Master's thesis

Master's Programme in Computer Science

Utilizing Clustering to Create New Industrial Classifications of Finnish Businesses: Design Science Approach

Miika Hyttinen

September 27, 2022

Faculty of Science

University of Helsinki

Contact information

P. O. Box 68 (Pietari Kalmin katu 5)
00014 University of Helsinki,Finland

Email address: info@cs.helsinki.fi
URL: http://www.cs.helsinki.fi/

HELSINGIN YLIOPISTO – HELSINGFORS UNIVERSITET – UNIVERSITY OF HELSINKI

Tiedekunta — Fakultet — Faculty	Koulutusohjelma — Utbildningsprogram — Study programme
Faculty of Science	Master's Programme in Computer Science

Tekijä — Författare — Author

Miika Hyttinen

Työn nimi — Arbetets titel — Title

Utilizing Clustering to Create New Industrial Classifications of Finnish Businesses: Design Science Approach

Ohjaajat — Handledare — Supervisors

Prof. Tomi Männistö, PhD Simo Linkola

Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
Master's thesis	September 27, 2022	45 pages, 9 appendix pages

Tiivistelmä — Referat — Abstract

An industrial classification system is a set of classes meant to describe different areas of business. Finnish companies are required to declare one main industrial class from TOL 2008 industrial classification system. However, the TOL 2008 system is designed by the Finnish authorities and does not serve the versatile business needs of the private sector. The problem was discovered in Alma Talent Oy, the commissioner of the thesis. This thesis follows the design science approach to create new industrial classifications. To find out what is the problem with TOL 2008 industrial classifications, qualitative interviews with customers were carried out. Interviews revealed several needs for new industrial classifications. According to the customer interviews conducted, classifications should be 1) more detailed, 2) simpler, 3) updated regularly, 4) multi-class and 5) able to correct wrongly assigned TOL classes. To create new industrial classifications, unsupervised natural language processing techniques (clustering) were tested on Finnish natural language data sets extracted from company websites. The largest data set contained websites of 805 Finnish companies. The experiment revealed that the interactive clustering method was able to find meaningful clusters for 62%-76% of samples, depending on the clustering method used. Finally, the found clusters were evaluated based on the requirements set by customer interviews. The number of classes extracted from the data set was significantly lower than the number of distinct TOL 2008 classes in the data set. Results indicate that the industrial classification system created with clustering would contain significantly fewer classes compared to TOL 2008 industrial classifications. Also, the system could be updated regularly and it could be able to correct wrongly assigned TOL classes. Therefore, interactive clustering was able to satisfy three of the five requirements found in customer interviews.

ACM Computing Classification System (CCS)

Software and its engineering → Software creation and management → Designing software

Computing methodologies → Artificial intelligence → Natural language processing → Information extraction

Information systems → Information retrieval → Document representation → Content analysis and feature selection

Avainsanat — Nyckelord — Keywords

design science, industrial classification, natural language processing, unsupervised learning, clustering

Säilytyspaikka — Förvaringsställe — Where deposited

Helsinki University Library

Muita tietoja — övriga uppgifter — Additional information

Software study track

Contents

1 Introduction

This thesis was written as a commission for Alma Talent Oy. One of the Alma Talent's divisions enhances and delivers data about Finnish companies to their customers. As the amount of data is large, roughly 2.7 million registered businesses in Finland, customers need an efficient to way to find the right data. Relevant classes assigned to companies would clearly help customer to find the right data. To solve this problem, Alma Talent Oy was interested to find new automatic ways to classify Finnish companies based on natural language data.

Term *industrial classification* is used throughout this thesis to describe the basic block of a classification system. Industrial classification describes briefly, preferably with one or two words, what a company does or in which industry it operates. The set of industrial classifications which can be used to describe different kind of businesses is called *industrial classification system*. An example of an industrial classification could be "TV commercials" which is short and easily understandable. The classification could as well be "media" which is not wrong but describes the industry on a higher level. Industrial classifications are often ambiguous and, in the end, their usefulness is decided by the one utilizing them.

