-classified messages). The approach of using both word and supersense embeddings combined (SUPER-EMBED) outperforms using only word embeddings (W-EMBED, left side of the architecture on figure 6.4) and using only supersense embeddings with the words replaced by their supersenses (SUPER-ONLY, right side of the architecture on figure 6.4). Supersense embeddings alone perform worse than word embeddings alone. This is because some of the

information is lost by the high abstraction - for example, the verbs *love* and *hate* are both replaced by the supersense *verb.emotion*, thus having the same vector. However, adding the supersense information on top of the word embeddings helps the classifier to learn general patterns in addition to the specific ones. Whenever there is an unknown expression in the test set, but the same supersense has been seen in the training set in the same context, the classifier can default to the more generic solution rather than taking an arbitrary decision based on the unknown word.

Sentiment Polarity and Subjectivity Classification

Table 6.2 displays our results for a 10-fold cross-validation on the sentiment and subjectivity dataset. As we intended to reach comparable results to the state of the art, published in machine learning conferences, 10% of the data was withheld for parameter tuning of the network in both cases. For the same reason, we also compare only the accuracy score, as customary in previous papers. The line *W-EMBED* displays the performance using only the leftmost part of our architecture, i.e. only the text input with our Wikipedia-based word embeddings. The line *SUPER-EMBED* shows the result of using the full supersense architecture. *SUPER-ONLY* shows the performance of the rightmost part of the architecture only. As it can be seen from the table, combining the supersense embeddings with the word embeddings improves the accuracy by about 2% over word embeddings. The *SUPER-EMBED* and *W-EMBED* systems are significantly different ($p < 0.01$), using the McNemar's test, in both cases - sentiment polarity and subjectivity prediction.

Personality essays			**Facebook personality**		
Dataset/setup	Accuracy	F-score	Dataset/setup	Accuracy	F-score
WORD	0.546	0.548	WORD	0.567	0.576
SUPER	**0.649***	**0.587**	SUPER	0.621*	0.617
W-EMBED	0.546	0.548	W-EMBED	0.621*	0.619
SUPER-EMBED	0.557	0.562	SUPER-EMBED	**0.672***	**0.660**
SUPER-ONLY	0.523*	0.520	SUPER-ONLY	0.613	0.611
BASELINE	0.517	0.498	BASELINE	0.616	0.472
SoA [Mohammad, Kiritchenko, 2013]	-	0.563	SoA [Verhoeven et al., 2013]	-	**0.700**

Tab. 6.1: Extraversion classification performance. Supersense features outperform bag-of-word configurations for both SVM and neural network settings. The deep learning model (SUPER-EMBED) performs better on the Facebook dataset, while SVM achieves higher scores on the Essays. Configurations with a statistically significant difference (McNemar's test, $p < 0.05$) from the ngram (WORD) setup are denoted with a star(*) on the accuracy column. There is no human upper bound as the gold labels are psychological self-assessments.

Table 6.3 shows an example of positive and negative reviews which were consistently (5x in repeated experiments with different random seeds) classified incorrectly with word embeddings and classified correctly with supersense embeddings. It appears that the supersense generalization is helpful in cases, where the authors of the movie reviews attempt to use unusual expressions to describe their feelings (e.g., *a rambling and incoherent manifesto to vagueness...*). While the wit of such expressions is often lost for the benefit of generalization, the conceptual information seems to cover the lexical gap. Some improvements also appear to be a result of replacing proper names by NOUN.PERSON. Additionally, the supersense findings from the previous chapter apply, e.g., noun concepts such as GROUP, LOCATION, TIME and PERSON occur somewhat more frequently in positive reviews while certain verb supersenses such as PERCEPTION, SOCIAL are present more often in the negative ones. Similarly, subjective sentences contain more supersenses of PERCEPTION, COMMUNICATION and ATTRIBUTE, while objective ones more frequently feature the supersenses POSSESSION and PERSON.

Metaphor Identification

The metaphor identification task consists of word pairs only, an adjective and a noun. The task is to determine if the expression is figurative or not. Since this setup is simpler than the sentence classification tasks, we use only a subset of our architecture, specifically our word embeddings, supersense similarity vectors and supersense frequency vectors. Supersense similarity vectors express the cosine similarity of the examined noun or adjective to each

Dataset/setup	Accuracy	Dataset/setup	Accuracy
Movie review sentiment		**Sentence subjectivity**	
WORD	0.744	WORD	0.910
SENSE-SUPER	0.751*	SENSE-SUPER	0.925*
SUPER	0.761*	SUPER	0.926*
W-EMBED	0.794*	W-EMBED	0.921*
SUPER-EMBED	**0.817***	SUPER-EMBED	**0.939***
SUPER-ONLY	0.767*	SUPER-ONLY	0.879*
BASELINE	0.500	BASELINE	0.500
[Socher et al., 2011]	0.777	[Pang and Lee, 2004]	0.900
[Socher et al., 2012]	0.790	[Pang and Lee, 2004]	0.920
[Wang and Manning, 2013]	0.791	[Kim, 2014]	0.934
[Dong et al., 2015a]	0.795	[Wang and Manning, 2013]	0.936
[Kim, 2014] (SoA)	0.815	[Zhang et al., 2016](SoA)	0.939

Tab. 6.2: 10-fold cross-validation accuracy of our system and as reported in previous work for the sentiment classification task on [Pang and Lee, 2005] movie review data. Configurations with a statistically significant difference (McNemar's test, $p < 0.05$) from the ngram (WORD) setup are denoted with a star(*) on the accuracy column. Also the *SUPER-EMBED* and *W-EMBED* systems are significantly different ($p < 0.01$). The standard error of the SUPER-EMBED accuracy measurements is approximately 0.004 in both tasks.

Positive reviews	
Text	**Supersenses**
beating the austin powers film at their own game ,	verb.stative the noun.location noun.cognition noun.artifact at their own , noun.communication ,
this blaxploitation spoof downplays the raunch in favor	this noun.act noun.communication verb.stative the noun.cognition in noun.communication
of gags that rely on the strength of their own cleverness	of that verb.cognition on the noun.cognition of their own noun.cognition
as oppose to the extent of their outrageousness.	as verb.communication to the noun.event of their noun.attribute .
there is problem with this film that	there verb.stative noun.cognition with this noun.communication that
even 3 oscar winner ca n't overcome ,	even 3 noun.event noun.person ca n't verb.emotion ,
but it 's a nice girl-buddy movie	but it verb.stative a nice girl-buddy noun.communication
once it get rock-n-rolling .	once it verb.stative rock-n-rolling
godard 's ode to tackle life 's wonderment is a	noun.person noun.communication to verb.stative noun.cognition 's noun.cognition verb.stative
rambling and incoherent manifesto about the vagueness of topical	a rambling and incoherent noun.communication about the noun.attribute of topical
excess . in praise of love remain a ponderous and pretentious	excess . in noun.cognition of noun.cognition verb.stative a ponderous and pretentious
endeavor that 's unfocused and tediously exasperating .	nounact that verbstative unfocused and tediously exasperating
Negative reviews	
Text	**Supersenses**
the action scene has all the suspense of a 20-car pileup ,	the noun.act noun.location verb.stative all the noun.cognition of a 20-car noun.cognition ,
while the plot hole is big enough for a train car to drive	while the noun.location verb.stative big enough for a noun.artifact noun.artifact to verb.motion
through – if kaos have n't blow them all up .	through – if noun.person have n't verb.communication them all up .
the scriptwriter is no less a menace to society than the film 's character .	the noun.person verb.stative no less noun.state to noun.group than the noun.communication noun.person .
a very slow , uneventful ride around a pretty tattered old carousel .	a very slow , uneventful noun.act around a pretty tattered old noun.artifact .
the milieu is wholly unconvincing . . . and the histrionics reach a truly annoying pitch .	the noun.cognition verb.stative wholly unconvincing and the noun.communication verb.stative a truly annoying noun.attribute .

Tab. 6.3: Example of documents classified incorrectly with word embeddings and correctly with word and supersense embeddings on [Pang and Lee, 2005] movie review data

of the 42 supersense embedding vectors, and the supersense frequency vectors express how often this word has been annotated in the Wikipedia corpus with each supersense, if at all. Since there are only two words in each document, we leave out the LSTM layer. We merge the similarity and frequency layers by multiplication (both vectors are of the same length) and concatenate the result to the word embedding convolution, feeding the output of the concatenation directly to the dense layer. Table 6.4 shows our results on a provided test set. Our system using supersense features is significantly better than the one using word embeddings only. This corresponds to the previous findings of [Gershman et al., 2014], who points out that "supersenses are particularly attractive for metaphor detection, since concept mapping is a process in which metaphors are born". They illustrate this on an example of "(a car) drinks gasoline", where mapping to supersenses yields a pair <*verb.consumption, noun.substance*>, contrasted with <*verb.consumption, noun.food*> for "(a person) drinks juice".

System	[Gershman et al., 2014]	WORDS	SUPER
F1-score on test-set	0.850	0.819	**0.872***

Tab. 6.4: F1-score on a provided test set for the adjective-noun metaphor prediction task [Gershman et al., 2014]. WORDS: word embeddings only, SUPER: multi-channel word embeddings with the supersense similarity and frequency vectors added. Based on McNemar's test, there is a significant difference ($p < 0.01$) between our WORDS and SUPER systems.

6.2.5 Result summary

In our experiments, we manifested that the supersense enrichment can lead to a significant improvement in a range of downstream classification tasks, using our embeddings in a neural network model based on a combination of convolutional and recurrent neural networks. Based on our experiments, the improvements are higher on shorter documents, such as sentences or social media posts. While the word embedding vectors already capture certain type of contextual similarity between words, supersenses help the classifier to deal with rare words, which are either not present in the underlying word embedding corpus at all, or may be positioned imprecisely in the vector space due to their infrequent occurrence. Additionally, the richer feature space is capturing more abstract relations. This way, also a deep learning system is able to learn more robust patterns and can be potentially trained on a smaller data set than if only word embeddings are used.

The benefits of supersenses in text classification also conceptually overlap with the idea of *zero-shot learning* [Larochelle et al., 2008, Palatucci et al., 2009, Socher et al., 2013a], in a way that abstracting unseen lexemes to supersenses allows us to build a model for them by projecting/recycling the knowledge from seen lexemes with the same supersenses. Zero-shot learning is an extreme form of transfer learning, which attempts to assign class labels at test time without seeing any examples of it at training time [Goodfellow et al., 2016], i.e., where some of the possible values for the class label have been omitted from the training examples. This learning is only possible when additional information has been exploited during training. For example, the classifier might be able to recognize an image of a cat, if it obtained an unlabeled textual description that cats have four legs, pointed ears and a tail [Palatucci et al., 2009].

Zero-shot learning requires to be represented in a way that allows some sort of generalization. [Palatucci et al., 2009] use a semantic feature space containing answers to questions such as *Does it stand on two legs? Can you hold it?* to classify fMRI scans of people thinking about certain words. [Socher et al., 2013a] show that the language feature representations for the zero-shot classes of images can be learned from unsupervised and unaligned corpora as vector representations (embeddings) instead of manually defining semantic or visual attributes. Since they use a set of word embeddings to represent each image class, our supersense embeddings could be easily used to enrich such representations with an additional, possibly more robust semantic information.

6.3 Chapter summary

In this chapter, we provided an overview of neural network approaches for text classification and discussed the most common **neural network architectures** and their typical

use cases. We reviewed the concept of **word embeddings** in more detail from the neural network perspective and explained the differences between vectorial and traditional features. We proposed a **deep learning approach**, combining the most recent neural network architectures for NLP (convolutional and recurrent networks), which we enrich with the supersense embeddings. We conducted a range of document classification experiments, demonstrating that the additional semantic information from supersenses improves the classification performance also in deep learning settings, and that the performance gains with our network architecture are higher for shorter documents.

A comprehensive overview of our results obtained in this and the previous chapter is presented at Figure 6.5. We can see that adding supersense features always improves the results over using word features only. We also find that the classification with neural networks is powerful and outperforms SVM in most cases, except for the personality essays dataset, where the documents are very long and their length largely varies (see the corpus statistics in Table 3.2 in Chapter 3). Besides quantitative results, we explored the classification outcomes qualitatively and found that supersenses help the model to generalize over rare expressions, which is a promising strategy for training deep learning models on smaller datasets than currently required. The main contributions of this chapter were published in [Flekova and Gurevych, 2016].

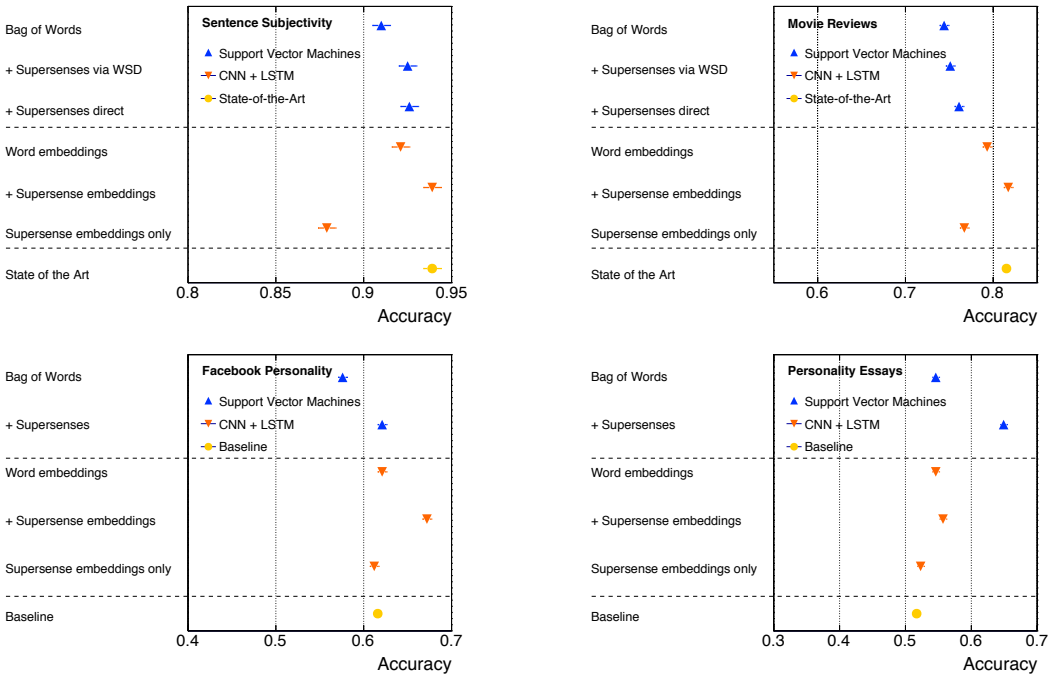

Fig. 6.5: Overview of our text classification results obtained in this and the previous chapter, i.e., with and without supersenses, using either SVM or CNN+LSTM neural network.

7 Challenges of Generating Labeled Data

> *The saddest aspect of life right now is that science gathers knowledge faster than society gathers wisdom.*
>
> **— Isaac Asimov**

An important prerequisite to conduct any supervised text classification experiment is defining the ground-truth data, which will be used for the training and evaluation. Often, we need to construct new data for our purpose by conducting an annotation study. In some other cases, data sets for a given problem might be readily available. However, even then it is important to understand how the class labels were obtained and what are the limitations of the approach selected.

In this chapter, we discuss several factors which may influence the quality of obtained annotations - Figure 7.1 highlights in blue why this is crucial for the entire text classification process. We explicitly examine the influence of the formulation of the task, the annotator's prior assumptions about the data and the personal settings of an annotator. For annotations which imply making judgments about other people, we further explore which language features may influence annotator's perception and what consequences does that have for the end system.

We focus on a class of crowdsourcing tasks called consensus tasks [Kamar et al., 2012]. The goal of a consensus task is to identify a previously unknown correct answer via the aggregation of predictions provided by workers. Consensus tasks are common in crowdsourcing and provide workers with labeling challenges.

7.1 Background and related work

The rise of Amazon Mechanical Turk and other crowdsourcing platforms opened the door to solving human intelligence tasks at scale. Thanks to the access to a large base of human annotators, the process of obtaining ground truth data for the tasks difficult for machines (such as object recognition, assessment of emotions conveyed, or disambiguation of senses)

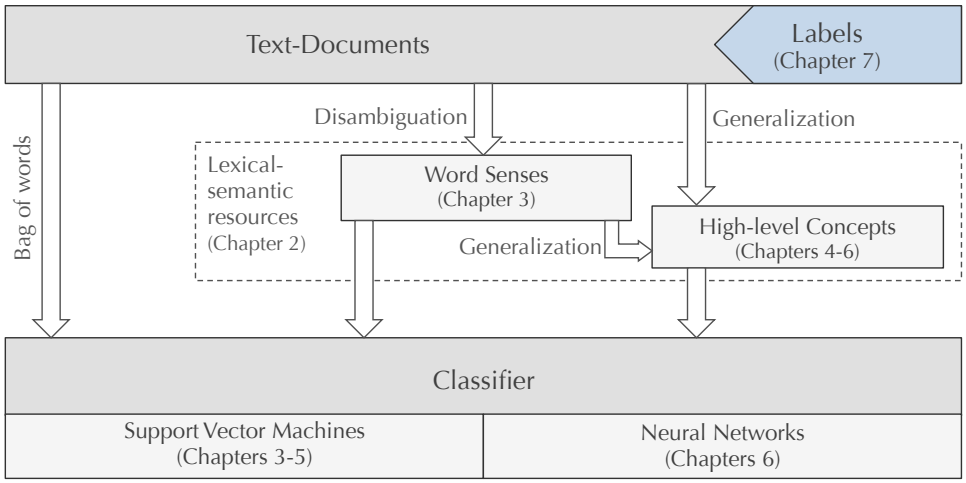

Fig. 7.1: The concept explored in this chapter (highlighted blue).

becomes faster, cheaper, and the base of annotators more diverse than for example in a university environment. At the same time, the decisions of individual annotators are noisy [Ipeirotis et al., 2010]. Researchers usually expect that the aggregation of a large numbers of annotations accounts for the individual noisy performance [Bachrach et al., 2012, Snow et al., 2008]. This expectation is formally supported by the Condorcet Jury Theorem [Condorcet, 1785, Ladha, 1995], under the condition that each of the individual annotators can perform better than chance.

Deriving ground truth task labels by aggregating a set of individual annotations in such a way that the quality is optimized and the biases mitigated, can be non-trivial. Researchers recently proposed multiple approaches to represent the relationship among task labels, worker annotations, and worker ability [Raykar et al., 2010, Welinder et al., 2010, Kamar et al., 2012], and the contributions of these aspects to task difficulty [Raykar et al., 2010, Whitehill et al., 2009]. Several works explicitly model worker bias using confusion matrices to capture the types of mistakes made by individual annotators [Zhou et al., 2012, Simpson et al., 2013]. However, the relationship between task characteristics and the errors of annotators is not represented in these models. Creating this link has been addressed by adding task difficulty as a latent variable [Whitehill et al., 2009, Bachrach et al., 2012, Simpson et al., 2013]. Despite these efforts, in order to learn about a task-dependent worker bias, i.e., to infer not only whether a worker is likely to make a mistake, but which mistake, we need to capture more than the relationship between individual bias and task difficulty.

While there has been a large body of work focusing on computational approaches to remedy imperfect human annotations, the discussion on the causes of these imperfections and their systematic characterization is scarce. Based on our experiments, we suggest that the issues

in human annotation can be grouped to the following areas, discussed in detail in the following sections:

- Annotation task formulation
- Annotator's assumptions about the data
- Annotator's personal settings

Annotation task formulation plays an important role in the quality of results obtained. The formulation of annotation guidelines, the questions asked, the type of answers provided or the annotation environment can all guide the annotators to prefer a certain type of answer. While these effects have not been central to the crowdsourcing research in natural language processing, there is a large body of work in human sciences exploring them. [Reder and Ritter, 1992] report that the terms used in a question influence the feeling of knowing the answer more than the actual user's ability to solve the problem. [Semin et al., 1995] show that when a question is formulated with an action verb, the answers focus more on the subjects themselves than when using a state verb. [Waterman et al., 2004] demonstrate that users are more likely to admit they don't know an answer if an open question is asked, in contrast to a yes/no question. [Tourangeau and Smith, 1996] found that people are likely to report a different number of sexual partners on average, if the options are more fine-grained towards the lower or the higher side of the scale. Finally, [Bowling, 2005] show that the mode of questionnaire administration, such as its length, pace, or the order of questions, can have serious effects on data quality.