1.1 Structure of the Thesis

Following Chapters 1.2 and 1.3 describe briefly the present state of industrial classifications in Finland. Chapter 1.4 defines the research problem and three research questions. Design science method is followed throughout the thesis; the purpose of such method is to provide a framework for the design process. At the heart of design science there are seven guidelines that the research process should follow (Hevner et al., 2004). All seven guidelines are discussed in detail in Chapter 2.1. To ensure this thesis will focus on solving real-world problems, I decided to interview customers. The structure of interviews, as well as the interview questions are presented in Chapter 2.2. Chapter 2.3 introduces an automatic approach to prepare natural language data set for machine learning applications. Chapter 2.4 goes through different clustering methods which can be used to extract clusters from natural language data. These clusters could potentially be labelled and used as industrial classifications. In Chapter 3, customer interviews are analyzed and different clustering

methods are applied to the natural language set. I will estimate the results of clustering experiments from the perspective of the customer interviews. Chapter 3 also reports the results of the clustering experiments in relation to the research questions. In Chapter 4, the design process is evaluated based on the seven guidelines introduced in Chapter 2.1. Finally, Chapter 5 answers the research problem along with possible future work.

1.2 Background

As mentioned above, this thesis follows design science process. To design a solution, it is crucial to know the problem. The second guideline of design science instructs that the object under design should solve a real-life business problem (Hevner et al., 2004). The problem this thesis tries to design a solution for, was recognized in the company I work for: problem is the official Finnish industrial classification system *TOL 2008* does not serve business needs of customers. The classes of TOL 2008 were created by the Finnish authorities for the use of, for example taxation and customs authorities. It may not come as a surprise that the classification system does not serve private sector's needs. This thesis studies the industrial classification system from the perspective of business needs. In the private sector, industrial classifications are utilized in different ways; businesses might want to focus their marketing or sales campaigns to companies in a specific industry, or they might want to know about new companies in their industry. In any case, the common denominator of these kind of business needs is a need to analyze companies effectively. The topic is discussed in detail in Chapter 3.1.

1.3 TOL 2008 Industrial Classification System

TOL 2008 is a hierarchical classification system that includes over 2000 industry-specific classes (Tilastokeskus, 2021). It has top-level classes, level one subclasses, level two subclasses, and so on. When starting a new business in Finland, a company is required to declare exactly one lowest level sub-class from which one can see also its higher-level classes. Table 1.1 shows an example how hierarchy is formed. Table also shows how many classes there exists in that particular class level when a top-level class is assigned. For example, for top-level class "J - Web Portals", there exists 7 level one subclasses and for level one subclass "63 - Information Services" there exists only one level two subclass.

What are the real-life problems with TOL 2008 classifications? Let us consider an example:

Table 1.1: TOL 2008 industrial classification example

Top-level Class	Subclass 1	Subclass 2	Subclass 3
J - ICT	63 - Information Services	631 - Computing and Renting Servers	6311 - Web Portals
+ 21 others	+ 6 others	-	+ 1 other

if a company runs a real estate web portal business, should the company report its business as a web portal or a real estate business? Hence, one of the problems is that companies can declare only one class. That makes TOL 2008 inflexible. Also, its terminology seems outdated. "Web Portals" and "Computing and Renting Servers" might have been relevant terms in 2008 but things have changed since, especially in the IT industry. At the same time, some traditional fields have moved closer to IT fields. Problems that TOL 2008 causes are discussed in detail in Chapter 3.1.

1.4 Research Problem and Questions

In Alma Talent Oy, the team working with company data, had presumptions that new classes could be discovered using *natural language processing (NLP)* techniques, more precisely *clustering* which is an *unsupervised learning* method (James et al., 2021). Clustering can be used to group any kind of numerical data into groups that are mathematically similar. In Chapter 2.3, I will go through how natural language data can be presented in numerical format. To use clustering to create industrial classifications, natural language about companies is needed. This thesis focuses on the processing of Finnish natural language (data) found in company websites. Other data sources could be used as well but company websites were chosen because of their accessibility. The Finnish language was chosen because Alma Talent Oy processes mostly data of Finnish companies. From these presumptions and restrictions, and above mentioned problems of TOL 2008, the research problem was formed: "*What is needed from a useful industrial classification system and what kind of process can automatize the creation of it?*".

The outcome of the research problem is connected to business decisions. If the study shows NLP's potential in forming of new classifications, further development may be considered. On the other hand, if NLP showed no potential in solving the problem, no further investments would be made. To address the research problem, three research questions were

defined. The first research question (RQ1) asks the following: *"What requirements does the classification system have from the perspective of its users?"* To answer RQ1, customer interviews was conducted to find 1) the problems of the existing TOL 2008 classifications and 2) the requirements for the new industrial classification system. Interviewees are professionals who utilize industrial classifications in their everyday work. The results of customer interviews are went through in Chapter 3.1.

Research question number two (RQ2) is related to practical natural language processing and clustering of natural language data: *"How to cluster natural language data set in practise?"* Natural language processing and clustering methods are studied comprehensively in Chapter 2.

Finally, research question number three (RQ3) asks *"Is it possible to create a new industrial classification system that satisfies the requirements found in customer interviews by using unsupervised NLP methods (clustering)?"*. As the whole problem was introduced by the customers, it is important to evaluate can these methods produce industrial classifications that are beneficial to them. The answer determines if the approach is suitable for business purposes. To make the process of answering research questions more rigorous, design science research method was utilized. RQ3 is answered in Chapter 3.3.

2 Methods

2.1 Design Science

Suppose that Y is a system that solves a customer problem and X is the process or steps which will make Y happen, simply put X causes Y. To ensure that the system under design serves its needs, it is best not to rely on assumptions. In most cases, these assumptions will eventually prove themselves wrong resulting unsatisfied customers, financial losses and frustrated developers and managers. A better approach is to use proven methods to find what the X might be. This study uses design science method to get a better perspective find X (Hevner et al., 2004). Information Systems Design Theory (ISDT) provides a theoretical background for design science (Walls et al., 1992). In the definition of design theory, Walls et al. (1992) mention that design is both "a noun and a verb", and "a product and a process". They mention that design as a process means that the plans to build the system will ensure that all requirements set for the system are satisfied. Indeed by definition, design promises to meet the requirements. However, it can fail to do so. How to ensure that the design will keep its promise to satisfy the requirements? How to find the right design X that causes Y, the system that meets its requirements? That is a question design science tries to answer. Fundamentally, design science is a problem-solving paradigm. It seeks to create something new and innovative that will increase the potential of individuals and organizations (Hevner et al., 2004). For example, in organizational information systems, increase in productivity is often sought-after.

So-called IT artifacts are at the heart of design science. They can be divided to constructs, models, methods and instantiations (March and Smith, 1995). Table 2.1 below describes IT artifacts in detail and gives some examples what they could be in the industrial classification system under design. Additionally, ISDT defines four components of a design product. First two components are meta-requirements and meta-design. It is worth of mentioning, that the four types of IT artifacts shown in Table 2.1 are all part of the second component meta-design. Requirements are set by underlying kernel theories, which is the third component. Fourth component is a testable design process hypothesis (Walls et al., 1992). What would these four components be in the industrial classification system? Meta-requirements of the industrial classification system are mostly defined by business

Table 2.1: IT Artifacts as described by March and Smith, 1995 with examples.

IT artifact	Definition	Example
Construct	Vocabulary and language used to describe the design.	Industrial classification, NLP, TOL 2008.
Model	Propositions or statements expressing relationships between constructs.	Entity-relationship model describing the design.
Method	Detailed instructions of the possible solution (design).	Unsupervised NLP methods, program code.
Instantiation	Implementation of the design	Proof of concept that demonstrates the effectiveness of the system.

and customer needs. To find out what those meta-requirements are, a rigorous method should be used. This thesis uses qualitative interview method to find requirements for the system under design. Meta-design consists of IT artifacts of the system. Motivation for doing customer interviews, as well as the kernel theory applied in this thesis, is that customer oriented design will lead to a better business success of the product (Lukas and Ferrell, 2000). Fourth component is the hypothesis that tests if the design satisfies the customer requirements. Table 2.2 describes the four components in detail.