Annotator's assumptions about the data may lead to a systematic bias in the annotations. In some cases, annotators make prior assumptions about the data distribution. For example, [Nguyen et al., 2014] show that when annotating age on Twitter, the workers systematically underestimate the age of adults over 30 years, rating them as younger. In other cases, workers adjust their label distribution to the set of instances observed. For example, [Zhuang and Young, 2015] demonstrate that for the task of identifying inappropriate comments in social media, the annotators label the same comment as inappropriate when presented in a batch of appropriate comments, and as appropriate when presented in a batch of inappropriate comments. These assumption about the appropriate labels can therefore change as a result of previously observed instances. [Marcus et al., 1993] pointed out that when two annotators tagged for POS, the interannotator disagreement rate was 7.2%, while if this was changed to a task of correcting the output of an automatic tagger, the disagreement rate dropped to 4.1%. This also opens up the question of annotator training. [Ipeirotis et al., 2010] state that the annotation quality improved when each worker labeled at least 20 to 30 instances. [Snow et al., 2008] investigated the crowdsourcing of five natural language processing tasks - affect recognition, word similarity, recognizing textual

entailment, event temporal ordering, and word sense disambiguation - and found that individual expert annotators were better than individual non-experts. However, a majority vote on 2-9 non-experts resulted in a similar performance on each task. [Alonso and Romeo, 2014] explored the biases in expert and crowdsourced annotations on the word sense disambiguation task, with findings contradicting the ones of [Snow et al., 2008]. They report that non-expert annotators manifest a behavior that makes them choose the easiest option as a default, showing a bias against the unusual sense that is stronger than in the data annotated by experts.

Annotator's personal settings refers to the personal characteristics of an annotator, which are not necessarily explicitly related to the task expertise. This can be a mixture of psychological and demographic factors, such as the gender, nationality, personality, and cultural background, forming a unique set of annotator's beliefs and opinions. For example, previous research in psychology has shown that people with highly developed actively open-minded thinking [Baron, 1991] are better at a variety of mental tasks, such as estimating relative amounts [Haran et al., 2013] or distinguishing between good and bad arguments [Stanovich and West, 1997]. [Kazai et al., 2012] analyzed multiple user traits in the context of crowdsourcing relevance labels, finding that geolocation of the workers plays a major role in performance with smaller effects for gender and age. [Kazai et al., 2011] examined the personality of crowdsourcing workers with regards to the task performance and found a strong correlation between worker's openness and annotation accuracy. [Li et al., 2014] propose a framework to target crowdsourcing tasks to specific annotator groups in order to improve the annotation quality on that task. The targeted crowd is defined by the worker characteristics such as nationality, education level, gender, and personality test score.

7.2 Experiments

Based on our typology of the factors which may influence the quality of obtained annotations, the experimental section of this chapter is structured as follows:

In Section 7.2.1, we report the results of our experiments with different task settings for personality and sentiment annotations. In Section 7.2.2, we investigate the age annotation bias reported by [Nguyen et al., 2014] on our own crowdsourcing experiment. We further examine the qualitative difference between annotators labeling a small or a large number of instances, and we investigate the differences between expert and non-expert annotators. In Section 7.2.3 we analyze, for the first time, the impact of annotator's personality, age, and gender on the quality of their judgment of authors' demographics from written text. Additionally, we examine the relation between annotator's demographics, confidence, and accuracy, and we explore in detail the lexical choices which trigger stereotypical conclusions contrasting the ground truth.

7.2.1 Effects of the task formulation

In this section, we conduct two experiments. The first one focuses on annotating personality traits of fictional characters through two different questionnaires. Our hypothesis is that the task formulation influences the shape of distribution of the crowdsourced ratings. The second experiment is focused on the annotation of sentiment polarity of word bigrams. Our hypothesis is that the number of answer options provided in the questionnaire influences the disagreement patterns.

Personality assessment task

In this experiment, our goal was to obtain annotations for perceived personality of fictional characters, with the aim of classifying personality from text automatically later on. Our study here hence focuses on annotating the Extraversion scale in two different settings.

The first setting, in which we benefit from the popularity of the MBTI scheme, was described in Chapter 5. Online users (book fans) classify the book characters along the four MBTI axes, of which one is the Extraversion/Introversion. In this settings, user makes a simple binary vote, in which she marks if she perceives the character as introverted (+1) or extraverted (-1). The final score for the character is obtained by averaging the score of all users. The percentage agreement of voters in our data is 84.9%.

In the second setting, we followed the Five Factor Model methodology more closely. In the model of [Costa and McCrae, 2008], the extraversion is characterized by its six underlying facets - gregariousness, assertiveness, activity, excitement seeking, warmth and positive emotion. Each of these facets is measured by a positive and a negative question, resulting in a 60-item questionnaire. The self-assessment questions are behavioral, such as "*When confronted with ..., I prefer to...*" and the users are asked to express their agreement or disagreement with the statement. Since the fictional characters obviously cannot fill these questionnaires by themselves, we took inspiration in these forms to paraphrase the questions for each trait to be answered by the fans. Each question has three possible answers which count as -1, 0 or 1 towards the overall score. The weight of each question is determined by the correlation of the trait facet to the overall trait strength, as reported by [Costa and McCrae, 2008]. The percentage agreement of voters in our data is 75.1%

For our experiments in this section, the important finding is the following: when we compare the results obtained for the same characters by both methods, the distribution of the scores differs. This is illustrated at Figures 7.2 and 7.3. We selected those fictional

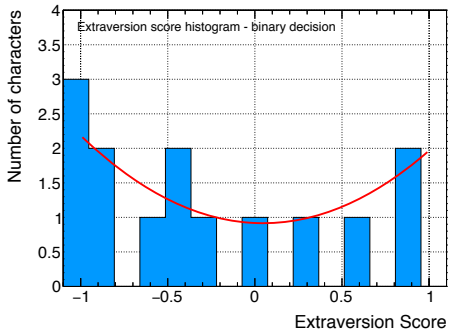

Fig. 7.2: Extraversion Assessment: Dichotomic question

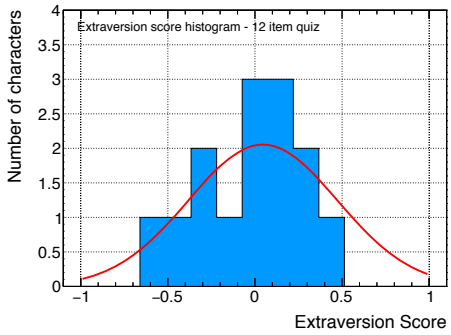

Fig. 7.3: Extraversion Assessment: Behavioral Quiz

characters with at least 30 votes in the direct MBTI rating, and at least 3 votes in the detailed 60-item test. Since extraversion is normally distributed in the population [Goldberg, 1990], we would expect that users, even when asked a dichotomic question directly, will disagree on the characters that are not clearly extraverted or introverted, drawing the score closer towards 0. However, this is not the case. While in the 60-item questionnaire the assigned extraversion scores result to a normal distribution of the trait among characters (Figure 7.2), with the majority of them being close to the mean, averaging the direct question results to a bipolar distribution (Figure 7.3). This can possibly be due to the fact that the users are reluctant to express their opinion when uncertain - a phenomenon observed also in quality reviews of products or services. By circumventing this problem through a set of indirect questions, we can obtain a more representative result.

Sentiment analysis

The goal of this annotation experiment was to obtain human sentiment labels for emotion-laden bigrams described in chapter 3 of this thesis.

Our human annotation experiment served as a validation for our automated approach. The crowdsourcing workers were presented a list of two times 100 bigrams based on two word polarity lexicons (the MPQA lexicon [Wilson et al., 2005] and the HL lexicon [Hu and Liu, 2004]) and had to answer the question “Which polarity does this bigram have?”. In the first setting, they were given only the *positive* and *negative* option as answer, in the second setting they could choose between *positive*, *negative* and also *neutral*. Each bigram is rated by three annotators, and the majority vote is selected.

The results of both settings are presented in Table 7.1. We can see that when the annotators are presented with only two options (Table 7.1 left), the accuracy - evaluated as a match with the automatically assigned label - is relatively high, but the inter-annotator agreement

Two options	HL		MPQA		Three options	HL			MPQA		
	Pos.	Neg.	Pos.	Neg.		Pos.	Neu.	Neg.	Pos.	Neu.	Neg.
Pos.	38	11	39	6	Pos.	30	10	9	21	24	3
Neg.	14	37	9	46	Neg.	11	10	30	5	18	25

Tab. 7.1: Confusion matrix for the majority vote of word polarity by three annotators. In the first case (left), the annotators were given two options to choose from, in the second case (right) annotators had three options to choose from, for the same words. Columns show annotators majority vote while rows list the label assigned by our algorithm in Chapter 3.

is low. In the setting where the annotators are given an additional, neutral option (Table 7.1 right), the accuracy drops (since our automated labels only had the values *positive* and *negative*), but the inter-annotator agreement increases. This provides a new insight into the task, since indeed some of the retrieved bigrams are originating from a strongly negative lexicon word (such as *limited*) but result in a neutral expression, identified by our algorithm as slightly positive (such as *limited edition*).

7.2.2 Effects of the annotator's assumptions about the data

In this section, we present three experiments - crowdsourcing the annotations of age estimates of Twitter users based on their Tweets, an analysis of change of performance in crowdsourced estimates of demographics of Twitter users depending on the number of items annotated, and a comparison of expert and non-expert ratings of Wikipedia quality. In the first experiment, we intend to reproduce the findings reported in the previous work of [Nguyen et al., 2014], who found that people constantly underestimate age of social media users. In the second experiment, we examine how the annotators' performance changes with the number of items annotated. In the third experiment, we compare the ratings of quality of Wikipedia articles by people self-reporting themselves as experts in the domain and by general public.

Perception of authors' age on Twitter

In this experiment, we created an age annotation task on Amazon Mechanical Turk. Each HIT (question for an annotator) consisted of 20 tweets sampled from a pool of 100 tweets posted by one user over the past six months. The annotators tried to estimate the user's age, stating the confidence of their rating on a scale from 1 to 5. Each user was assessed independently by 9 annotators. For quality control, we used a set of HITs where the age was explicitly stated within the top 10 tweets displayed to the annotator. The control HIT appeared 10% of the time and an annotator missing the correct answer twice was excluded from annotation and all his HITs invalidated. A total of 28 annotators were banned from

the study. Further, we limited the annotator location to the US, and they had to spend at least 10 seconds on each HIT before they were allowed to submit their guess.

The dataset for evaluating the age annotations (and from which we sampled the tweets) is obtained by identifying $4,279$ users that were the target of a tweet such as 'Happy X birthday to @USERNAME'. For our experiment, we sampled 1000 users from each of the five culturally meaningful age groups: < 18, $18-22$, $23-30$, $31-40$, $41+$.

Figure 7.4 shows a scatter plot comparing real and predicted age together with a non-linear fit of the data. From this figure, we observe that annotators under-predict age, especially for older users. The correlation of MAE with real age is very high ($r = 0.824$) and the residuals are not normally distributed. We notice this trend across all workers, having similar prior beliefs about the user distribution on Twitter. This finding is consisted with the previous Twitter annotation experiment conducted by [Nguyen et al., 2014].

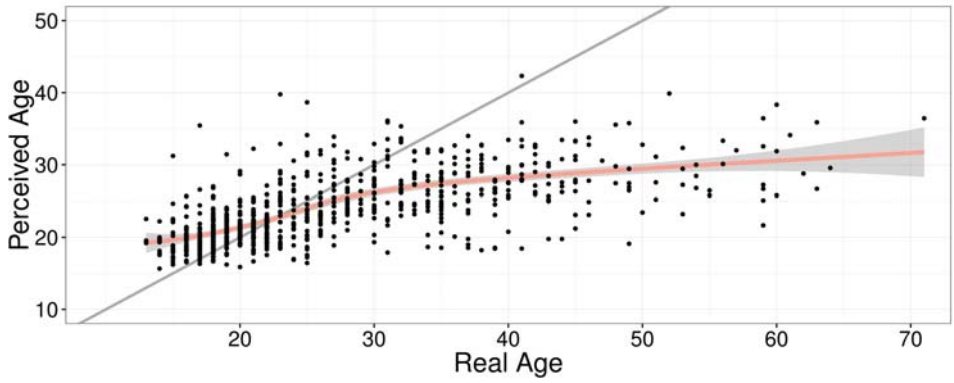

Fig. 7.4: Real age predictions compared to average predicted age. The line shows a LOESS fit.

Change of performance with items annotated

[Ipeirotis et al., 2010] have shown that the number of annotations per worker in crowdsourcing tasks is distributed according to the power law, meaning that most workers provide only a small number of annotations. They report that with the increasing number of annotations performed per annotator, her annotation accuracy increases. We observe the same trend in our gender prediction experiments, as displayed on Figure 7.5. There is a gentle, but steady increase in annotator's performance, linearly dependent on the number of HITs they completed.

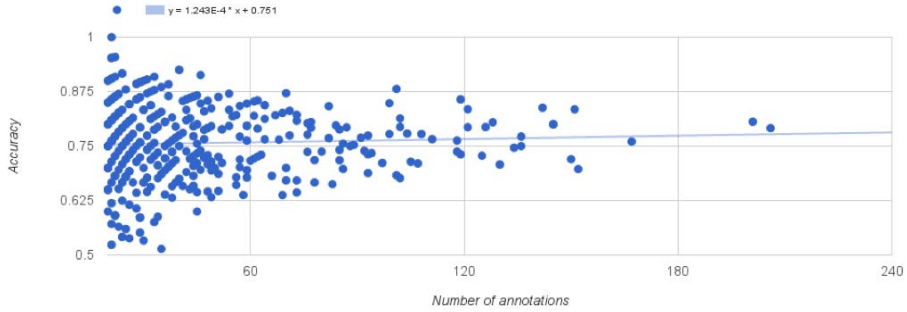

Fig. 7.5: Relation between the number of annotations per worker and individual classification accuracy.

Difference between expert and non-expert annotators

In this experiment, we compare the ratings of quality of the Wikipedia articles by people self-reporting themselves as experts in the domain and by general public. We hypothesize that the experts will provide harsher ratings, as it was the case in the *WikiProject Biography*, where the expert editors considered the majority of reviewed Wikipedia biographies unsatisfactory.[1]

In September 2010, the Wikimedia Foundation introduced the *Article Feedback Tool* (AFT), a project for gathering article feedback from Wikipedia users. It allows the whole Wikipedia community to evaluate articles along the dimensions *Trustworthy*, *Objective*, *Well written* and *Complete* on a five-star scale. The user interface is displayed in figure 7.6. In July 2011, the AFT has been deployed to the whole English Wikipedia.

For our experiments, we use a publicly available dataset of nearly 8 million ratings collected from March to September 2011, retrieved from the Wikimedia Toolserver[2].

Fig. 7.6: Wikipedia Article Feedback box (Version 4) as it appeared on article pages

[1] https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Biography/Assessment
[2] http://toolserver.org/

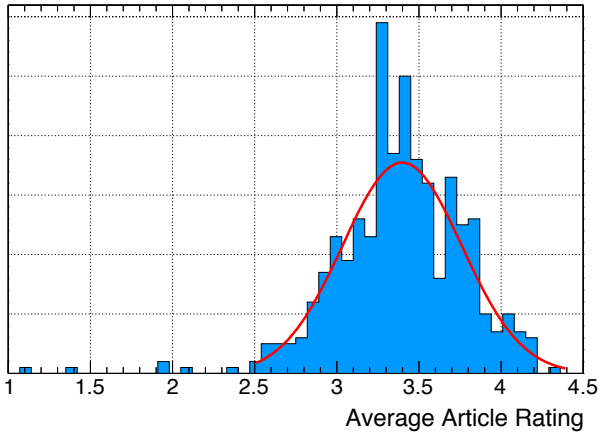

Fig. 7.7: Distribution of average article ratings (by both experts and non-experts together) for the dimension *Well written*

The average ratings per article are normally distributed, with a positively shifted mean. Figure 7.7 illustrates this positive shift of the distribution of average article rating scores, which we observed in all four dimensions. This is in contrast with the point of view of experts from the *WikiProject Biography*, who considered the majority of reviewed biographies unsatisfactory. We explain this phenomenon partly by the fact that the experts in that project focused on reviewing newly created, low quality articles, while the feedback in AFT was rather given to more popular, more often visited articles, which are likely to be more frequently edited and improved.

Based on the information from Wikimedia Toolserver, 24% of all raters claim to have certain expertise in the domain of the article they evaluated. These experts are, on average, less critical in their ratings, as displayed in figure 7.8. This phenomenon has already been observed in the Wikipedia user study conducted by [Chesney, 2006].

7.2.3 Effects of the annotator's personal settings

In this section, we present experiments assessing the interaction between annotator's own demographics and personality and her performance and errors in her annotations. We study the worker prediction of two user traits – gender and age – through Twitter posts. In the annotation task, we use a set of posts from users previously matched to their true age and gender. We let the workers annotate a dataset with previously known and verified age and gender labels, and compare their annotations to this "gold standard" information.

For gender, we use the dataset of Twitter users from [Burger et al., 2011], which are mapped to their self-identified gender by linking them to their other public profiles. This dataset consists of $67,337$ users, from which we create a balanced sample of 1000 users.

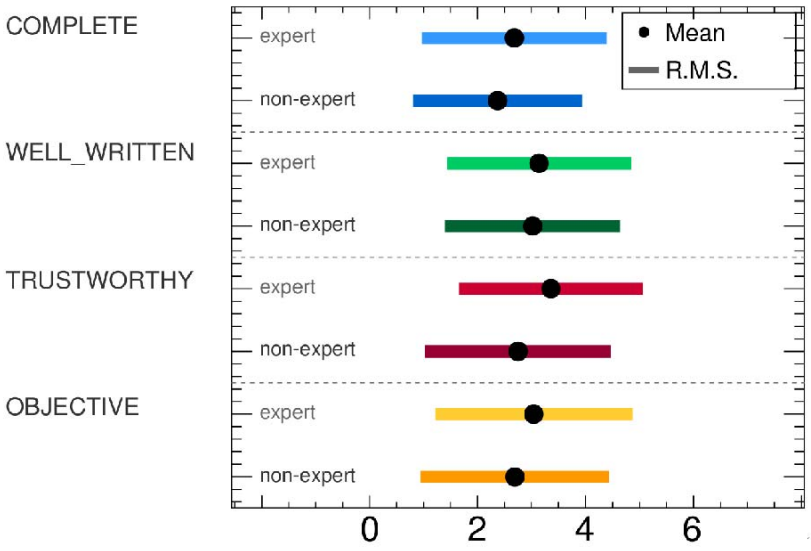

Fig. 7.8: Comparison of average expert and non-expert ratings per each of the Wikipedia quality dimensions. The experts were on average rating the articles higher (the dot). The standard deviation (the bar) should not be confused with the uncertainty of the mean, which is smaller than the marker (dot) size and hence cannot be seen.

The age dataset is obtained by identifying $4,279$ users that were the target of a tweet such as 'Happy X birthday to @USERNAME'. For our experiment, we sampled 1000 users from each of the five culturally meaningful age groups: < 18, $18-22$, $23-30$, $31-40$, $41+$.

Impact of own age and gender on perceived age and gender of Twitter users

We created an annotation task on Amazon Mechanical Turk. Each HIT consisted of 20 tweets sampled from a pool of 100 tweets posted by each user over the past six months. The workers predicted either age or gender, stating the confidence of their rating on a scale from 1 to 5. Each user was assessed independently by 9 workers. We administered a questionnaire to collect worker information. For quality control, we used a set of HITs where the age or gender was explicitly stated within the top 10 tweets displayed to the worker. The control HIT appeared 10% of the time and a worker missing the correct answer twice was excluded from annotation and all his HITs invalidated. Further, we limited the location of workers to the US. An example HIT is presented in Figure 1 at `http://bit.ly/1LFpDx8`. In total, we obtained 38.7% HITs from male workers on gender and 14.6% from 18-22, 49.8% from 23-30, 25.8% from 31-40 and 9.6% from 40+ year old workers on age.