The ultimate goal of design science is to produce new technological innovations. The core idea is that technological innovations do not born accidentally but are result of a determined design process. By design, computers do only what they are explicitly programmed to do. That is where design science method can help to boost innovations. Hevner et al. (2004) define 7 guidelines for design science research as follows:

1. Design As an Artifact
2. Problem Relevance
3. Design Evaluation
4. Research Contributions
5. Research Rigor
6. Design as a Search Process
7. Communication of Research

Table 2.2: Components of design product as described by Walls et al., 1992 with examples.

Component	General definition	Example
Requirements	The class of goals set for the product.	Customer and business needs of the classification system.
Design	Artifacts which hypothetically meet the requirements.	Construct, model, method and instantiation of the classification system.
Kernel theory	Natural or social science theories which govern requirements.	Customer oriented software development will increase the business success of the product.
Testable design process hypothesis	Test that tells if design satisfies the requirements.	Do customers see the new industrial classifications beneficial?

In the next chapter each guideline is described in detail and evaluated in relation to this particular study.

2.1.1 Design As an Artifact

The first guideline says that a design science process should produce an applicable artifact (Hevner et al., 2004). As was presented in Table 2.1, IT artifact can be in the form of construct, model, method or instantiation. Design science process can produce only one or multiple artifacts. It is easy to see that producing an instantiation of the design is easier with well produced construct, model and method. Eventually, instantiation will reveal if the design meets the goals or not. Success of the design is discussed in more detail below together the guideline four "Research Contributions".

2.1.2 Problem Relevance

Design science research should create new innovations which solve real business problems by utilizing new technology (Hevner et al., 2004). The problem the classification system tries to solve is a real business problem that was confirmed by the customers. Proposed solution uses natural language processing together which clustering, both relatively new

and actively researched technologies at the time of writing, 2022.

2.1.3 Design Evaluation

Design artifact needs to be evaluated rigorously (Hevner et al., 2004). Rigor means that the design is evaluated with well-known scientific methods. Because this thesis presents only a proof-of-concept, it is too soon to evaluate design with observational methods. Analytical methods can be used to analyze design from technical perspective, for example code analysis or performance testing. The nature of industrial classifications is that there is no ground truth. Therefore, it is impossible to measure exact accuracies of the classifications. When it comes to proof-of-concept designs, experimental evaluation is very natural way to evaluate them. Actually, a proof-on-concept is a controlled experiment or simulation on its own. Structural testing (white box) suits also well evaluating of, for example clustering: it can be easily tracked what went wrong if a company is categorized to a wrong class. Informed argument and scenario are forms of descriptive design evaluation methods. This thesis is an extensive informed argument on its own. Therefore, the suitableness of descriptive methods is obvious. Table 2.3 summarizes design evaluation methods and their suitableness in this particular study.

Table 2.3: Design evaluation methods as described by Hevner et al., 2004 and their suitableness in this study.

Type	Method	Suitableness
Observational	Case Study, Field Study	Partly
Analytical	Static Analysis, Architecture Analysis, Optimization, Dynamic Analysis	Partly
Experimental	Controlled Experiment, Simulation	Yes
Testing	Functional (Black Box) Test, Structural (White Box) Testing:	Partly
Descriptive	Informed Argument, Scenario	Yes

2.1.4 Research Contributions

The fourth guideline states that design should contribute in the fields of design artifact, design construction knowledge, or design evaluation knowledge (Hevner et al., 2004). If design science method is followed rigorously, it is almost inevitable, that design process will show at least some contributions in some of those areas. From the perspective of business, that is not clearly enough. Design artifact should provide a solution to real-life business problem to be considered successful (Hevner et al., 2004). What kind of business success can be expected from design processes? Any explicit studies on business success of design science processes could not be identified. However, could technology start-up companies be seen as a real-life design laboratories? Most of them strive business success with new technology. In a study of 214 start-up post mortem reports Cantamessa et al., 2018 found that 86% of the newly founded start-ups were out of business after 5 years and 58% of them survived under 3 years. Taking these failure rates into consideration, it is reasonable to assume that attempts to design an artifact that delivers long-lasting and profitable business value do fail relatively often.