We first analyze the performance of gender prediction across worker's genders. Table **??** shows the gender predictions at HIT level, separated out by worker gender. We can conclude

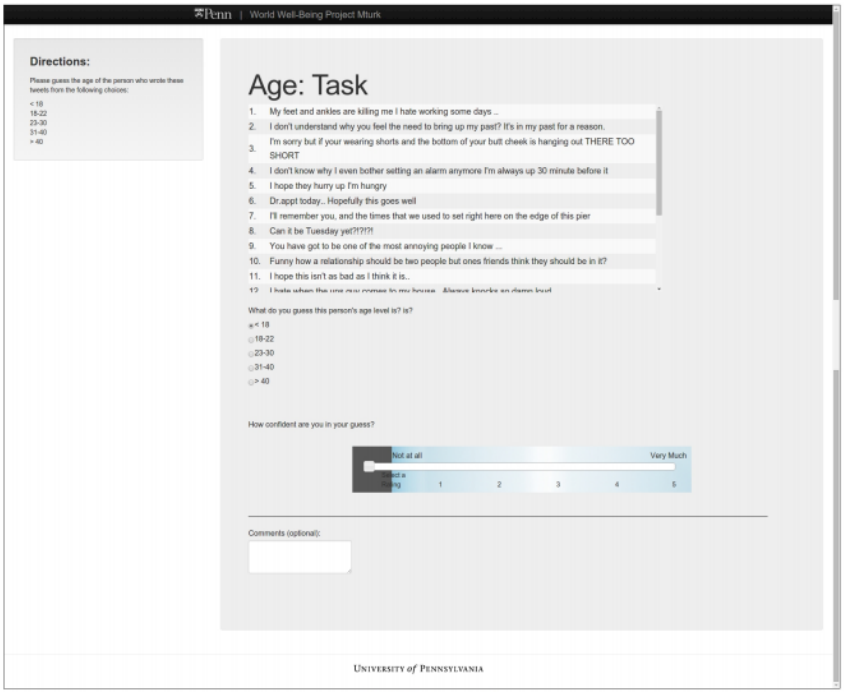

Fig. 7.9: An example HIT as presented to the annotators.

Real/Predicted	**Male**	**Female**	C_M	C_F
Male	0.339 / 0.347	0.159 / 0.138	3.26 / 3.37	3.18 / 3.31
Female	0.110 / 0.112	0.392 / 0.403	2.97 / 3.06	3.65 / 3.85

Tab. 7.2: Left part of the table displays normalized confusion matrices of workers' prediction of gender. Right part of the table displays average self-reported confidence on those prediction groups. In both cases, the values in a cell show the performance of male (left) and female (right) workers respectively.

that females are better at predicting gender overall. Analyzing the errors, we see that males have lower performance mostly due to failing to make accurate predictions when rating males. Overall, females are easier to accurately rate than males ($.392/.403$ vs. $.339/.347$). Additionally, females have a higher overall self-reported confidence in their prediction, even when the prediction is incorrect. The highest relative increase in confidence is when females predict other females (3.85 vs. 3.65 for male workers), which is where the highest decrease in error is also observed ($.138$ vs. $.159$) compared to their male counterparts.

In terms of age, the best precision is reached for the raters in the $31-40$ class, followed by the $23-30$ class. We also notice the trend across all workers, regardless of their own ages, to have similar prior beliefs about the user distribution on Twitter, while adjusting towards their own age group when unsure (indicated by higher recall and lower precision). The easiest class to predict are users between $23-30$ years old. Intriguingly, based on the self-identified user confidence in the ratings, users between $31-40$ are the least confident (3.20), compared to users aged $23-30$ (3.26) who were second best at prediction and

Predicted/worker	<18	18-22	23-30	31-40	41+
18-22	.312 / .221	.555 / .336	.369 / .441	.163 / .366	.081/ .166
23-30	.345 / .219	.491 / .346	.353 / .434	.177 / .311	.172 / .217
31-40	.273 / .266	.518 / .359	.402 / .443	.201 / .336	.271 / .326
41+	.202 / .229	.463 / .394	.384 /.436	.178 / .194	.269 / .194

Tab. 7.3: Performance of the workers of each age class (row) on each twitter user's age class (column). First value in a cell displays recall, second value precision. Note that precision decreases and recall increases as workers approach the class of the user.

second least confident. The groups $18-22$ (3.38) and $40+$ (3.46) were most confident and least accurate.

Impact of own Actively Open-minded Thinking (AOT) levels on perceived age and gender of Twitter users

Successful communication in the complex world of the internet may benefit from users' willingness to seek out new information, particularly evidence that could go against their immediate intuitions, when drawing conclusions about others. This cognitive style is known as Actively Open-Minded Thinking (AOT) [Baron, 1991]. AOT is considered an individual difference, but it can also be learned and trained over time [Baron, 1993].

People high in AOT are characterized by their willingness to disregard their preconceptions and to consciously seek out potentially counter-attitudinal information, particularly other people's perspectives, when making an assessment or decision [Baron, 1993]. They are better at a variety of mental tasks, such as estimating relative amounts [Haran et al., 2013], distinguishing between good and bad arguments [Stanovich and West, 1997], and forecasting future world events [Mellers et al., 2015].

In short, this particular cognitive style is a way of thinking well: having more accurate inferences by ignoring idiosyncratic biases and being open to sources of information that may be threatening to previously held beliefs. Therefore, AOT as a trait measures a dispositional tolerance of having one's beliefs disconfirmed, a longstanding interest in the beliefs and knowledge of other people, and a general level of intellectual humility.

In this study, we examine the same age and gender annotators as in the previous task, but in relation to the 9-item Actively Open-Minded Thinking self-assessment questionnaire [Haran et al., 2013] they were asked to complete before performing the annotation tasks.

Participants (n = 1078) completed the gender task an average of 21 times. Participants with higher trait AOT were more likely to assign the correct gender to authors. The odds ratio estimate was 1.062 (95% CI = 1.019, 1.106) indicating that for each 1-unit increase

in raters' AOT, each guess was 1.062 times more likely to be correct; a guess by a rater whose AOT is 7 would be over 50% more likely to be correct than a rater whose AOT is 1. Overall, participants' guesses were correct 75.70% of the time.

Participants (n =691) completed the age task an average of 11 times. Participants with higher trait AOT were overall more accurate at guessing authors' ages, b = -0.25, p = .002, indicating that for each unit increase in AOT, each guess was approximately a quarter of a year closer to the author's actual age. Overall, participants' guesses diverged from authors' actual ages by an average of 7.31 years.

Textual Differences between Perceived and Actual Traits

We have so far demonstrated that differences exist between the human perception of traits and real traits. Here we directly identify the textual cues that bias humans and cause them to mislabel users. In addition to unigram analysis, in order to aid interpretability of the feature analysis, we group words into clusters of semantically similar words or *topics* using a method from [Preoţiuc-Pietro et al., 2015]. We first obtain word representations using the popular skip-gram model with negative sampling introduced by [Mikolov et al., 2013a] and implemented in the Gensim package (layer size 50, context window 5). We train this model on a **separate** reference corpus containing ~ 400 million tweets. After computing the word vectors, we create a word $\times$ word semantic similarity matrix using cosine similarity between the vectors and group the words into clusters using spectral clustering [Shi and Malik, 2000]. Each word is only assigned to one cluster. We choose a number of 1,000 topics based on preliminary experiments. Further, we use the NRC Emotion Lexicon [Mohammad and Turney, 2013] to measure eight *emotions* (anger, fear, anticipation, trust, surprise, sadness, joy and disgust) and two *sentiments* (negative and positive). A user's score in each of these 10 dimensions is represented as a weighted sum of its words multiplied by their lexicon score.

To study gender perception, we first define a measure of *perceived gender expression*, calculated as the fraction of female guesses out of the 9 guesses for each Twitter user. We then compute univariate correlations of the text-derived features and the user labels. Table 7.4 displays the features with significant correlation to perceived gender expression when controlled for real gender using partial correlation, as well as the standalone correlations with the real gender label and perceived gender expression. Note that all correlations with both males and females have the same sign for both perceived gender and real gender. This highlights that humans are *not* wrong in using these features to make gender assessments. Rather, these stereotypical associates are overestimated by humans.

Perceived – Female				**Perceived – Male**			
Topic	**Perc**	**Real**	**Cont**	Topic	**Perc**	**Real**	**Cont**
such, loving, pretty, beautiful, gorgeous	.416	.348	.176	nation, held, rally, defend, supporters	-.372	-.281	-.176
bed, couch, blanket, lying, cozy	.424	.376	.165	players, teams, crowds, athletes, clubs	-.370	-.284	-.171
hair, blonde, shave, eyebrows, dye	.379	.325	.152	training, team, field, coach, career	-.323	-.246	-.148
friend, boyfriend, bf, bff, gf	.365	.308	.149	heat, game, nba, lakers, playoff	-.314	-.237	-.145
girl, lucky, she's, you're, he's	.378	.336	.143	draft, trade, deadline, stat, retire	-.303	-.223	-.143
sweet, angel, honey, pumpkin, bunny	.365	.322	.138	ref, offensive, foul, defensive, refs	-.324	-.255	-.142
cleaning, laundry, packing, dishes, washing	.350	.307	.133	second, third, grade, century, period	-.282	-.195	-.142
awake, dream, sleep, asleep, nights	.327	.276	.130	former, leader, chief, vice, minister	-.316	-.244	-.142
cry, heart, smile, deep, whenever	.331	.288	.125	private, claim, jail, removed, banned	-.299	-.224	-.138
cake, christmas, gift, cupcakes, gifts	.330	.287	.125	war, action, army, battle, zone	-.323	-.263	-.135
evening, day, rest, today, sunday	.249	.180	.118	security, transition, administration, support	-.295	-.225	-.134
light, dark, colors, bright, rainbow	.244	.178	.114	general, major, impact, signs, conflict	-.295	-.227	-.132
shopping, home, spend, packed, grocery	.326	.301	.111	largest, launches, announces, lands, add	-.273	-.196	-.132
dreams, live, forget, remember, along	.247	.194	.107	guns, planes, riot, weapons, soldiers	-.251	-.165	-.131
darling, xo, hugs	.259	.211	.106	title, tech, stats, division, technical	-.314	-.258	-.129
brother, mom, daddy, daughter, sister	.302	.275	.105	breaking, turns, breaks, falls, puts	-.266	-.190	-.128
moment, awkward, laugh, excitement, laughter	.282	.247	.103	million, billion	-.277	-.206	-.128
totally, awesome, favorite, love, fave	.272	.233	.103	steve, joe, dave, larry, phil	-.294	-.236	-.124
breakfast, dinner, lunch, cooking, meal	.280	.245	.103	football, pitch, blues, derby, lineup	-.276	-.211	-.124
makeup, glasses, lipstick	.264	.223	.102	ceo, warren	-.240	-.160	-.123
Unigrams	**Perc**	**Real**	**Cont**	**Unigrams**	**Perc**	**Real**	**Cont**
love,my,so,!,you,I,her,hair,feel,today,	.339	.259	.156	game,the,sports,against,football,teams,	-.270	-.236	-.130
friends,baby,cute,girls,beautiful,me,heart,				player,fans,report,team,ebola,vs,nba,games,			
little,shopping,happy,because,wonderful,				economy,score,government,ceo,americans,			
gorgeous,bed,clothes,am,have,yay,your	.179	.081	.071	goals,app,penalties,play,shit,political,war	-.117	-.062	-.065
Emotion	**Perc**	**Real**	**Cont**	**Emotion**	**Perc**	**Real**	**Cont**
Joy	.255	.245	.091	Anger	-.156	-.117	-.076
				Fear	-.183	-.145	-.084

Tab. 7.4: Textual features highlighting errors in human perception of gender compared to ground truth labels. Table shows correlation to perceived gender expression (**Perc**), to ground truth (**Real**) and to perceived gender expression controlled for ground truth (**Cont**). All correlations of gender unigrams, topics and emotions are statistically significant at $p < .001$ (t-test)

Gender – High Confidence				**Gender – Low Confidence**			
Topic	Conf	Real	Cont	Topic	Conf	Real	Cont
sibling,flirted,married,husband,wife	(.028)	(.071)	.240	wiser,easier,shittier,happier,worse	-.277	(.081)	-.295
fellaz,boyss,dayz,girlz,gurlz,sistas	(.118)	(.113)	.221	agenda,planning,activities,schedule	-.285	(.020)	-.289
brother, mom, daddy, daughter, sister	(.127)	.241	.214	horoscope,zodiac,gemini,taurus,virgo	-.269	(.087)	-.288
bathroom,wardrobe,toilet,clothes,bath	(.017)	.220	.212	reshape,enable,innovate,enhance,create	-.253	(-.110)	-.235
looked,winked,smiled,lol'd,yell,stare	(.035)	(.089)	.201	imperfect,emotional,break-down,commit	-.227	.024	-.232
hair, blonde, shave, eyebrows, dye	.163	.182	.199	major,brief,outlined,indicates,wrt	-.234	(-.045)	-.226
pyjama,shirt,coat,hoody,trousers	(.077)	(-.010)	.191	justification,circumstance,boundaries	-.224	(-.014)	-.221
awake, dream, sleep, asleep, nights	.160	(.132)	.184	experiencing, explanations, expressive	-.225	(-.039)	-.217
totally, awesome, favorite, love, fave	(.063)	(.135)	. 183	inferiority,sufficiently,adequately	-.209	(-.015)	-.206
days,minutes,seconds,years,months	(.087)	(-.013)	.177	specified,negotiable,exploratory,expert	-.190	(-.014)	-.187
baldy,gangster,boy,kid,skater,dude	(.071)	(.027)	.173	multiple,desirable,extensive,increasingly	-.199	(-.092)	-.183
shopping,grocery,ikea,manicure	(.052)	.204	.173	anticipate,optimist,unrealistic,exceed	(.053)	(.023)	-.182
happy,birthdayyyy,happyyyy,bday	.180	.222	.172	organisation,communication,corporate	-.200	-.148	-.175
girl, lucky, she's, you're, he's	(.118)	(.060)	.172	hostile,choppy,chaotic,cautious,neutral	-.178	(-.033)	-.172
worst,happiest,maddest,slowest,funniest	.173	(.113)	.172	security, transition, administration, supports	.185	(-.079)	-.170
bazillion,shitload,nonstop,spent,aand	.162	(.084)	.167	diminished,unemployment,rapidly	-.181	(-.101)	-.163
Emotion	**Conf**	**Real**	**Cont**	**Emotion**	**Conf**	**Real**	**Cont**
Joy	.202	.245	.164	–			
Anticipation	.140	(.086)	.124				
Unigrams	**Conf**	**Real**	**Cont**	**Unigrams**	**Conf**	**Real**	**Cont**
I,my,this,was,me,so,had,like,	.312	.267	.360	more,may,might,although,	.290	.081	.310
her,night,she,just,hair,gonna,				emotional,your,eager,url,			
ever,last,shirt,				desires,relationship,seem,existing,			
kid,girls,love	(.076)	(.047)	.160	emotions,surface,practical,source	.150	-.014	.180

Tab. 7.5: Textual features highlighting high and low confidence in human perception of gender. Table shows correlation to average self-reported confidence (**Conf**), to ground truth (**Real**) and with self-reported confidence controlled for ground truth (**Cont**). All correlations of gender unigrams, topics and emotions are statistically significant at $p < .001$ (t-test), except of the values in brackets.

Perceived – Older				Perceived – Younger			
Topic	**Perc**	**Real**	**Cont**	**Topic**	**Perc**	**Real**	**Cont**
golf, sport, semi, racing	.278	(.085)	.226	she's, youre, hes, lucky, girl, slut	-.328	-.243	-.184
bill, union, gov, labor, cuts	.349	.287	.181	boys, girls, hella, homies, ya'll	-.297	-.236	-.155
states, public, towns, area, employees, immigrants	.301	.213	.173	dumb, petty, weak, lame, bc, corny	-.295	-.232	-.155
roger, stanley, captain	.232	(.105)	.167	miss, doing, chilling, how's	-.305	-.268	-.145
available, service, apply, package, customer	.279	.197	.160	heart, cry, smile, deep, hug	-.258	-.186	-.144
serving, prime, serve, served, freeze	.215	(.097)	.154	friend, bestfriend, boyfriend, bff, bestest	-.281	-.254	-.127
support, leaders, group, youth, educate	.228	.121	.153	ugly, stubborn, bein, rude, childish, greedy	-.238	-.182	-.126
hillary, clinton, obama, president, scott, ed, sarah	.289	.230	.150	bitch, fuck, hoe, dick, slap, suck	-.278	-.251	-.125
via, daily, press, latest, report, globe	.311	.272	.149	kinda, annoying, weird, silly, emo, retarded, random	-.242	-.193	-.124
diverse, developed, multiple, among, several, highly	.266	.195	.147	everyone, everything, nothing, does, anyone, else	-.201	-.218	-.118
military, terrorist, citizens, iraq, refugees	.287	.235	.146	bruh, aye, fam, doin, yoo, dawg	-.227	-.178	-.117
julia, emma, annie, claire	.180	(.056)	.145	ever, cutest, worst, weirdest, biggest, happiest	-.275	-.264	-.115
liberty, pacific, north, eastern, 2020	.260	.198	.139	seriously, crazy, bad, shitty, yikes, insane	-.208	-.152	-.114
brooklyn, nyc, downtown, philly, hometown	.213	.120	.139	whoops, oops, remembered, forgot	-.179	(-.104)	-.113
Unigrams	**Perc**	**Real**	**Cont**	**Unigrams**	**Perc**	**Real**	**Cont**
golf, our, end, delay, favourite, low, holes, original, branch, the, of, stanley, our, . , story, , , forever, exciting, great, what, community, hurricane, for, brands, toward, kids, regarding, upcoming	.321	.(063)	.282	me, i, when, like, you, so, dude, don't, hate, im, u, girl, hate, life, my, wanna, literally, r, really, cute, someone, youre, miss, me , want, this okay, rt, school, snapchat, shit, crying	-.535	-.489	-.294
	.208	(.101)	.145		-.256	(-.051)	-.117
Emotion	**Perc**	**Real**	**Cont**	**Emotion**	**Perc**	**Real**	**Cont**
Positive	.325	.268	.166	Disgust	-.177	-.131	-.094
Trust	.243	.184	.130	Negative	-.104	(-.031)	-.084
Anticipation	.212	.176	.102	Sadness	-.126	-.072	-.081
				Anger	-.070	(-.009)	-.065

Tab. 7.6: Textual features highlighting errors in human perception of age compared to ground truth labels. Table shows correlation to perceived age expression (**Perc**), to ground truth (**Real**) and to perceived age expression controlled for ground truth (**Cont**). All correlations of age unigrams, topics and emotions are statistically significant at $p < .001$ (t-test), except of the values in brackets.

Age – High Confidence				Age – Low Confidence			
Topic	**Conf**	**Real**	**Cont**	**Topic**	**Conf**	**Real**	**Cont**
school, student, college, teachers, grad, classroom	.242	(-.054)	.227	mocho, gracias, chicos, corazon, quiero	-.195	(-.042)	-.207
done, homework, finished, essay, procrastinating	.251	-.125	.219	sweepstakes, giveaway, enter, retweet, prize	(-.044)	-.278	-.134
math, chem, biology, test, study, physics	.227	(-.060)	.210	injures, shot, penalty, strikes, cyclist, suffered	-.149	.153	-.108
cant, can't, wait, till, believe, afford	.226	-.171	.183	final, cup, europa, arsenal, match, league	-.135	.107	-.106
tomorrow, friday, saturday, date, starts	.175	(-.014)	.171	juventus, munich, lyon, bayern, 0-1	(-.101)	(-.005)	-.103
invitations, prom, attire, wedding, outfit, gowns	.172	(.005)	.170	castlevania, angels, eagles, demons, flames	-.138	.138	(-.101)
soexcited, next, week, weekend, summer, graduation	.153	(.009)	.155	devil, sword, curse, armor, die, obey	(-.081)	(-.055)	(-.097)
aaand, after, before, literally, off, left, gettingold	.182	(-.103)	.154	football, reds, kickoff, derby, pitch, lineup	-.125	.106	(-.096)
sleepy, work, shifts, longday, exhausted, nap	.126	(.064)	.144	anime, invader, shock, madoka, dragonball	(-.071)	(-.080)	(-.095)
life, daydream, remember, cherish eternally, reminiscing	.200	-.228	.143	paranormal, dragon, alien, zombie, dead	(-.099)	(.025)	(-.092)
happyyyy, birthdaaaay, b-day, bday, belated	.187	-.173	.142	earthquake, magniture, aftermath, devastating, victims	(-.101)	(.040)	(-.090)
Unigrams	**Conf**	**Real**	**Cont**	**Unigrams**	**Conf**	**Real**	**Cont**
my, i'm, can't, i, school, so, to, class, semester, college, homework, prom, me, in my,	.375	-.350	.314 ↓	rt, his, league, epic warriors, ! ,	(-.023)	-.320	-.128 ↓
friends, literally, when, exam, nap	.180	(.080)	.157	vintage	-.130	(.071)	-.111
Emotion	**Conf**	**Real**	**Cont**	**Emotion**	**Conf**	**Real**	**Cont**
Trust	(.077)	.184	.134	–			
Joy	.125	(.009)	.128				
Positive	(.031)	.268	.115				
Anticipation	(.060)	.176	.114				

Tab. 7.7: Textual features highlighting high and low confidence in human perception of age. Table shows correlation to average self-reported confidence (**Conf**), to ground truth (**Real**) and with self-reported confidence controlled for ground truth (**Cont**). Correlation values of age unigrams, topics and emotions statistically significant at $p < .001$ (t-test) unless in brackets.

By analyzing the topics that are still correlated with perception after controlling for ground truth correlation, we see that topics related to sports, politics, business and technology are considered by annotators to be stronger cues for predicting males than they really are. Female perception is dominated by topics and words relating to feelings, shopping, dreaming, housework and beauty. For emotions, joy is perceived to be more associated to females than the data shows, while users expressing more anger and fear are significantly more likely to be perceived as males than the data supports.

Our crowdsourcing experiment allowed annotators to self-report their confidence in each choice. This gives us the opportunity to measure which textual features lead to higher self-reported confidence in predicting user traits. Table 7.5 shows the textual features most correlated with self-reported confidence of the annotators when controlled for ground truth, in order to account for the effect that overall confidence is on average higher for groups of users that are easier to predict (i.e., females in case of gender, younger people in case of age).

Annotations are most confident when family relationships or other people are mentioned, which aid them to easily assign a label to a user (e.g., 'husband'). Other topics leading to high confidence are related to apparel or beauty. Also the presence of joy leads to higher confidence (for predicting females based on the previous result). Low confidence is associated with work related topics or astrology as well as to clusters of general adverbs and verbs and tentatively, to a more formal vocabulary e.g., 'specified', 'negotiable', 'exploratory'. Intriguingly, low confidence in predicting gender is also related to unigrams like 'emotions', 'relationship', 'emotional'.

Table 7.6 displays the features most correlated with perceived age – the average of the 9 annotator guesses – when controlled for real age, and the individual correlations to perceived and real age.

Again, annotators relied on correct stereotypes, but relied on them more heavily than warranted by data. The results show that the perception of users as being older compared to their biological age, is driven by topics including politics, business and news events. Vocabulary contains somewhat longer words (e.g., 'regarding', 'upcoming', 'original'). Additionally, annotators perceived older users to express more positive emotions, trust and anticipation. This is in accordance with psychology research, which showed that both positive emotion [Mather and Carstensen, 2005] and trust [Poulin and Haase, 2015] increase as people get older.

The perception of users being younger than their biological age is highly correlated with the use of short and colloquial words, and self-references, such as the personal pronoun 'I'. Remarkably, the negative sentiment is perceived as more specific of younger users, as

well as the negative emotions of disgust, sadness and anger, the latter of which is actually uncorrelated to age.

Table 7.7 displays the features with the highest correlation to annotation confidence in predicting age when controlling for the true age, as well as separate correlations to real and perceived age. Annotators appear to be more confident in their guess when the posts display more joy, positive emotion, trust and anticipation words. In terms of topics mentioned, these are more informal, self-referential or related to school or college. Topics leading to lower confidence are either about sports or online contests or are frequently retweets.

7.3 Chapter summary

In this chapter, we suggested several factors which may influence the quality of obtained annotations. We explicitly examined the influence of the formulation of the task, the annotator's assumptions about the data and the personal settings of an annotator. We have experimentally shown that the task formulation can influence the shape of the distribution of answers (ratings) obtained, and that the granularity of answer options provided can influence the disagreement patterns between annotators. We pointed out that annotator's general assumptions about the data can influence the annotation errors. We demonstrated this phenomenon on crowdsourcing age estimations for social media users, showing that these are systematically underestimated. We further show that the annotator's perception changes with the amount of data annotated and that expert annotations tend to differ from the general population. We suggested that for certain tasks the annotator's performance can depend on her personal settings, e.g. demographic or psychological factors. We examine this hypothesis on the task of estimating author's age and gender from text, and show that while all annotators are prone to certain stereotypes, some demographic groups do so more than others. To our knowledge, this was the first study to systematically analyze differences between real user traits and traits as perceived from text. Correlation analysis showed that aspects of stereotypes associated with errors tended not to be completely wrong but rather poorly applied. Annotators generally exaggerated the diagnostic utility of behaviors that they correctly associated with one group or another. Further, we used the same methodology to analyze self-reported confidence. Some of the results of this chapter were published in [Flekova et al., 2016a].

8 Conclusions

> *Science is the acceptance of what works and the rejection of what does not. That needs more courage than we might think.*
>
> — **Jacob Bronowski**

This dissertation focused on studying the connection between word semantics and text classification. For long, it was assumed that the lexical-semantic knowledge will not lead to better classification results, as the meaning of every word can be directly learned from the document itself. In this thesis, we show that this assumption is not valid as a general statement and present several approaches how resource-based semantic knowledge will lead to better results. Moreover, we show, why these improved results can be expected.

Figure 8.1 provides a comprehensive overview of our experimental results comparable across the key methods and datasets we used in this thesis. Overall, our novel combination of word and supersense embeddings in a multichannel CNN+LSTM classification model yielded the best results across classification tasks addressed in this thesis. The CNN layer, however, appeared to be sensitive to varying document length and brought higher performance gains on shorter documents. As we can see from the figure though, supersenses brought an improvement over word features on every task, even in the case where the deep learning model was outperformed by an SVM classifier using word and supersense

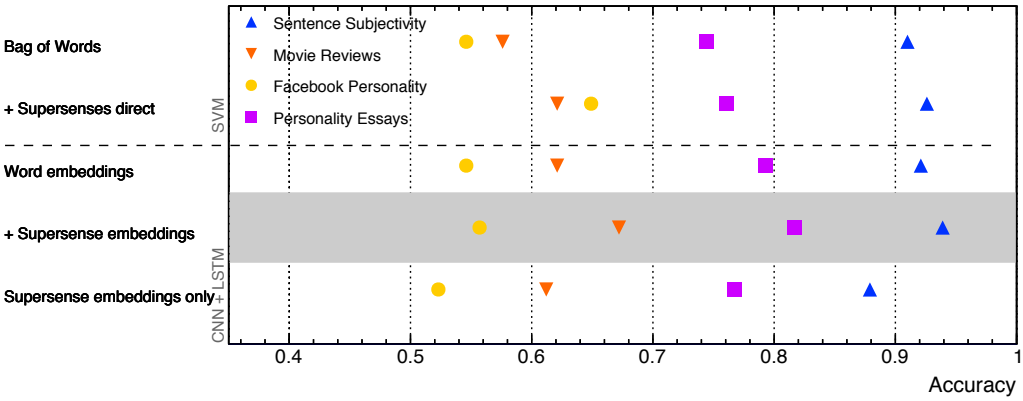

Fig. 8.1: Comparative overview of the main experimental results in this thesis.

features. This is a positive signal for combining supersenses with more advanced, emerging classification architectures in future work.

Below, we summarize the answers to the most important research questions (RQ) addressed in this thesis.

RQ: Which impact does the word sense disambiguation have on document classification and why?

One of the first problems that is encountered by any natural language processing system is that of lexical-semantic ambiguity. In text classification tasks, the ambiguity problem has been often neglected, with the prevalent assumption, that the document contain enough words to safely ignore the ambiguous ones, and the usefulness of lexical-semantic resources for document classification was not commonly accepted [Vossen et al., 2006, Voorhees and Harman, 1997, Sanderson, 1994, Gonzalo et al., 1998]. However, while both negative and positive word sense disambiguation results were occasionally reported [Vossen et al., 2006, Rentoumi et al., 2009], to our knowledge none of the previous works focused explicitly on explaining why the word sense disambiguation does not provide consistently better classification performance although it intuitively should.

In this thesis, we quantified the impact of word ambiguity on a range of document classification tasks and domains, systematically evaluating the performance of selected resource-based word sense disambiguation algorithms, comparing bag-of-words and bag-of-senses classification settings. We conduced an extensive error analysis on these tasks and conclude that the improvement in text classification accuracy, when occurring, can be largely attributed to the lemmatization and part-of-speech disambiguation, while the fine-grained sense distinction did not help to disambiguate between documents.

In accordance with [Kilgarriff, 1997] and [Krovetz, 2002], we note that the lexicographic sense distinctions provided by the lexical-semantic resources are not always optimal for every text classification task, and propose an alternative technique for disambiguation of word meaning in its context for sentiment analysis applications. We performed an extrinsic and intrinsic evaluation of our sentiment disambiguation method, and demonstrated, that our sentiment scores match human perception of polarity and bring improvement in the classification results. We conclude from this experiment that refining general-language lexical-semantic resources with a task- and domain-specific lexica derived from the data is helpful for a more efficient discrimination between class labels.

RQ: Is it helpful to supply the classifier with an additional semantic information about the content of the document? If so, how?

The second problem we identify in text classification is the lexical gap, or sparsity problem. For any document, the words used represent only a tiny fraction of the words in the total possible vocabulary. We proposed to address the word sparsity problem by automatically enriching the training and testing data with conceptual annotations accessible through lexical semantic resources. We show that such conceptual information (which we call "supersenses"), in combination with the previous word sense disambiguation step, helps to build more robust classifiers and improves classification performance of multiple tasks. We further circumvent the sense disambiguation step by training a supersense tagging model directly. We have shown that direct supersense annotation with a pre-trained model leads to better results than accessing supersense labels through fine-grained word senses. This can be attributed to the errors occurring in fine-grained WSD, in which we use knowledge-based algorithms, i.e., in some cases a system has to choose between more than 20 senses only based on the glosses available in WordNet and the word context in the document. However, these 20 or more senses are often grouped into only one or two supersenses, therefore the direct supersense disambiguation task is easier. On the other hand, when using a supervised supersense classification method instead, such as in our case, an annotated training corpus is needed, which may pose a challenge for some languages or domains.

We additionally illustrated how our approach can be extended to other lexical-semantic resources suitable for the task at hand, presenting a method to extend our approach beyond WordNet supersenses, using the sense-level links between WordNet and VerbNet. Not being limited to a single lexical-semantic resource is important, since different tasks may require a different level of granularity or different focus. For example, the verb *to love* has a supersense EMOTION in WordNet and ADMIRE in VerbNet, which may be relevant for a classification task where the verbs are expected to play an important role. Similarly, in a task focused on nouns, we may consider that the WordNet Domains [Magnini et al., 2001] resource provides a relatively fine-grained coverage of sports and scientific disciplines. Our methodology for obtaining supersense embeddings once we know the WordNet sense of a word is applicable to any of these and other resources, for example exploiting the sense-level links available in UBY [Gurevych et al., 2012].

RQ: Why do supersense annotations matter in these tasks?

For each of the classification task at hand, we analyzed the most informative features and classification errors, and proposed explanations for the way supersenses contribute to the classification.

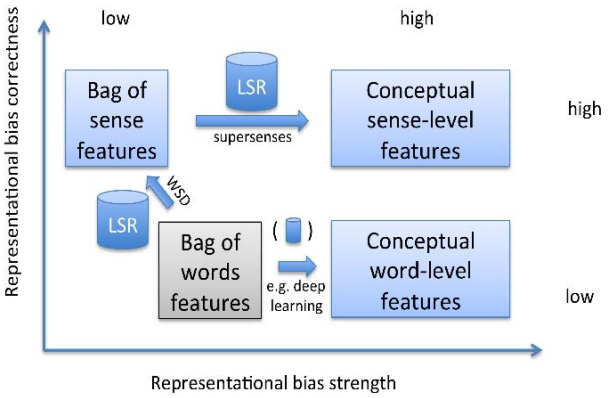

Fig. 8.2: Our model of contribution of lexical-semantic features to the generalization of a text classification learning algorithm via representational bias.

As illustrated on Figure 8.2, our interpretation of the improvement is that supersenses, by providing useful abstractions over individual words, lead to a desired increase in the strength of the representational bias of a text classification learning algorithm. Providing these abstractions on sense level, derived from WordNet synset structures grouped to high-level concepts, contributes to achieving higher correctness of the representational bias than the word-based features are capable of. As an example of the utility of high-level concepts, we have shown that the highest ranked supersenses for personality prediction of fictional characters are consistent with the previous findings of psychologists for human personality. Similarly, we have examined how individual supersenses improve classification over word embeddings in movie reviews, where the creative language is used, and word-level features would wrongly assign such words a different, more frequent interpretation.

RQ: With the rise of deep learning, outperforming the traditional, manually feature-engineered classifiers, on a wide range of tasks, using only word-based vectors, is the lexical-semantic information redundant and the approach obsolete?

One of the common text classification problems is the curse of dimensionality. Several techniques for dimensionality reduction were proposed, most recently the representation learning, producing continuous word representations in a dense vector space, also known as word embeddings. However, since these vectors are again produced on an ambiguous word level, the valuable piece of information about possible distinct senses of the same word is lost, in favor of the most frequent one(s). The resulting vector set is therefore very dependent on which corpora are used to capture the meaning of the words, and can be

easily biased. In this thesis, we explored if, or how, we can use external lexical-semantic resources to regain the sense-level notion of semantic relatedness back while operating within the deep learning paradigm. We proposed and evaluated a method to integrate supersenses into the deep learning setup by building supersense embeddings (as a parallel to word embeddings) from large sense-disambiguated resources (including, e.g., English Wikipedia).

Deep learning approaches with word-based vectors are suitable for capturing the notion of semantic similarity, however, additional inference beyond similarity is needed for accurate predictions. We proposed a novel concept of joint dense vectors of words and supersenses (supersense embeddings), built those and evaluated their semantic properties. We examined the impact of different training data for the quality of these embeddings, provided a visual interface for comparing their properties, and demonstrated how to employ them in deep learning text classification experiments. Using convolutional and recurrent neural networks enriched with supersense embeddings, we achieve a significant classification accuracy improvement in a range of downstream classification tasks. We qualitatively explored the classification results and found that the performance gains with our network architecture are higher for shorter documents and that supersenses help the model to generalize over rare expressions, which is a promising strategy to reduce the size of the training dataset required to train a deep learning model.

RQ: Does the choice of the strategy to obtain gold-standard classification labels influence their quality?

Human experts are prone to diverse biases when classifying data. We hypothesized that there are several factors which may influence the quality of obtained annotations. We explicitly examined the influence of the formulation of the task, the annotator's prior assumptions about the data, and the personal settings of an annotator.

We have experimentally shown that the task formulation can influence the shape of the distribution of answers (ratings) obtained, and that the granularity of answer options provided can influence the disagreement patterns between annotators. We pointed out that annotator's general assumptions about the data can influence the annotation errors. We demonstrated this phenomenon on crowdsourcing age estimations for social media users, showing that these are systematically underestimated. We further show that the annotator's perception changes with the amount of data annotated and that expert annotations tend to differ from the general population. We suggested that for certain tasks the annotator's performance can depend on her personal settings, e.g. demographic or psychological factors. We examine this hypothesis on the task of estimating author's age and gender from text, and show that while all annotators are prone to certain stereotypes, some demographic

groups do so more than others. Consciously avoiding such annotation errors shall lead to classification models of higher quality and fairness.

Summary

To summarize, we could show that lexical-semantic knowledge can improve text classification tasks by supplying the hierarchy of abstract concepts, enabling better generalization over words. We proposed a novel idea of jointly modeling words and supersenses in the same embedding space and have shown that these methods are viable especially in combination with deep learning techniques.

Future Research Directions

In this thesis, we mainly focused on leveraging the concept of supersenses, available for nouns and verbs in WordNet. As of 2016, supersense taxonomies are being developed also for adjectives, adverbs and prepositions, offering research extensions on supersense embeddings. For future work, we propose to investigate to what extent our supersense approach improves classification tasks in other languages. The coarse semantic categorization contained in supersenses was shown to be preserved in translation [Schneider et al., 2013], making them a perfect candidate for a multilingual adaptation of the vector space, e.g. extending [Faruqui and Dyer, 2014] or [Klementiev et al., 2012].

We also propose to evaluate if supersenses still improve results when trained on data from another domain, and which domains have the potential to be meaningfully combined for joint learning. Additionaly, a different level of granularity of concepts, such as WordNet Domains [Magnini and Cavaglia, 2000], and different resources could be explored. Furthermore, some of the previously proposed alternative approaches for building word sense embeddings, e.g. with a modified learning objective [Rothe and Schütze, 2015, Chen et al., 2014, Iacobacci et al., 2015], could be eventually extended to supersenses, and as such provide an interesting comparison benchmark by building the supersense-enriched word embeddings in an alternative way.

An interesting research direction could be to integrate supersenses into a deep multi-task learning architecture. Multi-task learning (MTL) can be seen as a way of regularizing model induction by sharing feature representations with other inductions [Caruana, 1993]. Thus, MTL models tend to perform better than when learning the classification tasks separately (report of [Alonso and Plank, 2016] on the MTL effectiveness gives a more diverse picture). [Søgaard and Goldberg, 2016] have shown that in the NLP context, deep MTL can be used not only for the tasks of the same “level”, but that it can be beneficial to learn the tasks,

traditionally perceived as “low-level” in the NLP processing pipeline (for example, POS tagging), in the lower layers of the deep learning architecture. Such MTL configuration is then beneficial for the “high-level” tasks in the outermost layer (for example, combinatory categorical grammar (CCG) supertagging). In our case, supersense tagging could serve as a low-level task for various sentence-level semantic classification problems.

On a more general level, this work focused on text classfication, yet it is unclear how these findings transfer to other areas of NLP, such as information retrieval or information extraction.

The benefits of supersenses in text classification also conceptually overlap with the idea of zero-shot learning [Larochelle et al., 2008, Palatucci et al., 2009, Socher et al., 2013a], in a way that abstracting unseen lexemes to supersenses allows us to build a model for them by projecting/recycling the knowledge from seen lexemes with the same supersenses. Zero-shot learning is an extreme form of transfer learning, which attempts to assign class labels at test time without seeing any examples of it at training time [Goodfellow et al., 2016]. This learning is only possible when additional information has been exploited during training, allowing some sort of generalization. For example, the classifier might be able to recognize an image of a cat, knowing that cats have four legs, pointed ears and a tail [Palatucci et al., 2009]. [Socher et al., 2013a] show that the language feature representations for the zero-shot classes of images can be learned from unsupervised and unaligned corpora as word embeddings instead of manually defining semantic or visual attributes. Since they use a set of word embeddings to represent each image class, our supersense embeddings could be easily used to enrich such representations with an additional, possibly more robust semantic information.

We also hope that a follow-up research will be pursued regarding our findings in Chapter 7, addressing annotator bias and the ethics and fairness in machine learning and NLP in general. Several workshops and conferences focusing on this area are already gaining momentum, and we can only emphasize that the human factor is vital in any machine learning applications and understanding its influence on the automated decisions should be considered in the future technology-driven society.

Software Packages and Datasets

Below we provide an overview of the software packages and datasets resulting from this thesis, which we make openly available.

Open source software

Supersense tagger Python source code to the supersense tagger, described in our Chapter 4.4, is available at the following URL:

`https://github.com/UKPLab/acl2016-supersense-embeddings/tree/master/tagger`

Deep learning model for sentiment classification CNN-LSTM neural network setup integrating semantic feature vectors, described in the paper [Flekova and Gurevych, 2016], is available at:

`https://github.com/UKPLab/acl2016-supersense-embeddings/tree/master/classification`

Supersense embedding training A GenSim script to build supersense word2vec embeddings on Wikipedia is available at:

`https://github.com/UKPLab/acl2016-supersense-embeddings/tree/master/embeddings-creator`

Feature-based SVM text classification framework with stylistic and semantic features A Java text classification framework [Daxenberger et al., 2014] for predicting text authors' age and gender has been submitted as a working software to the PAN Author Profiling challenge [Flekova and Gurevych, 2013] to enable benchmarking in the following years of the challenge. We have later adapted the software for German language and used for various text classification tasks, for example, classifying the teaching style in German schools, a source code for which is available at:

`https://github.com/UKPLab/jlcl2015-pythagoras`

NLP resources

Pretrained embeddings Pretrained Wikipedia word and supersense embeddings described in this thesis (Chapters 4 and 6) can be downloaded in Word2vec format at:

`public.ukp.informatik.tu-darmstadt.de/wikipedia/supersense-embeddings.txt.zip`

Sentiment polarity switching bigrams Polarity switching sentiment bigrams described in Chapter 3 can be downloaded here:

`https://www.ukp.tu-darmstadt.de/data/sentiment-analysis/inverted-polarity-bigrams/`

Datasets

Wikipedia Article Feedback ratings Article revision IDs rated on average above 3.5 and below 2.5 in each Wikipedia Article Feedback dimension, described in Chapter 7, are available at:

`https://www.ukp.tu-darmstadt.de/data/quality-assessment/wikipedia-article-feedback/`

Wikipedia Article Feedback ratings The dataset of personality labels of characters in books, which we manually collected and which is described in Chapter 5, is available at:

`https://www.ukp.tu-darmstadt.de/fileadmin/user_upload/Group_UKP/data/sentiment-analysis/Personality-GOLD_characters.tsv`

Other

Personality quiz game Fictional character personality assessment game, adapted from psychology questionnaires and discussed in Chapter 7, is available online at:

`http://books.ukp.informatik.tu-darmstadt.de/`

Bibliography

[Agirre et al., 2009a] Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., and Soroa, A. (2009a). A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of Human Language Technologies 2009: The Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT),* Boulder, Col., 31 May – 5 June 2009, pages 19–27. Association for Computational Linguistics.