2.1.5 Research Rigor

Hevner et al., 2004 mention that "Design science research requires the application of rigorous methods in both the construction and evaluation of the designed artifact." Construction of the industrial classification system artifact happens in two stages. First, the customers and business experts are interviewed to find the requirements for the design artifact. Interviews are analyzed in Chapter 3.1. Second, methods used to do the industrial classifications itself should be described rigorously. Assuming that some kind of unsupervised natural language processing method solves the problem, describing computational methods rigorously is most of the time straightforward. Chapter 2.3 describes these methods theoretically. Practical experiments are done by applying the described methods to real-life data. Finally, the results of the experiments are analyzed in Chapter 3.

2.1.6 Design as a Search Process

In *The Sciences of the Artificial* Simon (1996) says that design process can be thought to happen in two in phases. First, design process generates different alternatives and then tests these alternatives against known requirements and constraints. About the process,

Simon (1996) uses the name *generator-test cycle* while March and Smith (1995) have named the process as *build and evaluate*. Both describe a very similar process. Design process is not limited to one cycle but is naturally iterative (Hevner et al., 2004). This thesis acts as the first round of the design cycle. Aforementioned assumptions about challenges of the design process are connected to the iterative nature of design process; the more iterations there are, the more successful the design will be.

2.1.7 Communication of Research

Results of the design science research should be presented understandably to both technology- and business-oriented audiences (Hevner et al., 2004). Technical personnel, for example software developers, need detailed information about the technologies design utilizes. Business-oriented audiences need information on how much resources implementation of the design would need (Hevner et al., 2004). In large organizations, such as the commissioner of this thesis, new industrial classification system would definitely arouse wide interest. Therefore, the designer must take care that the results of the design science research are presented to wide audience.

2.2 Qualitative Interviews

Interviews are a common approach to achieve better perspective on the system under design. To find out what are the meta-requirements of the new industrial classification system, qualitative customer interviews were conducted. Before starting the interview process, an interview method needs to be defined. If not, a researcher easily ends up empirically proving researchers own pre-assumptions (Alasuutari, 2011). Because of the experimental nature of this thesis, the number of interviewees was limited to three. The goal of interviews was to find insights for defining requirements of the design. As Alasuutari (2011) mentions, when trying to understand what kind of meaning interviewees give to certain things, it is best to let them describe things in their own words. Alasuutari points out that recording is the most accurate way to document interviews. The interviews were arranged as remote video meetings which were recorded and then transcribed. Transcribed interviews can be found in appendices (in Finnish).

All three interviewees are specialists in their own field: sales and marketing professional, journalist and business administrator. Interviews were conducted as semi-structured in-

terviews and three questions were:

1. What kind of problems TOL-2008 classification causes?
2. What kind of industrial classifications would suit your needs?
3. How would you utilize the classifications you described?

Any information that emerges from interviews may be valuable. In fact, the questions are meant to act as starting point for a conversation. Alasuutari, 2011 mentions that qualitative analysis of the interviews happens in two stages: "simplifying of observations" and "solving the puzzle". Simplifying means looking for the common denominator, and solving the puzzle means making the right conclusions from the interview material. Based on the conclusions, relevant meta-requirements can be set for the new industrial classification system.

2.3 Natural Language Processing

Natural Language Processing (NLP) is a branch of computer science that combines both artificial intelligence and linguistics. Some examples of traditional NLP applications are email spam filters and customer feedback classifiers. Natural language can be also in the form speech but this thesis focuses only in the processing of written natural language i.e. *text documents*. As mentioned earlier, the problem I am designing a solution for, is that the official TOL 2008 Finnish industrial classifications do not represent the actual industries companies operate. The problems TOL 2008 classification causes are described in detail in Chapter 3.1. The research problem defined in Chapter 1.4 asked how to automatize the process that creates an industrial classification system. This study focuses on unsupervised method called clustering. Before it is possible to apply clustering to a natural language data set, several steps needs to taken. Figure 2.1 visualizes these steps, together often referred as *NLP pipeline*.