[Agirre et al., 2011] Agirre, E., Bengoetxea, K., Gojenola, K., and Nivre, J. (2011). Improving dependency parsing with semantic classes. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL),* Portland, Oreg., 19–24 June 2011, pages 699–703. Association for Computational Linguistics.

[Agirre et al., 2009b] Agirre, E., De Lacalle, O. L., Soroa, A., and Fakultatea, I. (2009b). Knowledge-based WSD and specific domains: Performing better than generic supervised WSD. In *Proceedings of the 21th International Joint Conference on Artificial Intelligence (IJCAI),* Pasadena, Cal., 14–17 July 2009, pages 1501–1506.

[Agirre and Edmonds, 2006] Agirre, E. and Edmonds, P. (2006). *Word Sense Disambiguation: Algorithms and Applications*. Springer.

[Agirre et al., 2010] Agirre, E., Fellbaum, C., Marchetti, A., and Toral, A. (2010). SemEval-2010 Task 17 : All-words Word Sense Disambiguation on a Specific Domain. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 123–128.

[Akkaya et al., 2011] Akkaya, C., Wiebe, J., Conrad, A., and Mihalcea, R. (2011). Improving the impact of subjectivity word sense disambiguation on contextual opinion analysis. In *Proceedings of the 15th Conference on Computational Natural Language Learning*, pages 87–96. Association for Computational Linguistics.

[Alam and Riccardi, 2014] Alam, F. and Riccardi, G. (2014). Predicting personality traits using multimodal information. In *Proceedings of the 2nd Workshop on Computational Personality Recognition (WCPR14) at the 22nd international ACM conference on multimedia* Orlando, Fl., 7 November 2014, pages 15–18. ACM.

[Alonso and Plank, 2016] Alonso, H. M. and Plank, B. (2016). Multitask learning for semantic sequence prediction under varying data conditions. *arXiv preprint arXiv:1612.02251*.

[Alonso and Romeo, 2014] Alonso, H. M. and Romeo, L. (2014). Crowdsourcing as a preprocessing for complex semantic annotation tasks. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)* Reykjavik, Iceland, 26–31 May 2014, pages 229–234. European Language Resources Association (ELRA).

[Andreevskaia and Bergler, 2006] Andreevskaia, A. and Bergler, S. (2006). Mining WordNet for a fuzzy sentiment: Sentiment tag extraction from WordNet glosses. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, EACL, pages 209–216.

[Aran and Gatica-Perez, 2013] Aran, O. and Gatica-Perez, D. (2013). Cross-domain personality prediction: From video blogs to small group meetings. In *Proceedings of the 15th ACM International Conference on Multimodal Interaction*, pages 127–130, New York, NY, USA. ACM.

[Argamon et al., 2005] Argamon, S., Dhawle, S., Koppel, M., and Pennebaker, J. W. (2005). Lexical predictors of personality type. In *Proceedings of the Joint Annual Meeting of the Interface and the Classification Society of North America* St. Louis, Missouri, 8 –12 June 2005, pages 1–15.

[Argamon et al., 2007] Argamon, S., Koppel, M., Pennebaker, J. W., and Schler, J. (2007). Mining the blogosphere: Age, gender and the varieties of self-expression. *First Monday (Online Peer-reviewed Journal)*, 12(9).

[Atserias et al., 2008] Atserias, J., Zaragoza, H., Ciaramita, M., and Attardi, G. (2008). Semantically annotated snapshot of the English Wikipedia. In Calzolari, N., editor, *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC),* Marrakech, Morocco, 26 May – 1 June 2008, Marrakech, Morocco. European Language Resources Association (ELRA).

[Baccianella et al., 2010] Baccianella, S., Esuli, A., and Sebastiani, F. (2010). SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC),* La Valetta, Malta, 17–23 May 2010, pages 2200–2204. European Language Resources Association (ELRA).

[Bachrach et al., 2012] Bachrach, Y., Graepel, T., Kasneci, G., Kosinski, M., and Van Gael, J. (2012). Crowd IQ: Aggregating opinions to boost performance. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 535–542. International Foundation for Autonomous Agents and Multiagent Systems.

[Bamman et al., 2014] Bamman, D., Eisenstein, J., and Schnoebelen, T. (2014). Gender identity and lexical variation in social media. *Journal of Sociolinguistics*, 18(2):135–160.

[Banerjee and Pedersen, 2002] Banerjee, S. and Pedersen, T. (2002). An Adapted Lesk Algorithm for Word Sense Disambiguation using WordNet. In *Computational Linguistics and Intelligent Text*, pages 136–145. Springer Berlin Heidelberg.

[Barocas and Selbst, 2016] Barocas, S. and Selbst, A. D. (2016). Big Data's Disparate Impact. *Social Science Research Network Working Paper Series (Online), http://ssrn.com/paper=2477899*.

[Baron, 1991] Baron, J. (1991). Beliefs about thinking. *Informal reasoning and education*, pages 169–186.

[Baron, 1993] Baron, J. (1993). Why teach thinking?-an essay. *Applied Psychology*, 42(3):191–214.

[Bastien et al., 2012] Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I. J., Bergeron, A., Bouchard, N., and Bengio, Y. (2012). Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop.

[Batchkarov et al., 2016] Batchkarov, M., Kober, T., Reffin, J., Weeds, J., and Weir, D. (2016). A critique of word similarity as a method for evaluating distributional semantic models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* Berlin, Germany 7–12 August 2016, page 7.

[Bengio et al., 2003] Bengio, Y., Ducharme, R., Vincent, P., and Jauvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, 3(Feb):1137–1155.

[Bian et al., 2014] Bian, J., Gao, B., and Liu, T.-Y. (2014). Knowledge-powered deep learning for word embedding. In *Machine Learning and Knowledge Discovery in Databases*, pages 132–148. Springer.

[Biel and Gatica-Perez, 2013] Biel, J.-I. and Gatica-Perez, D. (2013). The YouTube lens: Crowdsourced personality impressions and audiovisual analysis of vlogs. *Multimedia, IEEE Transactions on*, 15(1):41–55.

[Biemann and Riedl, 2013] Biemann, C. and Riedl, M. (2013). Text: Now in 2D! a framework for lexical expansion with contextual similarity. *Journal of Language Modelling*, 1(1):55–95.

[Bies et al., 2012] Bies, A., Mott, J., Warner, C., and Kulick, S. (2012). English web treebank. *Linguistic Data Consortium, Philadelphia, PA*.

[Biyani et al., 2014] Biyani, P., Bhatia, S., Caragea, C., and Mitra, P. (2014). Using non-lexical features for identifying factual and opinionative threads in online forums. *Knowledge-Based Systems*, 69:170–178.

[Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022.

[Blitzer et al., 2007] Blitzer, J., Dredze, M., and Pereira, F. (2007). Biographies, Bollywood, boomboxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL),* Prague, Czech Republic, 23–30 June 2007, ACL, pages 440–447.

[Boleda et al., 2012] Boleda, G., Padó, S., and Utt, J. (2012). Regular polysemy: A distributional model. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 151–160, Montréal, Canada. Association for Computational Linguistics.

[Bordes et al., 2012] Bordes, A., Glorot, X., Weston, J., and Bengio, Y. (2012). Joint learning of words and meaning representations for open-text semantic parsing. In *International Conference on Artificial Intelligence and Statistics*, pages 127–135.

[Bordes et al., 2011] Bordes, A., Weston, J., Collobert, R., and Bengio, Y. (2011). Learning structured embeddings of knowledge bases. In *Proceedings of the 25th Conference on the Advancement of Artificial Intelligence (AAAI),* San Francisco, Cal., 7–11 August 2011, pages 301–306.

[Bovi et al., 2015] Bovi, C. D., Anke, L. E., and Navigli, R. (2015). Knowledge base unification via sense embeddings and disambiguation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* Lisbon, Portugal, 17–21 September 2015, pages 726–736.

[Bowling, 2005] Bowling, A. (2005). Mode of questionnaire administration can have serious effects on data quality. *Journal of public health*, 27(3):281–291.

[Bruni et al., 2014] Bruni, E., Tran, N.-K., and Baroni, M. (2014). Multimodal distributional semantics. *Journal of Artificial Intelligence Research (JAIR)*, 49(1-47).

[Bubenhofer, 2009] Bubenhofer, N. (2009). *Sprachgebrauchsmuster: Korpuslinguistik als Methode der Diskurs- und Kulturanalyse*. Walter de Gruyter.

[Burger et al., 2011] Burger, D. J., Henderson, J., Kim, G., and Zarrella, G. (2011). Discriminating gender on Twitter. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP),* Edinburgh, Scotland, 27–29 July 2011, pages 1301–1309.

[Byrne, 1961] Byrne, D. (1961). Interpersonal attraction and attitude similarity. *The Journal of Abnormal and Social Psychology*, 62(3):713–730.

[Cambria et al., 2013] Cambria, E., Schuller, B., Xia, Y., and Havasi, C. (2013). New avenues in opinion mining and sentiment analysis. *IEEE Intelligent Systems*, 28(2):15–21.

[Carenini et al., 2013] Carenini, G., Cheung, J. C. K., and Pauls, A. (2013). Multi-document summarization of evaluative text. *Computational Intelligence*, 29(4):545–576.

[Carpenter et al., 2016] Carpenter, J., Preoţiuc-Pietro, D., Flekova, L., Giorgi, S., Hagan, C., Kern, M., Buffone, A., Ungar, L., and Seligman, M. (2016). Real men don't say 'cute': Using automatic language analysis to isolate inaccurate aspects of stereotypes. *Social Psychological and Personality Science*, pages 1–13.

[Carpuat and Wu, 2005] Carpuat, M. and Wu, D. (2005). Word sense disambiguation vs. statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL),* Ann Arbor, Mich., 25–30 June 2005, pages 387–394. Association for Computational Linguistics.

[Carpuat and Wu, 2007] Carpuat, M. and Wu, D. (2007). Improving statistical machine translation using word sense disambiguation. In *Proceedings of the 11th Conference on Computational Natural Language Learning (CoNLL) conference held at EMNLP 2007,* Prague, Czech Republic, 28–30 June 2007, volume 7, pages 61–72.

[Caruana, 1993] Caruana, R. (1993). Multitask learning: A knowledge-based source of inductive bias. In *Proceedings of the 10th International Conference on Machine Learning,* Amherst, Mass., 27–29 June 1993, pages 41–48.

[Celli et al., 2014] Celli, F., Lepri, B., Biel, J.-I., Gatica-Perez, D., Riccardi, G., and Pianesi, F. (2014). The workshop on computational personality recognition 2014. In *Proceedings of the ACM International Conference on Multimedia*, pages 1245–1246. ACM.

[Celli et al., 2013] Celli, F., Pianesi, F., Stillwell, D., and Kosinski, M. (2013). Workshop on computational personality recognition (shared task). In *Proceedings of the workshop on computational personality recognition (WCPR13) at the 7th international AAAI conference on weblogs and social media (Online proceedings),* Boston, Mass., 11 July 2013.

[Chan et al., 2007] Chan, Y. S., Ng, H. T., and Chiang, D. (2007). Word sense disambiguation improves statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL),* Prague, Czech Republic, 23–30 June 2007, volume 45, page 33.

[Chang et al., 2009] Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J. L., and Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. In *Advances in Neural Information Processing Systems 22*, pages 288–296.

[Chartrand and Bargh, 1999] Chartrand, T. L. and Bargh, J. A. (1999). The chameleon effect: the perception–behavior link and social interaction. *Journal of personality and social psychology*, 76(6):893–905.

[Chen and Manning, 2014] Chen, D. and Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* Doha, Qatar, 25–29 October 2014, pages 740–750.

[Chen et al., 2011] Chen, M., Jin, X., and Shen, D. (2011). Short text classification improved by learning multi-granularity topics. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI),* Barcelona, Spain, 19–22 July 2011, pages 1776–1781.

[Chen et al., 2014] Chen, X., Liu, Z., and Sun, M. (2014). A unified model for word sense representation and disambiguation. In *Proceedings of the 2014 Conference on Empirical Methods on Natural Language Processing*, pages 1025–1035.

[Chesney, 2006] Chesney, T. (2006). An empirical examination of Wikipedia's credibility. *First Monday (Online Peer-reviewed Journal)*, 11(11).

[Cho et al., 2014] Cho, K., Van Merriënboer, B., Bahdanau, D., and Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

[Choi and Cardie, 2009] Choi, Y. and Cardie, C. (2009). Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP),* Singapore, 6–7 August 2009, pages 590–598.

[Ciaramita and Altun, 2006] Ciaramita, M. and Altun, Y. (2006). Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In *Proceedings of the 2006 Conference on Empirical Methods on Natural Language Processing*, pages 594–602. Association for Computational Linguistics.

[Ciaramita and Johnson, 2003] Ciaramita, M. and Johnson, M. (2003). Supersense tagging of unknown nouns in WordNet. In *Proceedings of the 2003 Conference on Empirical Methods on Natural Language Processing*, pages 168–175. Association for Computational Linguistics.

[Cohen, 1968] Cohen, J. (1968). Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213–220.

[Collobert and Weston, 2008] Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning,* Helsinki, Finland, 5–9 July 2008, pages 160–167. ACM.

[Collobert et al., 2011] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

[Coltheart, 1981] Coltheart, M. (1981). The MRC psycholinguistic database. *The Quarterly Journal of Experimental Psychology*, 33(4):497–505.

[Condorcet, 1785] Condorcet, J.-A.-N. d. C. (1785). Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix. *L'imprimerie royale*.

[Cook and Stevenson, 2010] Cook, P. and Stevenson, S. (2010). Automatically identifying changes in the semantic orientation of words. In *Proceedings of the 7th International Conference on*

Language Resources and Evaluation (LREC), La Valetta, Malta, 17–23 May 2010, pages 129–149. European Language Resources Association (ELRA).

[Copestake and Briscoe, 1995] Copestake, A. and Briscoe, T. (1995). Semi-productive polysemy and sense extension. *Journal of semantics*, 12(1):15–67.

[Corney et al., 2002] Corney, M., de Vel, O., Anderson, A., and Mohay, G. (2002). Gender-preferential text mining of e-mail discourse. In *Proceedings of 18th Annual Computer Security Applications Conference*, pages 282–290. IEEE.

[Costa and McCrae, 2008] Costa, P. T. and McCrae, R. R. (2008). The revised NEO personality inventory (NEO-PI-R). *The SAGE handbook of personality theory and assessment*, 2:179–198.

[Culotta et al., 2015] Culotta, A., Kumar, N. R., and Cutler, J. (2015). Predicting the demographics of Twitter users from website traffic data. In *Proceedings of the 29th Conference on the Advancement of Artificial Intelligence (AAAI),* Austin, Texas, 25–30 January 2015, pages 72–78.

[Dahl et al., 2012] Dahl, G. E., Yu, D., Deng, L., and Acero, A. (2012). Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(1):30–42.

[Danescu-Niculescu-Mizil et al., 2009] Danescu-Niculescu-Mizil, C., Lee, L., and Ducott, R. (2009). Without a 'doubt'?: unsupervised discovery of downward-entailing operators. In *Proceedings of the 2009 Conference of the North American Chapter of the Association for Computational Linguistics*, ACL, pages 137–145.

[Daumé III et al., 2009] Daumé III, H., Langford, J., and Marcu, D. (2009). Search-based structured prediction. *Machine learning*, 75(3):297–325.

[Daxenberger et al., 2014] Daxenberger, J., Ferschke, O., Gurevych, I., and Zesch, T. (2014). DKPro TC: A Java-based framework for supervised learning experiments on textual data. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics - System Demonstrations* Baltimore, MD, 23–25 June 2014, pages 61–66.

[De Marneffe and Manning, 2008] De Marneffe, M.-C. and Manning, C. D. (2008). Stanford typed dependencies manual. Technical report, Stanford University.

[De Marneffe et al., 2010] De Marneffe, M.-C., Manning, C. D., and Potts, C. (2010). Was it good? It was provocative. Learning the meaning of scalar adjectives. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL),* Uppsala, Sweden, 11–16 July 2010, ACL, pages 167–176.

[Deerwester et al., 1990] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391.

[Dewaele and Furnham, 1999] Dewaele, J.-M. and Furnham, A. (1999). Extraversion: The unloved variable in applied linguistic research. *Language Learning*, 49(3):509–544.

[Dong et al., 2015a] Dong, L., Wei, F., Liu, S., Zhou, M., and Xu, K. (2015a). A statistical parsing framework for sentiment classification. *Computational Linguistics*, 41:293–336.

[Dong et al., 2015b] Dong, L., Wei, F., Zhou, M., and Xu, K. (2015b). Question answering over freebase with multi-column convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics,* Beijing, China, 26–31 July 2015, pages 260–269.

[Dragut et al., 2012] Dragut, E., Wang, H., Yu, C., Sistla, P., and Meng, W. (2012). Polarity consistency checking for sentiment dictionaries. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL),* Jeju Island, Korea, 8–14 July 2012, pages 997–1005.

[Dyer et al., 2015] Dyer, C., Ballesteros, M., Ling, W., Matthews, A., and Smith, N. A. (2015). Transition-based dependency parsing with stack long short-term memory. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics,* Beijing, China, 26–31 July 2015, pages 334–343.

[Eger and Mehler, 2016] Eger, S. and Mehler, A. (2016). On the linearity of semantic change: Investigating meaning variation via dynamic graph models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* Berlin, Germany 7–12 August 2016, pages 52–60.

[Elman, 1990] Elman, J. L. (1990). Finding structure in time. *Cognitive science*, 14(2):179–211.

[Estival et al., 2007] Estival, D., Gaustad, T., Pham, S. B., Radford, W., and Hutchinson, B. (2007). Author profiling for english emails. In *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics*, pages 262–272.

[Esuli and Sebastiani, 2006] Esuli, A. and Sebastiani, F. (2006). SentiWordNet: a publicly available lexical resource for opinion mining. In *Proceedings of the 5th International Conference on Language Resources and Evaluation,* Genoa, Italy, 24-26 May, 2006, pages 417–422. European Language Resources Association (ELRA).

[Ettinger et al., 2016] Ettinger, A., Resnik, P., and Carpuat, M. (2016). Retrofitting sense-specific word vectors using parallel text. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* San Diego, CA, 12 –– 17 June 2016, pages 1378–1383.

[Farnadi et al., 2013] Farnadi, G., Zoghbi, S., Moens, M.-F., and De Cock, M. (2013). Recognising personality traits using facebook status updates. In *Proceedings of the workshop on computational personality recognition (WCPR13) at the 7th international AAAI conference on weblogs and social media (Online proceedings),* Boston, Mass., 11 July 2013. AAAI.

[Faruqui et al., 2014] Faruqui, M., Dodge, J., Jauhar, S. K., Dyer, C., Hovy, E., and Smith, N. A. (2014). Retrofitting word vectors to semantic lexicons. *arXiv preprint arXiv:1411.4166*.

[Faruqui and Dyer, 2014] Faruqui, M. and Dyer, C. (2014). Improving vector space word representations using multilingual correlation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL),* Gothenburg, Sweden, 26–30 April 2014, pages 462–471. Association for Computational Linguistics.

[Faruqui et al., 2016] Faruqui, M., Tsvetkov, Y., Rastogi, P., and Dyer, C. (2016). Problems with evaluation of word embeddings using word similarity tasks. *arXiv preprint*, arXiv:1605.02276.

[Fellbaum, 1990] Fellbaum, C. (1990). English verbs as a semantic net. *International Journal of Lexicography*, 3(4):278–301.

[Fellbaum, 1998] Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Wiley Online Library.

[Feng et al., 2013] Feng, S., Kang, J. S., Kuznetsova, P., and Choi, Y. (2013). Connotation lexicon: A dash of sentiment beneath the surface meaning. In *Proceedings of the 51st Conference of the Association for Computational Linguistics,* Sofia, Bulgaria 4–9 August 2013, pages 1774–1784.

[Ferrucci and Lally, 2004] Ferrucci, D. and Lally, A. (2004). UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. *Natural Language Engineering*, 10(3-4):327–348.

[Finkel et al., 2005] Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL),* Ann Arbor, Mich., 25–30 June 2005, pages 363–370. Association for Computational Linguistics.

[Flekova et al., 2016a] Flekova, L., Carpenter, J., Giorgi, S., Ungar, L., and Preoţiuc-Pietro, D. (2016a). Analysing biases in human perception of user age and gender from text. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* Berlin, Germany 7–12 August 2016, pages 843–854. Association for Computational Linguistics.

[Flekova et al., 2014a] Flekova, L., Ferschke, O., and Gurevych, I. (2014a). UKPDIPF: A lexical-semantic approach to sentiment polarity prediction in Twitter data. In Nakov, P. and Zesch, T., editors, *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014),* Dublin, Ireland, 23–24 August 2014, pages 704–710. Association for Computational Linguistics and Dublin City University.

[Flekova et al., 2014b] Flekova, L., Ferschke, O., and Gurevych, I. (2014b). What makes a good biography? multidimensional quality analysis based on wikipedia article feedback data. In *Proceedings of the 23rd International World Wide Web Conference (WWW), Seoul, Korea, 7 – 11 April, 2014*, pages 855–866. International World Wide Web Conferences Steering Committee.

[Flekova et al., 2015a] Flekova, L., Giorgi, S., Carpenter, J., Ungar, L., and Preotiuc-Pietro, D. (2015a). Analyzing crowdsourced assessment of user traits through twitter posts. In *Third AAAI Conference on Human Computation and Crowdsourcing (Online proceedings - short papers)* , San Diego, CA 8–11 November 2015.

[Flekova and Gurevych, 2013] Flekova, L. and Gurevych, I. (2013). Can We Hide in the Web? Large Scale Simultaneous Age and Gender Author Profiling in Social Media - Notebook for PAN at CLEF 2013. In Forner, P., Navigli, R., and Tufis, D., editors, *CLEF 2013 Labs and Workshops - Online Working Notes*, Padua, Italy. PROMISE.

[Flekova and Gurevych, 2015] Flekova, L. and Gurevych, I. (2015). Personality profiling of fictional characters using sense-level links between lexical resources. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* Lisbon, Portugal, 17–21 September 2015, pages 1805–1816, Lisbon, Portugal. Association for Computational Linguistics.

[Flekova and Gurevych, 2016] Flekova, L. and Gurevych, I. (2016). Supersense embeddings: A unified model for supersense interpretation, prediction and utilization. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* Berlin, Germany 7–12 August 2016, pages 2029–2041. Association for Computational Linguistics.

[Flekova et al., 2016b] Flekova, L., Preoţiuc-Pietro, D., and Ungar, L. (2016b). Exploring stylistic variation with age and income on twitter. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* Berlin, Germany 7–12 August 2016, pages 313–319. Association for Computational Linguistics.

[Flekova et al., 2015b] Flekova, L., Ruppert, E., and Preotiuc-Pietro, D. (2015b). Analysing domain suitability of a sentiment lexicon by identifying distributionally bipolar words. In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 77–84. Association for Computational Linguistics.

[Gale et al., 1992] Gale, W. A., Church, K. W., and Yarowsky, D. (1992). One sense per discourse. In *Proceedings of the workshop on Speech and Natural Language*, pages 233–237. Association for Computational Linguistics.

[Gershman et al., 2014] Gershman, A., Tsvetkov, Y., Boytsov, L., Nyberg, E., and Dyer, C. (2014). Metaphor detection with cross-lingual model transfer. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics,* Baltimore, MD, 23–25 June 2014, pages 248–258.

[Gill and Oberlander, 2002] Gill, A. J. and Oberlander, J. (2002). Taking care of the linguistic features of extraversion. In *Proceedings of the 24th Annual Conference of the Cognitive Science Society*, pages 363–368.

[Giménez and Màrquez, 2007] Giménez, J. and Màrquez, L. (2007). Context-aware discriminative phrase selection for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation,* Prague, Czech Republic, pages 159–166. Association for Computational Linguistics.

[Gleser et al., 1959] Gleser, G. C., Gottschalk, L. A., and John, W. (1959). The relationship of sex and intelligence to choice of words: A normative study of verbal behavior. *Journal of Clinical Psychology*, 15(2):182–191.

[Glorot et al., 2011] Glorot, X., Bordes, A., and Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics* Reykjavik, Iceland, 22–25 April 2014, volume 15, pages 315–323.

[Go et al., 2009] Go, A., Bhayani, R., and Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1.

[Goikoetxea et al., 2015] Goikoetxea, J., Soroa, A., Agirre, E., and Donostia, B. C. (2015). Random walks and neural network language models on knowledge bases. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies*, pages 1434–1439.

[Goldberg, 1990] Goldberg, L. R. (1990). An alternative description of personality: the Big-Five factor structure. *Journal of personality and social psychology*, 59(6):1216–1235.

[Goldberg et al., 2006] Goldberg, L. R., Johnson, J. A., Eber, H. W., Hogan, R., Ashton, M. C., Cloninger, C. R., and Gough, H. G. (2006). The international personality item pool and the future of public-domain personality measures. *Journal of Research in personality*, 40(1):84–96.

[Goldberg, 2016] Goldberg, Y. (2016). A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research (JAIR)*, 57:345–420.

[Goller and Kuchler, 1996] Goller, C. and Kuchler, A. (1996). Learning task-dependent distributed representations by backpropagation through structure. In *Neural Networks, 1996., IEEE International Conference on*, volume 1, pages 347–352. IEEE.

[Gonzalo et al., 1998] Gonzalo, J., Verdejo, F., Chugur, I., and Cigarran, J. (1998). Indexing with WordNet synsets can improve text retrieval. In *Proceedings of the COLING/ACL '98 Workshop on Usage of WordNet for NLP.*, pages 38–44.

[Goodfellow et al., 2016] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT Press.

[Gosling et al., 2003] Gosling, S. D., Rentfrow, P. J., and Swann, W. B. (2003). A very brief measure of the Big-Five personality domains. *Journal of Research in personality*, 37(6):504–528.

[Gurevych et al., 2009] Gurevych, I., Bernhard, D., Ignatova, K., and Toprak, C. (2009). Educational question answering based on social media content. In *Proceedings of the 14th International Conference on Artificial Intelligence in Education. Building learning systems that care: From knowledge Representation to affective modelling.*, pages 133–140.

[Gurevych et al., 2012] Gurevych, I., Eckle-Kohler, J., Hartmann, S., Matuschek, M., Meyer, C. M., and Wirth, C. (2012). Uby: A large-scale unified lexical-semantic resource based on LMF. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL),* Avignon, France, 23–27 April 2012, pages 580–590.

[Gurevych et al., 2016] Gurevych, I., Eckle-Kohler, J., and Matuschek, M. (2016). Linked lexical knowledge bases: Foundations and applications. *Synthesis Lectures on Human Language Technologies*, 9(3):1–146.

[Gurevych et al., 2007] Gurevych, I., Mühlhäuser, M., Müller, C., Steimle, J., Weimer, M., and Zesch, T. (2007). Darmstadt Knowledge Processing Repository Based on UIMA. In *Proceedings of the First Workshop on Unstructured Information Management Architecture at Biannual Conference of the Society for Computational Linguistics and Language Technology*, pages 1–3, Tübingen, Germany.

[Guyon and Elisseeff, 2003] Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182.

[Hall, 1999] Hall, M. A. (1999). *Correlation-based feature selection for machine learning*. PhD thesis, The University of Waikato.

[Hamilton et al., 2016] Hamilton, W. L., Leskovec, J., and Jurafsky, D. (2016). Diachronic word embeddings reveal statistical laws of semantic change. *arXiv preprint arXiv:1605.09096*.

[Haran et al., 2013] Haran, U., Ritov, I., and Mellers, B. A. (2013). The role of actively open-minded thinking in information acquisition, accuracy, and calibration. *Judgment and Decision Making*, 8(3):188.

[Harris, 1954] Harris, Z. S. (1954). Distributional structure. *Word*, 10(2-3):146–162.

[He et al., 2013] He, H., Barbosa, D., and Kondrak, G. (2013). Identification of speakers in novels. In *Proceedings of the 51st Conference of the Association for Computational Linguistics,* Sofia, Bulgaria 4–9 August 2013, pages 1312–1320.

[Heilman, 2011] Heilman, M. (2011). *Automatic factual question generation from text*. PhD thesis, Carnegie Mellon University.

[Heylighen and Dewaele, 2002] Heylighen, F. and Dewaele, J.-M. (2002). Variation in the contextuality of language: An empirical measure. *Foundations of Science*, 7(3):293–340.

[Hinton et al., 2012a] Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., et al. (2012a). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97.

[Hinton, 1986] Hinton, G. E. (1986). Learning distributed representations of concepts. In *Proceedings of the 8th annual Conference of the Cognitive Science Society*, volume 1, pages 46–61. Amherst, MA.

[Hinton et al., 2012b] Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012b). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

[Hofmann, 1999] Hofmann, T. (1999). Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 289–296. Morgan Kaufmann Publishers Inc.

[Hovy and Sogaard, 2015] Hovy, D. and Sogaard, A. (2015). Tagging performance correlates with author age. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics,* Beijing, China, 26–31 July 2015, ACL, pages 483–488.

[Hovy and Spruit, 2016] Hovy, D. and Spruit, S. L. (2016). The social impact of natural language processing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* Berlin, Germany 7–12 August 2016, volume 2, pages 591–598.

[Hu and Liu, 2004] Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD International conference on Knowledge Discovery and Data Mining*, KDD, pages 168–177.

[Huang et al., 2015] Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

[Iacobacci et al., 2015] Iacobacci, I., Pilehvar, M. T., and Navigli, R. (2015). Sensembed: learning sense embeddings for word and relational similarity. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics,* Beijing, China, 26–31 July 2015, pages 95–105.

[Ide and Wilks, 2007] Ide, N. and Wilks, Y. (2007). Making sense about sense. In *Word sense disambiguation*, pages 47–73. Springer.

[Ikeda et al., 2008] Ikeda, D., Takamura, H., Ratinov, L.-A., and Okumura, M. (2008). Learning to shift the polarity of words for sentiment classification. In *Proceedings of the International Joint Conference on Natural Language Processing*, pages 296–303.

[Iosif and Mishra, 2014] Iosif, E. and Mishra, T. (2014). From speaker identification to affective analysis: A multi-step system for analyzing children stories. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL),* Gothenburg, Sweden, 26–30 April 2014, pages 40–49.

[Ipeirotis et al., 2010] Ipeirotis, P. G., Provost, F., and Wang, J. (2010). Quality management on Amazon Mechanical Turk. In *Proceedings of the ACM SIGKDD workshop on human computation*, pages 64–67. ACM.

[Izquierdo et al., 2009] Izquierdo, R., Suárez, A., and Rigau, G. (2009). An empirical study on class-based word sense disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL),* Athens, Greece, 30 March – 3 April 2009, pages 389–397. Association for Computational Linguistics.

[Jauhar et al., 2015] Jauhar, S. K., Dyer, C., and Hovy, E. H. (2015). Ontologically grounded multi-sense representation learning for semantic vector space models. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* Denver, Colorado, 31 May -– 5 June 2015, pages 683–693.

[Jiang and Argamon, 2008] Jiang, M. and Argamon, S. (2008). Exploiting subjectivity analysis in blogs to improve political leaning categorization. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* Singapore, 20–24 July 2008, pages 725–726. ACM.

[Joachims, 1998] Joachims, T. (1998). Text categorization with support vector machines: learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning,* Chemnitz, Germany, 21–23 April 1998, pages 137–142.

[Joachims, 1999] Joachims, T. (1999). Svmlight: Support vector machine. *SVM-Light Support Vector Machine http://svmlight. joachims. org/, University of Dortmund*, 19(4).

[Johannsen et al., 2014] Johannsen, A., Hovy, D., Alonso, H. M., Plank, B., and Søgaard, A. (2014). More or less supervised supersense tagging of Twitter. *Proceedings of the 3rd Joint Conference on Lexical and Computational Semantics*, pages 1–11.

[Johannsen et al., 2015] Johannsen, A., Hovy, D., and Søgaard, A. (2015). Cross-lingual syntactic variation over age and gender. In *Proceedings of the 19th Conference on Computational Language Learning*, pages 103–112.

[John et al., 1991] John, O. P., Donahue, E. M., and Kentle, R. L. (1991). The Big Five inventory—versions 4a and 54.

[John and Srivastava, 1999] John, O. P. and Srivastava, S. (1999). The Big Five trait taxonomy: History, measurement, and theoretical perspectives. *Handbook of personality: Theory and research*, 2(1999):102–138.

[Johnson and Zhang, 2014] Johnson, R. and Zhang, T. (2014). Effective use of word order for text categorization with convolutional neural networks. *arXiv preprint arXiv:1412.1058*.

[Jorgensen, 1990] Jorgensen, J. C. (1990). The psychological reality of word senses. *Journal of psycholinguistic research*, 19(3):167–190.

[Jurafsky and Martin, 2009] Jurafsky, D. and Martin, J. H. (2009). *Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition*. Pearson / Prentice Hall, 2nd ed edition.

[Kalchbrenner et al., 2014] Kalchbrenner, N., Grefenstette, E., and Blunsom, P. (2014). A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics,* Baltimore, MD, 23–25 June 2014, pages 655–665.

[Kamar et al., 2012] Kamar, E., Hacker, S., and Horvitz, E. (2012). Combining human and machine intelligence in large-scale crowdsourcing. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 467–474. International Foundation for Autonomous Agents and Multiagent Systems.

[Kaufman and Libby, 2012] Kaufman, G. F. and Libby, L. K. (2012). Changing beliefs and behavior through experience-taking. *Journal of personality and social psychology*, 103(1):1–19.

[Kazai et al., 2011] Kazai, G., Kamps, J., and Milic-Frayling, N. (2011). Worker types and personality traits in crowdsourcing relevance labels. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1941–1944. ACM.

[Kazai et al., 2012] Kazai, G., Kamps, J., and Milic-Frayling, N. (2012). The face of quality in crowdsourcing relevance labels: Demographics, personality and labeling accuracy. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2583–2586. ACM.

[Kehagias et al., 2003] Kehagias, A., Petridis, V., Kaburlasos, V. G., and Fragkou, P. (2003). A comparison of word-and sense-based text categorization using several classification algorithms. *Journal of Intelligent Information Systems*, 21(3):227–247.

[Kennedy and Inkpen, 2006] Kennedy, A. and Inkpen, D. (2006). Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, 22(2):110–125.

[Kilgarriff, 1997] Kilgarriff, A. (1997). I don't believe in word senses. *Computers and the Humanities*, 31(2):91–113.

[Kilgarriff and Rosenzweig, 2000] Kilgarriff, A. and Rosenzweig, J. (2000). English Senseval: Report and Results. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation,* Athens, Greece, May, 2000, volume 6, page 2. European Language Resources Association (ELRA).

[Kilgarriff et al., 2004] Kilgarriff, A., Rychly, P., Smrz, P., and Tugwell, D. (2004). Itri-04-08 the sketch engine. *Information Technology*, 105.

[Kim and Hovy, 2004] Kim, S.-M. and Hovy, E. (2004). Determining the sentiment of opinions. In *Proceedings of the 20th International Conference on Computational Linguistics,* Geneva, Switzerland, 23 August – 27 August 2004, pages 1367–1378.

[Kim, 2014] Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* Doha, Qatar, 25–29 October 2014, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

[Kipper et al., 2008] Kipper, K., Korhonen, A., Ryant, N., and Palmer, M. (2008). A Large-scale Classification of English Verbs. *Language Resources and Evaluation*, 42(1):21–40.

[Kipper-Schuler, 2005] Kipper-Schuler, K. (2005). *VerbNet: A broad-coverage, comprehensive verb lexicon*. PhD thesis, University of Pennsylvania.

[Klementiev et al., 2012] Klementiev, A., Titov, I., and Bhattarai, B. (2012). Inducing crosslingual distributed representations of words. In *Proceedings of the 24th International Conference on Computational Linguistics,* Mumbay, India, December 2012, pages 1459–1474.

[Klenner et al., 2014] Klenner, M., Amsler, M., and Hollenstein, N. (2014). Inducing domain-specific noun polarity guided by domain-independent polarity preferences of adjectives. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics,* Baltimore, MD, 23–25 June 2014, pages 18–23.

[Kohonen and Somervuo, 1998] Kohonen, T. and Somervuo, P. (1998). Self-organizing maps of symbol strings. *Neurocomputing*, 21(1):19–30.

[Kosinski et al., 2014] Kosinski, M., Bachrach, Y., Kohli, P., Stillwell, D., and Graepel, T. (2014). Manifestations of user personality in website choice and behaviour on online social networks. *Machine learning*, 95(3):357–380.

[Kosinski et al., 2013] Kosinski, M., Stillwell, D., and Graepel, T. (2013). Private traits and attributes are predictable from digital records of human behavior. *Proceedings of the National Academy of Sciences*, 110(15):5802–5805.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C., Bottou, L., and Weinberger, K., editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc.

[Krovetz, 2002] Krovetz, B. (2002). On the importance of word sense disambiguation for information retrieval. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation,* Las Palmas, Canary Islands, Spain, 29-31 May, 2002. European Language Resources Association (ELRA).

[Kulkarni et al., 2015] Kulkarni, V., Al-Rfou, R., Perozzi, B., and Skiena, S. (2015). Statistically significant detection of linguistic change. In *Proceedings of the 24th International Conference on World Wide Web*, pages 625–635. ACM.

[Ladha, 1995] Ladha, K. K. (1995). Information pooling through majority-rule voting: Condorcet's jury theorem with correlated votes. *Journal of Economic Behavior & Organization*, 26(3):353–372.

[Lakoff and Johnson, 2008] Lakoff, G. and Johnson, M. (2008). *Metaphors we live by*. University of Chicago press.

[Landis and Koch, 1977] Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, pages 159–174.

[Laparra and Rigau, 2013] Laparra, E. and Rigau, G. (2013). Impar: A deterministic algorithm for implicit semantic role labelling. In *Proceedings of the 51st Conference of the Association for Computational Linguistics,* Sofia, Bulgaria 4–9 August 2013, pages 1180–1189.

[Larochelle et al., 2008] Larochelle, H., Erhan, D., and Bengio, Y. (2008). Zero-data Learning of New Tasks. In *Proceedings of the 23rd Conference on the Advancement of Artificial Intelligence (AAAI),* Chicago, Ill., 13–17 July 2008, pages 646–651. AAAI Press.

[LeCun and Bengio, 1995] LeCun, Y. and Bengio, Y. (1995). Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361:255–258.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Orr, G., and Muller, K. (1998). Neural networks: Tricks of the trade. *Springer Lecture Notes in Computer Sciences*, 1524(5–50):6.

[Lepri et al., 2010] Lepri, B., Subramanian, R., Kalimeri, K., Staiano, J., Pianesi, F., and Sebe, N. (2010). Employing social gaze and speaking activity for automatic determination of the extraversion trait. In *International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction*, pages 7:1–7:8, New York, NY, USA. ACM.

[Lesk, 1986] Lesk, M. (1986). Automatic Sense Disambiguation using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In *Proceedings of the 5th Annual International Conference on Systems Documentation*, pages 24–26. ACM.

[Levin, 1993] Levin, B. (1993). *English Verb Classes and Alternations: a preliminary investigation*. University of Chicago Press, Chicago and London.

[Levy et al., 2015] Levy, O., Goldberg, Y., and Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

[Li et al., 2008a] Li, B., Liu, Y., Ram, A., Garcia, E. V., and Agichtein, E. (2008a). Exploring question subjectivity prediction in community qa. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* Singapore, 20–24 July 2008, pages 735–736. ACM.

[Li et al., 2014] Li, H., Zhao, B., and Fuxman, A. (2014). The wisdom of minority: discovering and targeting the right group of workers for crowdsourcing. In *Proceedings of the 23rd International World Wide Web Conference (WWW), Seoul, Korea, 7 – 11 April, 2014*, pages 165–176. ACM.

[Li and Jurafsky, 2015] Li, J. and Jurafsky, D. (2015). Do multi-sense embeddings improve natural language understanding? In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* Lisbon, Portugal, 17–21 September 2015, pages 1722–1732. Association for Computational Linguistics.

[Li et al., 2015] Li, J., Luong, T., Jurafsky, D., and Hovy, E. (2015). When are tree structures necessary for deep learning of representations? In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* Lisbon, Portugal, 17–21 September 2015, pages 2304–2314. Association for Computational Linguistics.