2.3.1 Data Collection

NLP starts by finding relevant data sources, usually a sufficient amount of text documents. Text documents can be in many forms: physical text documents, PDF files, text files, HTML documents, databases, hand written documents, social media posts etc. Because

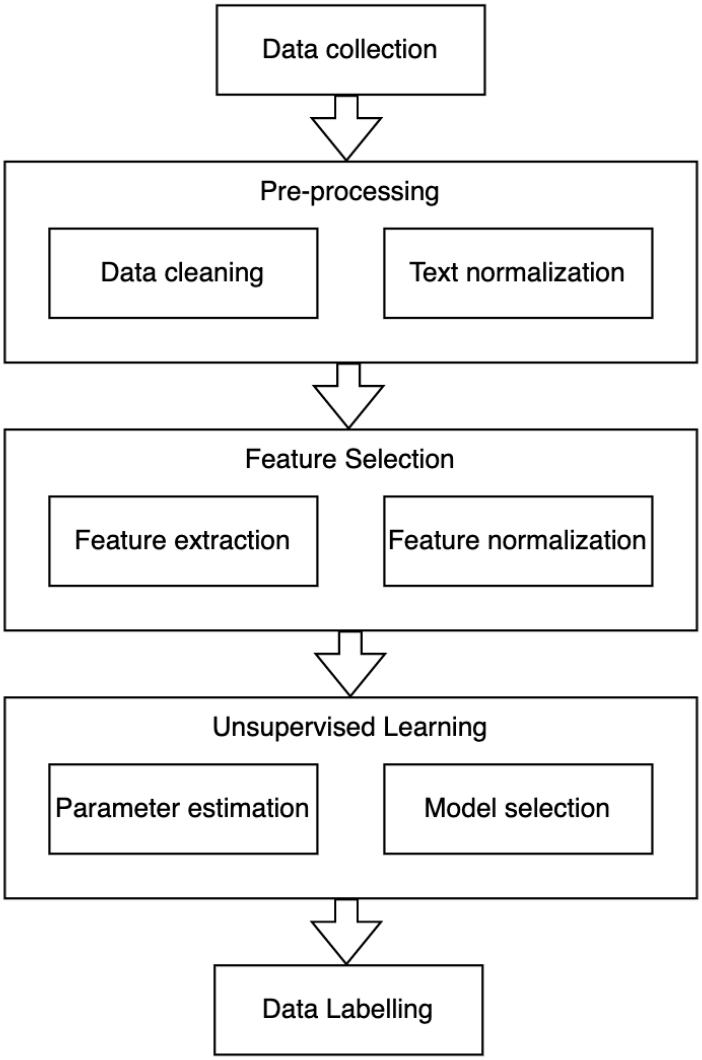

Figure 2.1: Unsupervised Learning NLP Pipeline

the commissioner was interested in applying unsupervised learning methods to company website data, HTML documents were used as the data source. Programmatic collection of text documents is a topic of its own and is not in the scope of this thesis.

2.3.2 Pre-processing

Data Cleaning

After collecting the HTML documents, natural language needs to be extracted from HTML documents. Python library Beautiful Soup 4 was used to pull natural language data out

of HTML documents as well as to remove all scripts (Richardson, 2022). As Finnish language contains "Ä" and "Ö" letters, it was important to convert all of the text data to Unicode standard (Unicode, 2021). Python 3.9.7 documentation states that "strings are immutable sequences of Unicode code points" meaning that the Unicode support is built-in in modern Python (Python Software Foundation, 2021).

Tokenization

As a single text document is represented by single string, the purpose of tokenization is to cut string "into identifiable linguistic units" (Bird et al., 2009). There are existing NLP libraries to perform tokenization, for example Natural Language Tool Kit (NLTK) for English language (NLTK Project, 2021) and Voikko library for Finnish (Pitkänen, 2021). Lower-casing and spelling check are also a part of text cleaning process and both functionalities are included in aforementioned libraries. Table 2.4 shows an example sentence taken from Nokia website to illustrate what happens in tokenization in practise.

Table 2.4: Example of tokenization

Raw text	"As a trusted partner for critical networks, we are committed to innovation and technology leadership across mobile, fixed and cloud networks."
Tokinezed text	["as", "a", "trusted", "partner", "for", "critical", "networks", "," , "we", "are", "committed", "to", "innovation", "and", "technology", "leadership", "across", "mobile", "fixed", "and", "cloud" "networks", "."]