[Li et al., 2010] Li, S., Lee, S. Y. M., Chen, Y., Huang, C.-R., and Zhou, G. (2010). Sentiment classification and polarity shifting. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL),* Uppsala, Sweden, 11–16 July 2010, pages 635–643.

[Li et al., 2008b] Li, W., Sun, L., and Zhang, D.-K. (2008b). Text classification based on labeled LDA model. *Chinese Journal of Computers*, 31(4):620–627.

[Ly et al., 2011] Ly, D. K., Sugiyama, K., Lin, Z., and Kan, M.-Y. (2011). Product review summarization from a deeper perspective. In *Proceedings of the 11th annual international ACM/IEEE Joint Conference on Digital Libraries*, pages 311–314. ACM.

[MacDonald et al., 1994] MacDonald, D. A., Anderson, P. E., Tsagarakis, C. I., and Holland, C. J. (1994). Examination of the relationship between the Myers-Briggs Type Indicator and the NEO personality inventory. *Psychological Reports*, 74(1):339–344.

[Magnini and Cavaglia, 2000] Magnini, B. and Cavaglia, G. (2000). Integrating subject field codes into WordNet. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation,* Athens, Greece, May, 2000. European Language Resources Association (ELRA).

[Magnini et al., 2001] Magnini, B., Strapparava, C., Pezzulo, G., and Gliozzo, A. (2001). Using domain information for word sense disambiguation. In *The Proceedings of the Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 111–114. Association for Computational Linguistics.

[Mairesse et al., 2007] Mairesse, F., Walker, M. A., Mehl, M. R., and Moore, R. K. (2007). Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of Artificial Intelligence Research*, 30(1):457–500.

[Manning and Schütze, 1999] Manning, C. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. MIT Press.

[Manning et al., 2014] Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

[Mar et al., 2009] Mar, R. A., Oatley, K., and Peterson, J. B. (2009). Exploring the link between reading fiction and empathy: Ruling out individual differences and examining outcomes. *Communications Journal*, 34(4):407–428.

[Marcus et al., 1993] Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational linguistics*, 19(2):313–330.

[Marrero et al., 2009] Marrero, M., Sánchez-Cuadrado, S., Lara, J. M., and Andreadakis, G. (2009). Evaluation of named entity extraction systems. *Advances in Computational Linguistics, Research in Computing Science*, 41:47–58.

[Martens, 2010] Martens, J. (2010). Deep learning via hessian-free optimization. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 735–742.

[Mather and Carstensen, 2005] Mather, M. and Carstensen, L. L. (2005). Aging and motivated cognition: The positivity effect in attention and memory. *Trends in Cognitive Sciences*, 9(10):496–502.

[McCrae and Costa, 1987] McCrae, R. R. and Costa, P. T. (1987). Validation of the Five-Factor Model of personality across instruments and observers. *Journal of personality and social psychology*, 52(1):81–89.

[McCrae and Costa, 1989] McCrae, R. R. and Costa, P. T. (1989). Reinterpreting the Myers-Briggs type indicator from the perspective of the Five-Factor Model of personality. *Journal of personality*, 57(1):17–40.

[McCrae et al., 2012] McCrae, R. R., Gaines, J. F., and Wellington, M. A. (2012). The Five-Factor Model in fact and fiction. *Handbook of Psychology*, pages 65–91.

[McCulloch and Pitts, 1943] McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.

[McMillan et al., 1977] McMillan, J. R., Clifton, A. K., McGrath, D., and Gale, W. S. (1977). Women's language: Uncertainty or interpersonal sensitivity and emotionality? *Sex Roles*, 3(6):545–559.

[McNemar, 1947] McNemar, Q. (1947). Note on the Sampling Error of the Difference between Correlated Proportions or Percentages. *Psychometrika*, 12(2):153–157.

[Mehl et al., 2006] Mehl, M. R., Gosling, S. D., and Pennebaker, J. W. (2006). Personality in its natural habitat: manifestations and implicit folk theories of personality in daily life. *Journal of Personality and Social Psychology*, 90(5):862.

[Mellers et al., 2015] Mellers, B., Stone, E., Atanasov, P., Rohrbaugh, N., Metz, S. E., Ungar, L., Bisop, M., Horowitz, M., Merkle, E., and Tetlock, P. (2015). The psychology of intelligence

analysis: Drivers of prediction accuracy in world politics. *Journal of experimental psychology*, 21:1–14.

[Mihalcea and Csomai, 2005] Mihalcea, R. and Csomai, A. (2005). SenseLearner: Word sense disambiguation for all words in unrestricted text. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL), System Demonstrations,* Ann Arbor, Mich., 25–30 June 2005, pages 53–56. Association for Computational Linguistics.

[Mikolov et al., 2013a] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. In *Proceedings of Workshop Track at the International Conference on Learning Representations*, ICLR, pages 1–12.

[Mikolov et al., 2013b] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

[Mikolov et al., 2013c] Mikolov, T., Yih, W.-t., and Zweig, G. (2013c). Linguistic regularities in continuous space word representations. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 746–751.

[Miller, 1995] Miller, G. A. (1995). WordNet: a lexical database for english. *Communications of the Association for Computing Machinery (ACM)*, 38(11):39–41.

[Miller and Charles, 1991] Miller, G. A. and Charles, W. G. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28.

[Miller et al., 1994] Miller, G. A., Chodorow, M., Landes, S., Leacock, C., and Thomas, R. G. (1994). Using a semantic concordance for sense identification. In *Proceedings of the workshop on Human Language Technology*, pages 240–243. Association for Computational Linguistics.

[Miller et al., 2013] Miller, T., Erbs, N., Zorn, H.-P., Zesch, T., and Gurevych, I. (2013). DKPro WSD: A generalized UIMA-based framework for word sense disambiguation. In *Proceedings of the ACL 2013 System Demonstrations,* Sofia, Bulgaria, 4–9 August 2013, pages 37–42.

[Mitra et al., 2014] Mitra, S., Mitra, R., Riedl, M., Biemann, C., Mukherjee, A., and Goyal, P. (2014). That's sick dude!: Automatic identification of word sense change across different timescales. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics,* Baltimore, MD, 23–25 June 2014, ACL, pages 1020–1029.

[Mohammad, 2012] Mohammad, S. (2012). #emotional tweets. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics*, pages 246–255.

[Mohammad et al., 2009] Mohammad, S., Dunne, C., and Dorr, B. (2009). Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus. In *Pro-*

ceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP), Singapore, 6–7 August 2009, pages 599–608.

[Mohammad et al., 2013a] Mohammad, S., Kiritchenko, S., and Zhu, X. (2013a). NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. In *Second Joint Conference on Lexical and Computational Semantics*, volume 2 of **SeM*, pages 321–327.

[Mohammad and Kiritchenko, 2013] Mohammad, S. M. and Kiritchenko, S. (2013). Using nuances of emotion to identify personality. In *Proceedings of the workshop on computational personality recognition (WCPR13) at the 7th international AAAI conference on weblogs and social media (Online proceedings),* Boston, Mass., 11 July 2013.

[Mohammad et al., 2013b] Mohammad, S. M., Kiritchenko, S., and Zhu, X. (2013b). Nrc-canada: building the state-of-the-art in sentiment analysis of tweets. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013),* Atlanta, Georgia, 14–15 June 2013, pages 321–327.

[Mohammad and Turney, 2013] Mohammad, S. M. and Turney, P. D. (2013). Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3):436–465.

[Moilanen and Pulman, 2008] Moilanen, K. and Pulman, S. (2008). The good, the bad, and the unknown: Morphosyllabic sentiment tagging of unseen words. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies,* Columbus, Ohio, 15–20 June 2008, pages 109–112.

[Moldovan and Rus, 2001] Moldovan, D. I. and Rus, V. (2001). Explaining Answers with Extended WordNet. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics,* Toulouse, Toulouse, France.

[Montoya et al., 2008] Montoya, R. M., Horton, R. S., and Kirchner, J. (2008). Is actual similarity necessary for attraction? A meta-analysis of actual and perceived similarity. *Journal of Social and Personal Relationships*, 25(6):889–922.

[Moro et al., 2014] Moro, A., Raganato, A., and Navigli, R. (2014). Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

[Moschitti and Basili, 2004] Moschitti, A. and Basili, R. (2004). Complex linguistic features for text classification: A comprehensive study. In *Proceedings of the 26th European Conference on IR Research - Advances in Information Retrieval,* Sunderland, U.K., 5–7 April 2004, pages 181–196. Springer.

[Mukherjee and Liu, 2010] Mukherjee, A. and Liu, B. (2010). Improving gender classification of blog authors. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP),* Cambridge, Mass., 9–11 October 2010, pages 207–217. Association for Computational Linguistics.

[Mulac et al., 1990] Mulac, A., Studley, L., and Blau, S. (1990). The gender-linked language effect in primary and secondary students' impromptu essays. *Sex Roles*, 23(9-10):439–470.

[Myers et al., 1985] Myers, I. B., McCaulley, M. H., and Most, R. (1985). *Manual, a guide to the development and use of the Myers-Briggs type indicator*. Consulting Psychologists Press.

[Navigli, 2009] Navigli, R. (2009). Word Sense Disambiguation: A Survey. *ACM Computing Surveys (CSUR)*, 41(2):10:1–10:69.

[Navigli, 2012] Navigli, R. (2012). A quick tour of word sense disambiguation, induction and related approaches. In *SOFSEM 2012: Theory and practice of computer science*, pages 115–129. Springer.

[Navigli and Lapata, 2010] Navigli, R. and Lapata, M. (2010). An Experimental Study of Graph Connectivity for Unsupervised Word Sense Disambiguation. *IEEE transactions on pattern analysis and machine intelligence*, 32(4):678–692.

[Navigli et al., 2007] Navigli, R., Litkowski, K. C., and Hargraves, O. (2007). SemEval-2007 task 07: Coarse-grained English all-words task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 30–35. Association for Computational Linguistics.

[Navigli and Ponzetto, 2012] Navigli, R. and Ponzetto, S. P. (2012). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

[Navigli and Velardi, 2005] Navigli, R. and Velardi, P. (2005). Structural semantic interconnections: a knowledge-based approach to word sense disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(7):1075–1086.

[Neale et al., 2015] Neale, S., Gomes, L., and Branco, A. (2015). First steps in using word senses as contextual features in maxent models for machine translation. In *1st Deep Machine Translation Workshop*, page 64.

[Neuman and Cohen, 2014] Neuman, Y. and Cohen, Y. (2014). A vectorial semantics approach to personality assessment. *Nature Publishing Group, Scientific reports (Online)*, 4:4761.

[Newman et al., 2008] Newman, M. L., Groom, C. J., Handelman, L. D., and Pennebaker, J. W. (2008). Gender differences in language use: An analysis of 14,000 text samples. *Discourse Processes*, 45(3):211–236.

[Nguyen et al., 2013] Nguyen, D., Gravel, R., Trieschnigg, D., and Meder, T. (2013). 'How old do you think I am?'; A study of language and age in Twitter. In *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media*, ICWSM, pages 439–448.

[Nguyen et al., 2014] Nguyen, D.-P., Trieschnigg, R., Doğruöz, A., Gravel, R., Theune, M., Meder, T., and de Jong, F. (2014). Why gender and age prediction from tweets is hard: Lessons from a crowdsourcing experiment. In *Proceedings of the 25th International Conference on Computational Linguistics,* Dublin, Ireland, 23–29 August 2014, pages 1950–1961. Association for Computational Linguistics.

[Nguyen and Grishman, 2015] Nguyen, T. H. and Grishman, R. (2015). Event detection and domain adaptation with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics,* Beijing, China, 26–31 July 2015, volume 2, pages 365–371.

[Nowson, 2007] Nowson, S. (2007). Identifying more bloggers: Towards large scale personality classification of personal weblogs. In *In Proceedings of the International Conference on Weblogs and Social Media,* Boulder, Colorado, 26–28 March 2007.

[Nowson et al., 2005] Nowson, S., Oberlander, J., and Gill, A. J. (2005). Weblogs, genres and individual differences. In *Proceedings of the 27th Annual Conference of the Cognitive Science Society*, pages 1666–1671.

[Oberlander and Nowson, 2006] Oberlander, J. and Nowson, S. (2006). Whose thumb is it anyway? classifying author personality from weblog text. In *Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics (ACL),* Sydney, Australia, 17-–21 July 2006, volume Posters.

[O'Keefe et al., 2012] O'Keefe, T., Pareti, S., Curran, J. R., Koprinska, I., and Honnibal, M. (2012). A sequence labelling approach to quote attribution. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing and Natural Language Learning,* Jeju Island, Korea, 12–14 July 2012, pages 790–799. Association for Computational Linguistics.

[Palatucci et al., 2009] Palatucci, M., Pomerleau, D., Hinton, G. E., and Mitchell, T. M. (2009). Zero-shot learning with semantic output codes. In *Advances in Neural Information Processing Systems 22*, pages 1410–1418.

[Palmer, 2000] Palmer, M. (2000). Consistent criteria for sense distinctions. *Computers and the Humanities*, 34(1-2):217–222.

[Palmer et al., 2001] Palmer, M., Fellbaum, C., Cotton, S., Delfs, L., and Dang, H. T. (2001). English tasks: All-words and verb lexical sample. In *The Proceedings of the Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 21–24. Association for Computational Linguistics.

[Pang and Lee, 2004] Pang, B. and Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics,* Barcelona, Spain, 21–26 July 2004, ACL, pages 271–278.

[Pang and Lee, 2005] Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL),* Ann Arbor, Mich., 25–30 June 2005, pages 115–124. Association for Computational Linguistics.

[Pang and Lee, 2008] Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2:1–135.

[Pathak, 2014] Pathak, M. A. (2014). *Beginning Data Science with R*. Springer.

[Patwardhan and Pedersen, 2006] Patwardhan, S. and Pedersen, T. (2006). Using wordnet-based context vectors to estimate the semantic relatedness of concepts. In *Proceedings of the EACL 2006 Workshop on Making Sense of Sense: Bringing Computational Linguistics and Psycholinguistics together*, volume 1501, pages 1–8. Trento.

[Pennebaker et al., 2001] Pennebaker, J. W., Francis, M. E., and Booth, R. J. (2001). *Linguistic Inquiry and Word Count*. Lawerence Erlbaum Associates.

[Pennebaker and King, 1999] Pennebaker, J. W. and King, L. A. (1999). Linguistic styles: language use as an individual difference. *Journal of personality and social psychology*, 77(6):1296.

[Pennebaker et al., 2003] Pennebaker, J. W., Mehl, M. R., and Niederhoffer, K. G. (2003). Psychological aspects of natural language Use: Our words, our selves. *Annual Review of Psychology*, 54(1):547—-577.

[Pennington et al., 2014] Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global vectors for word representation. In *Conference on Empirical Methods on Natural Language Processing*, volume 14, pages 1532–1543.

[Picca et al., 2008] Picca, D., Gliozzo, A. M., and Ciaramita, M. (2008). Supersense tagger for italian. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC),* Marrakech, Morocco, 26 May – 1 June 2008.

[Pittenger, 2005] Pittenger, D. J. (2005). Cautionary comments regarding the myers-briggs type indicator. *Consulting Psychology Journal: Practice and Research*, 57(3):210.

[Plank et al., 2014] Plank, B., Johannsen, A., and Søgaard, A. (2014). Importance weighting and unsupervised domain adaptation of pos taggers: a negative result. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* Doha, Qatar, 25–29 October 2014, pages 968–973. Association for Computational Linguistics.

[Polanyi and Zaenen, 2006] Polanyi, L. and Zaenen, A. (2006). Contextual valence shifters. In *Computing attitude and affect in text: Theory and applications*, pages 1–10. Springer.

[Ponzetto and Navigli, 2010] Ponzetto, S. P. and Navigli, R. (2010). Knowledge-rich Word Sense Disambiguation rivaling supervised systems. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1522–1531, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Poulin and Haase, 2015] Poulin, M. and Haase, C. (2015). Growing to trust. Evidence that trust increases and sustains well-being across the life span. *Social Psychological and Personality Science*, 6(6):614–621.

[Pradet et al., 2014] Pradet, Q., Danlos, L., and De Chalendar, G. (2014). Adapting VerbNet to French using existing resources. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)* Reykjavik, Iceland, 26–31 May 2014. European Language Resources Association (ELRA).

[Preoţiuc-Pietro et al., 2015] Preoţiuc-Pietro, D., Lampos, V., and Aletras, N. (2015). An analysis of the user occupational class through twitter content. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics,* Beijing, China, 26–31 July 2015, ACL, pages 1754–1764.

[Preoţiuc-Pietro et al., 2015] Preoţiuc-Pietro, D., Lampos, V., and Aletras, N. (2015). An analysis of the user occupational class through Twitter content. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, ACL '15, pages 1754–1764.

[Preotiuc-Pietro et al., 2016] Preotiuc-Pietro, D., Schwartz, H. A., Park, G., Eichstaedt, J. C., Kern, M., Ungar, L., and Shulman, E. P. (2016). Modelling valence and arousal in Facebook posts. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* San Diego, CA, 12 –– 17 June 2016, pages 9–15.

[Qiu et al., 2011] Qiu, L., Wu, Y., and Shao, Y. (2011). Combining contextual and structural information for supersense tagging of Chinese unknown words. In *Computational Linguistics and Intelligent Text Processing*, pages 15–28. Springer.

[Quinlan, 1993] Quinlan, J. R. (1993). C4. 5: Programming for machine learning. *Morgan Kauffmann Publishers*, 38.

[Rammstedt and John, 2007] Rammstedt, B. and John, O. P. (2007). Measuring personality in one minute or less: A 10-item short version of the Big Five Inventory in English and German. *Journal of research in Personality*, 41(1):203–212.

[Ramshaw and Marcus, 1995] Ramshaw, L. and Marcus, M. (1995). Text chunking using transformation-based learning. In *Third Workshop on Very Large Corpora*, pages 82–94.

[Rangel et al., 2014] Rangel, F., Rosso, P., Chugur, I., Potthast, M., Trenkmann, M., Stein, B., Verhoeven, B., and Daelemans, W. (2014). Overview of the 2nd Author Profiling Task at PAN 2014. In *Proceedings of the Conference and Labs of the Evaluation Forum (Working Notes)*, CLEF.

[Rangel et al., 2013] Rangel, F., Rosso, P., Moshe Koppel, M., Stamatatos, E., and Inches, G. (2013). Overview of the author profiling task at PAN 2013. In *Proceedings of the Conference and Labs of the Evaluation Forum (Working Notes)*, CLEF.

[Rangel et al., 2015] Rangel, F., Rosso, P., Potthast, M., Stein, B., and Daelemans, W. (2015). Overview of the 3rd Author Profiling Task at PAN 2015. In *Proceedings of the Conference and Labs of the Evaluation Forum (Working Notes)*, CLEF.

[Rastogi et al., 2015] Rastogi, P., Van Durme, B., and Arora, R. (2015). Multiview LSA: Representation Learning via Generalized CCA. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* Denver, Colorado, 31 May –– 5 June 2015, pages 556–566.

[Raykar et al., 2010] Raykar, V. C., Yu, S., Zhao, L. H., Valadez, G. H., Florin, C., Bogoni, L., and Moy, L. (2010). Learning from crowds. *Journal of Machine Learning Research*, 11(Apr):1297–1322.

[Reder and Ritter, 1992] Reder, L. M. and Ritter, F. E. (1992). What determines initial feeling of knowing? Familiarity with question terms, not with the answer. *Journal of Experimental Psychology: Learning, memory, and cognition*, 18(3):435.

[Reese et al., 2010] Reese, S., Boleda Torrent, G., Cuadros Oller, M., Padró, L., and Rigau Claramunt, G. (2010). Word-sense disambiguated multilingual Wikipedia corpus. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC),* La Valetta, Malta, 17–23 May 2010. European Language Resources Association (ELRA).

[Řehůřek and Sojka, 2010] Řehůřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. European Language Resources Association (ELRA).

[Rennie et al., 2003] Rennie, J. D., Shih, L., Teevan, J., Karger, D. R., et al. (2003). Tackling the poor assumptions of Naive Bayes text classifiers. In *Proceedings of the 20th International Conference on Machine Learning,* Washington, D.C., 21–24 August 2003, volume 3, pages 616–623. Washington DC).

[Rentfrow et al., 2011] Rentfrow, P. J., Goldberg, L. R., and Levitin, D. J. (2011). The structure of musical preferences: a Five-Factor model. *Journal of personality and social psychology*, 100(6):1139–1157.

[Rentoumi et al., 2009] Rentoumi, V., Giannakopoulos, G., Karkaletsis, V., and Vouros, G. A. (2009). Sentiment analysis of figurative language using a word sense disambiguation approach. In *Proceedings of the Conference on Recent Advances in Natural Language Processing*, pages 370–375.

[Resnik, 2006] Resnik, P. (2006). WSD in NLP applications. *Word Sense Disambiguation: Algorithms and Applications*, pages 299–337.

[Rohrdantz et al., 2011] Rohrdantz, C., Hautli, A., Mayer, T., Butt, M., Keim, D. A., and Plank, F. (2011). Towards tracking semantic change by visual analytics. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 305–310. Association for Computational Linguistics.

[Rose et al., 2002] Rose, T., Stevenson, M., and Whitehead, M. (2002). The Reuters Corpus Volume 1-from yesterday's news to tomorrow's language resources. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation,* Las Palmas, Canary Islands, Spain, 29-31 May, 2002, volume 2, pages 827–832.

[Rosenblatt, 1958] Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386.

[Rosenthal and McKeown, 2011] Rosenthal, S. and McKeown, K. (2011). Age prediction in blogs: A study of style, content, and online behavior in pre-and post-social media generations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL),* Portland, Oreg., 19–24 June 2011, ACL, pages 763–772.

[Rosenthal et al., 2015] Rosenthal, S., Nakov, P., Kiritchenko, S., Mohammad, S., Ritter, A., and Stoyanov, V. (2015). Semeval-2015 task 10: Sentiment analysis in Twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval*, pages 451–463.

[Rosenthal et al., 2014] Rosenthal, S., Nakov, P., Ritter, A., and Stoyanov, V. (2014). Semeval-2014 task 9: Sentiment analysis in Twitter. In *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval*.

[Rothe and Schütze, 2015] Rothe, S. and Schütze, H. (2015). AutoExtend: Extending word embeddings to embeddings for synsets and lexemes. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics,* Beijing, China, 26–31 July 2015, pages 1793–1803. Association for Computational Linguistics.

[Rubenstein and Goodenough, 1965] Rubenstein, H. and Goodenough, J. B. (1965). Contextual Correlates of Synonymy. *Communications of the Association for Computing Machinery (ACM)*, 8(10):627–633.

[Rüd et al., 2011] Rüd, S., Ciaramita, M., Müller, J., and Schütze, H. (2011). Piggyback: Using search engines for robust cross-domain named entity recognition. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL),* Portland, Oreg., 19–24 June 2011, pages 965–975. Association for Computational Linguistics.

[Rumelhart et al., 1986] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning internal representation by back propagation. *Parallel distributed processing: exploration in the microstructure of cognition*, 1.

[Ruppenhofer et al., 2010] Ruppenhofer, J., Ellsworth, M., Petruck, M. R. L., Johnson, C. R., and Scheffczyk, J. (2010). *FrameNet II: Extended Theory and Practice*. International Computer Science Institute.

[Saif et al., 2013] Saif, H., Fernandez, M., He, Y., and Alani, H. (2013). Evaluation datasets for Twitter sentiment analysis: A survey and a new dataset, the STS-gold. In *Proceedings of the 1st International Workshop on Emotion and Sentiment in Social and Expressive Media: Approaches and Perspectives from AI*, ESSEM.

[Sainath et al., 2015] Sainath, T. N., Vinyals, O., Senior, A., and Sak, H. (2015). Convolutional, long short-term memory, fully connected deep neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 4580–4584. IEEE.

[Sanderson, 1994] Sanderson, M. (1994). Word sense disambiguation and information retrieval. In *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval,* Dublin, Ireland, 3–6 July 1994, pages 142–151. Springer-Verlag New York, Inc.

[Sap et al., 2014] Sap, M., Park, G., Eichstaedt, J., Kern, M., Ungar, L., and Schwartz, H. A. (2014). Developing age and gender predictive lexica over social media. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* Doha, Qatar, 25–29 October 2014, pages 1146–1151.

[Schler et al., 2006] Schler, J., Koppel, M., Argamon, S., and Pennebaker, J. (2006). Effects of age and gender on blogging. In *Proceedings of 2006 AAAI Spring Symposium on Computational Approaches for Analyzing Weblogs*.

[Schneider et al., 2013] Schneider, N., Mohit, B., Dyer, C., Olflazer, K., and Smith, N. A. (2013). Supersense tagging for Arabic: the MT-in-the-middle attack. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 661–668. Association for Computational Linguistics.

[Schneider et al., 2012] Schneider, N., Mohit, B., Oflazer, K., and Smith, N. A. (2012). Coarse lexical semantic annotation with supersenses: an Arabic case study. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL),* Jeju Island, Korea, 8–14 July 2012, pages 253–258. Association for Computational Linguistics.

[Schneider and Smith, 2015] Schneider, N. and Smith, N. A. (2015). A corpus and model integrating multiword expressions and supersenses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* Denver, Colorado, 31 May –– 5 June 2015.

[Schölkopf and Smola, 2002] Schölkopf, B. and Smola, A. J. (2002). *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press.

[Schwartz et al., 2013] Schwartz, H. A., Eichstaedt, J. C., Dziurzynski, L., Kern, M. L., Blanco, E., Ramones, S., Seligman, M. E. P., and Ungar, L. H. (2013). Choosing the right words: Characterizing and reducing error of the word count approach. In *Proceedings of the Second Joint Conference on Lexical and Computational Semantics*, *SEM, page 296–305.

[Scozzafava et al., 2015] Scozzafava, F., Raganato, A., Moro, A., and Navigli, R. (2015). Automatic identification and disambiguation of concepts and named entities in the multilingual wikipedia. In *AI* IA 2015 Advances in Artificial Intelligence*, pages 357–366. Springer.

[Seide et al., 2011] Seide, F., Li, G., and Yu, D. (2011). Conversational speech transcription using context-dependent deep neural networks. In *Proceedings of the 9th European Conference on Speech Communication and Technology (INTERSPEECH '11),* Florence, Italy, 27–31 August 2011, pages 437–440.

[Semin et al., 1995] Semin, G. R., Rubini, M., and Fiedler, K. (1995). The answer is in the question: The effect of verb causality on locus of explanation. *Personality and Social Psychology Bulletin*, 21(8):834–841.

[Seneviratne et al., 2015] Seneviratne, S., Seneviratne, A., Mohapatra, P., and Mahanti, A. (2015). Your installed apps reveal your gender and more! *ACM SIGMOBILE Mobile Computing and Communications Review*, 18(3):55–61.

[Severyn and Moschitti, 2015] Severyn, A. and Moschitti, A. (2015). Twitter sentiment analysis with deep convolutional neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 959–962.

[Severyn et al., 2013] Severyn, A., Nicosia, M., and Moschitti, A. (2013). Learning semantic textual similarity with structural representations. In *Proceedings of the 51st Conference of the Association for Computational Linguistics,* Sofia, Bulgaria 4–9 August 2013, pages 714–718.

[Shi and Malik, 2000] Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905.

[Shi and Mihalcea, 2005] Shi, L. and Mihalcea, R. (2005). Putting pieces together: Combining FrameNet, VerbNet and WordNet for robust semantic parsing. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 100–111. Springer.

[Simpson et al., 2013] Simpson, E., Roberts, S., Psorakis, I., and Smith, A. (2013). Dynamic Bayesian combination of multiple imperfect classifiers. In *Decision Making and Imperfection*, pages 1–35. Springer.

[Sinclair, 1991] Sinclair, J. (1991). *Corpus, Concordance, Collocation*. Oxford University Press Oxford, 1st edition.

[Snow et al., 2008] Snow, R., O'Connor, B., Jurafsky, D., and Ng, A. Y. (2008). Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the*

2008 Conference on Empirical Methods in Natural Language Processing (EMNLP), Waikiki, Honolulu, Hawaii, 25–27 October 2008, pages 254–263. Association for Computational Linguistics.

[Snyder and Palmer, 2004] Snyder, B. and Palmer, M. (2004). The English all-words task. In *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 41–43.

[Socher et al., 2013a] Socher, R., Ganjoo, M., Manning, C. D., and Ng, A. (2013a). Zero-shot learning through cross-modal transfer. In *Advances in neural information processing systems*, pages 935–943.

[Socher et al., 2012] Socher, R., Huval, B., Manning, C. D., and Ng, A. Y. (2012). Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1201–1211. Association for Computational Linguistics.

[Socher et al., 2010] Socher, R., Manning, C. D., and Ng, A. Y. (2010). Learning continuous phrase representations and syntactic parsing with recursive neural networks. In *Proceedings of the NIPS-2010 Deep Learning and Unsupervised Feature Learning Workshop*, pages 1–9.

[Socher et al., 2011] Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., and Manning, C. D. (2011). Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the 2011 Conference on Empirical Methods on Natural Language Processing*, pages 151–161. Association for Computational Linguistics.

[Socher et al., 2013b] Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. (2013b). Recursive deep models for semantic compositionality over a sentiment treebank. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* Jeju Island, Korea, July 2012, pages 1642–1649.

[Søgaard and Goldberg, 2016] Søgaard, A. and Goldberg, Y. (2016). Deep multi-task learning with low level tasks supervised at lower layers. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* Berlin, Germany 7–12 August 2016, volume 2, pages 231–235. Association for Computational Linguistics.

[Sordoni et al., 2015] Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J.-Y., Gao, J., and Dolan, B. (2015). A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* Denver, Colorado, 31 May –– 5 June 2015, pages 196–205.

[Stanovich and West, 1997] Stanovich, K. E. and West, R. F. (1997). Reasoning independently of prior belief and individual differences in actively open-minded thinking. *Journal of Educational Psychology*, 89(2):342.

[Stein et al., 2010] Stein, B., Zu Eissen, S. M., and Lipka, N. (2010). Web genre analysis: Use cases, retrieval models, and implementation issues. In *Genres on the Web*, pages 167–189. Springer.

[Steinberger et al., 2012] Steinberger, J., Ebrahim, M., Ehrmann, M., Hurriyetoglu, A., Kabadjov, M., Lenkova, P., Steinberger, R., Tanev, H., Vazquez, S., and Zavarella, V. (2012). Creating sentiment dictionaries via triangulation. *Decision Support Systems*, 53(4).

[Stoffel et al., 2015] Stoffel, F., Flekova, L., Oelke, D., Gurevych, I., and Keim, D. A. (2015). Feature-based visual exploration of text classification. In *Proceedings of the Symposium on Visualization in Data Science (VDS) at IEEE VIS 2015 (Online)*.

[Strapparava et al., 2004] Strapparava, C., Valitutti, A., et al. (2004). WordNet Affect: An affective extension of WordNet. In *Proceedings of the 4th International Conference on Language Resources and Evaluation,* Lisbon, Portugal, 26-28 May, 2004, pages 1083–1086. European Language Resources Association (ELRA).

[Sumanth and Inkpen, 2015] Sumanth, C. and Inkpen, D. (2015). How much does word sense disambiguation help in sentiment analysis of micropost data? In *6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA 2015)*, page 115.

[Surdeanu et al., 2011] Surdeanu, M., Ciaramita, M., and Zaragoza, H. (2011). Learning to rank answers to non-factoid questions from web collections. *Computational Linguistics*, 37(2):351–383.

[Sutskever et al., 2014] Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112.

[Taboada et al., 2011] Taboada, M., Brooke, J., Tofiloski, M., Voll, K., and Stede, M. (2011). Lexicon-based Methods for Sentiment Analysis. *Computational Linguistics*, 37(2):267–307.

[Terracciano et al., 2008] Terracciano, A., Löckenhoff, C. E., Crum, R. M., Bienvenu, O. J., and Costa, P. T. (2008). Five-Factor Model personality profiles of drug users. *BMC Psychiatry*, 8(1):22.

[Tett et al., 1991] Tett, R. P., Jackson, D. N., and Rothstein, M. (1991). Personality measures as predictors of job performance: a meta-analytic review. *Personality psychology*, 44(4):703–742.

[Tirre and Dixit, 1995] Tirre, W. C. and Dixit, S. (1995). Reading interests: Their dimensionality and correlation with personality and cognitive factors. *Personality and Individual Differences*, 18(6):731–738.

[Tognini-Bonelli, 2001] Tognini-Bonelli, E. (2001). *Corpus Linguistics at Work*, volume 6. John Benjamins Publishing.

[Tourangeau and Smith, 1996] Tourangeau, R. and Smith, T. W. (1996). Asking sensitive questions: The impact of data collection mode, question format, and question context. *Public opinion quarterly*, 60(2):275–304.

[Trivedi and Eisenstein, 2013] Trivedi, R. S. and Eisenstein, J. (2013). Discourse connectors for latent subjectivity in sentiment analysis. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 808–813.

[Trobst et al., 2002] Trobst, K. K., Herbst, J. H., Masters, H. L., and Costa, P. T. (2002). Personality pathways to unsafe sex: Personality, condom use, and HIV risk behaviors. *Journal of Research in personality*, 36(2):117–133.

[Tsvetkov et al., 2015] Tsvetkov, Y., Faruqui, M., Ling, W., Lample, G., and Dyer, C. (2015). Evaluation of word vector representations by subspace alignment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* Lisbon, Portugal, 17–21 September 2015. Association for Computational Linguistics.

[Tsvetkov et al., 2013] Tsvetkov, Y., Gershman, A., and Mukomel, E. (2013). Cross-lingual metaphor detection using common semantic features. In *The First Workshop on Metaphor in NLP*, pages 45–51, Atlanta, Georgia.

[Tsvetkov et al., 2014] Tsvetkov, Y., Schneider, N., Hovy, D., Bhatia, A., Faruqui, M., and Dyer, C. (2014). Augmenting english adjective senses with supersenses. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)* Reykjavik, Iceland, 26–31 May 2014. European Language Resources Association.

[Turney and Littman, 2003] Turney, P. D. and Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems*, 21(4):315–346.

[Turney et al., 2011] Turney, P. D., Neuman, Y., Assaf, D., and Cohen, Y. (2011). Literal and metaphorical sense identification through concrete and abstract context. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP),* Edinburgh, Scotland, 27–29 July 2011, pages 680–690.

[Utgoff, 1986] Utgoff, P. E. (1986). Shift of bias for inductive concept learning. *Machine learning: An artificial intelligence approach*, 2:107–148.

[Van der Maaten and Hinton, 2008] Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(2579-2605):85.

[Verhoeven et al., 2013] Verhoeven, B., Daelemans, W., and De Smedt, T. (2013). Ensemble methods for personality recognition. *Proceedings of the workshop on computational personality recognition (WCPR13) at the 7th international AAAI conference on weblogs and social media (Online proceedings),* Boston, Mass., 11 July 2013.

[Verhoeven et al., 2014] Verhoeven, B., Solar Company, J., and Daelemans, W. (2014). Evaluating Content-Independent Features for Personality Recognition. In *Proceedings of the 2nd Workshop on Computational Personality Recognition (WCPR14) at the 22nd international ACM conference on multimedia* Orlando, Fl., 7 November 2014, Orlando, FL, USA.

[Véronis, 1998] Véronis, J. (1998). A Study of Polysemy Judgements and Inter-annotator Agreement. In *Programme and Advanced Papers of the SensEval Workshop*, pages 2–4, Herstmonceux, England.

[Volkova et al., 2013] Volkova, S., Wilson, T., and Yarowsky, D. (2013). Exploring sentiment in social media: Bootstrapping subjectivity clues from multilingual twitter streams. In *Proceedings of the 51st Conference of the Association for Computational Linguistics,* Sofia, Bulgaria 4–9 August 2013, ACL, pages 505–510.

[Voorhees and Harman, 1997] Voorhees, E. and Harman, D. (1997). Overview of the fifth Text Retrieval Conference (TREC-5). *NIST Special Publication (SP)*, pages 1–28.

[Vossen et al., 2006] Vossen, P., Fuentes, M., et al. (2006). Meaningful results for information retrieval in the meaning project. In *Proceedings of the 3rd Global WordNet Conference*, pages 22–26.

[Wang and Manning, 2013] Wang, S. and Manning, C. (2013). Fast dropout training. In *Proceedings of the 30th International Conference on Machine Learning*, pages 118–126.

[Waterman et al., 2004] Waterman, A. H., Blades, M., and Spencer, C. (2004). Indicating when you do not know the answer: The effect of question format and interviewer knowledge on children's 'don't know'responses. *British Journal of Developmental Psychology*, 22(3):335–348.

[Welinder et al., 2010] Welinder, P., Branson, S., Perona, P., and Belongie, S. J. (2010). The multidimensional wisdom of crowds. In *Advances in neural information processing systems*, pages 2424–2432.

[White et al., 2004] White, J. K., Hendrick, S. S., and Hendrick, C. (2004). Big Five personality variables and relationship constructs. *Personality and individual differences*, 37(7):1519–1530.

[Whitehill et al., 2009] Whitehill, J., Wu, T.-f., Bergsma, J., Movellan, J. R., and Ruvolo, P. L. (2009). Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in Neural Information Processing Systems 22*, pages 2035–2043.

[Wiebe et al., 1999] Wiebe, J. M., Bruce, R. F., and O'Hara, T. P. (1999). Development and use of a gold-standard data set for subjectivity classifications. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics,* College Park, Md., 20–26 June 1999, pages 246–253. Association for Computational Linguistics.

[Wiegand and Klakow, 2010] Wiegand, M. and Klakow, D. (2010). Predictive features for detecting indefinite polar sentences. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC),* La Valetta, Malta, 17–23 May 2010, pages 3092–3096.

[Wilson, 1988] Wilson, M. (1988). MRC psycholinguistic database: Machine-usable dictionary, version 2.00. *Behavior Research Methods, Instruments, & Computers*, 20(1):6–10.

[Wilson et al., 2005] Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing*, pages 347–354.

[Wilson et al., 2009] Wilson, T., Wiebe, J., and Hoffmann, P. (2009). Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational Linguistics*, 35(3):399–433.

[Wu and Palmer, 1994] Wu, Z. and Palmer, M. (1994). Verbs semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics,* Las Cruces, N.M., 27–30 June 1994, pages 133–138. Association for Computational Linguistics.

[Xiong and Zhang, 2014] Xiong, D. and Zhang, M. (2014). A sense-based translation model for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics,* Baltimore, MD, 23–25 June 2014, pages 1459–1469.

[Yan and Yan, 2006] Yan, X. and Yan, L. (2006). Gender classification of weblog authors. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, pages 228–230.

[Yin and Schütze, 2015] Yin, W. and Schütze, H. (2015). Convolutional neural network for paraphrase identification. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* Denver, Colorado, 31 May -– 5 June 2015, pages 901–911.

[Yu and Dredze, 2014] Yu, M. and Dredze, M. (2014). Improving lexical embeddings with semantic knowledge. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics,* Baltimore, MD, 23–25 June 2014, pages 545–550.

[Zafar et al., 2015] Zafar, M. B., Valera, I., Gomez Rodriguez, M., and Gummadi, K. (2015). Fairness constraints: A mechanism for fair classification. In *2nd Workshop on Fairness, Accountability, and Transparency in Machine Learning (Online ArXiv Proceedings),* Lille, France, 11 July 2015.

[Zesch, 2010] Zesch, T. (2010). *Study of semantic relatedness of words using collaboratively constructed semantic resources*. PhD thesis, TU Darmstadt.

[Zesch et al., 2007] Zesch, T., Gurevych, I., and Mühlhäuser, M. (2007). Analyzing and accessing wikipedia as a lexical semantic resource. *Data Structures for Linguistic Resources and Applications*, pages 197–205.

[Zhang et al., 2016] Zhang, Y., Roller, S., and Wallace, B. (2016). MGNC-CNN: A simple approach to exploiting multiple word embeddings for sentence classification. *arXiv preprint arXiv:1603.00968*.

[Zhang and Wallace, 2015] Zhang, Y. and Wallace, B. (2015). A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv:1510.03820*.

[Zhao et al., 2015] Zhao, H., Lu, Z., and Poupart, P. (2015). Self-adaptive hierarchical sentence model. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 4069–4076. AAAI Press.

[Zhou et al., 2012] Zhou, D., Basu, S., Mao, Y., and Platt, J. C. (2012). Learning from the wisdom of crowds by minimax entropy. In *Advances in Neural Information Processing Systems 25*, pages 2195–2203.

[Zhu et al., 2014] Zhu, X., Guo, H., Mohammad, S., and Kiritchenko, S. (2014). An empirical study on the effect of negation words on sentiment. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics,* Baltimore, MD, 23–25 June 2014, ACL, pages 304–313.

[Zhuang and Young, 2015] Zhuang, H. and Young, J. (2015). Leveraging in-batch annotation bias for crowdsourced active learning. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pages 243–252. ACM.

List of Figures

List of Tables