2.3.3 Text Normalization

After tokenization, words need to be reduced to their base forms. English language includes fairly little inflected forms while Finnish has an enormous number of suffixes. Yet, the goal of text normalization is the same: finding the base form of the words in text documents. For example "programmers" would be normalized to "programmer" and Finnish equivalent "ohjelmoijat" would become "ohjelmoija". To achieve this programmatically, two common approaches are used: *stemming* and *lemmatization*.

Stemming and Lemmatization

Stemming is a process where affixes of inflected words are cut away. For example, Porter's algorithm is a well-known stemming algorithm for English language (Porter, 1980). Stemming works well when language has a limited number of suffixes, such as English. Normalization of languages with a large number of suffixes like Finnish, requires slightly different approache. Kettunen (2006) showed that inflectional stems and lemmatization produce the best results in best-match search tasks in Finnish. Lemmatization method utilizes dictionary to find the base form of the word whereas inflectional stems generates inflected form of the word. In modern text normalization, the conceptual difference between stemming and lemmatization is not that clear. Both Python libraries, Voikko and NLTK provide off-the-shelf normalization. At this point, so called *stopwords* are also removed. Stopwords are the words which have only a little significance when making a difference between text documents (Sarkar, 2016). There is no comprehensive list of stopwords but commonly at least articles, adverbs, dots, commas, and prepositions are considered stopwords. In Table 2.5, the words of tokenized example sentence are reduced to their basic forms. Also stopwords are removed.

Table 2.5: Example of normalization

Normalized text with stopwords removed	[“trust”, “partner”, “critical”, “network”, “we”, “is”, “commit”, “innovation”, “technology”, “leadership”, “across”, “mobile”, “fix”, “cloud” “network”]

Word Tagging and Categorization

POS tagging stands for *parts-of-speech* tagging. Parts-of-speech are also known as word classes or lexical categories such as nouns, adjectives and verb (Bird et al., 2009). Intuitively, it is easy to see that frequently appearing words do not make the differences in text document classification. Therefore, it might be good idea to filter out some words before clustering text documents. Again, both Voikko and NLTK libraries contain POS tagger. Table 2.6 shows example what POS tagged text would look like. In Finnish language word classes are slightly different but the idea is the same.

Table 2.6: Example of POS tagged text

POS tagged text	[{ word: “trust” tag: [“verb” , “noun”] }, { word: “partner” tag: “noun” }, { word: “critical” tag: “adjective” }, { word: “network” tag: “noun” }, { word: “we” tag: “pronoun” } , ...]

2.3.4 Feature Extraction

Previous steps have presented how to pre-process text documents for machine learning purposes. To apply any mathematical methods, text documents need to be transformed into numerical format. In machine learning, this process is called *feature extraction*. *Feature* is a machine learning term that Sarkar (2016) describes as “unique, measurable attribute or property for each observation or data point in a dataset”. In this case, features are the words in text documents. *The Vector Space Model* is an algebraic model to represent text documents (Sarkar, 2016). Formally n-dimensional vector space VS is defined as

$$VS = \{W_1, W_2, ...W_n\} \ ,$$

where W stands for each distinct word and n is the number of distinct words. Document D can be represented in vector space VS as

$$D = \{w_1, w_2, ...w_n\} \ ,$$

where w stands for the weight of word n in specific document D (Sarkar, 2016). Weight w is a numeric value that could be, for example, occurrences of a word in a text document.

Before text document are converted into weights, it is possible to filter out some words in addition to stopwords. This process is known as *feature selection*. Both Sarkar (2016) and Bird et al. (2009) mention that using only certain word classes like nouns, may be an advantage sometimes. Let us take an another example sentence from KONE corporations home page: *“As a global leader in the elevator and escalator technology, KONE provides elevators, escalators and automatic building doors”*. Table 2.7 shows normalized nouns of both example documents D_{Nokia} and D_{KONE}. In machine learning terminology, collection of documents is often called a *data set*. Data set is a broader term that can refer to any

type of data, not just documents. Let us call the data set of two documents as *demo data set*.

Table 2.7: The demo data set with two example documents

D_{Nokia} nouns	[... "network" ,"innovation", "technology", "network" ...]
D_{KONE} nouns	[... "elevator","escalator", "technology" ...]

Bag of Words

Next, the selected features need to be converted into numerical form. *Bag of Words* is a technique to extract features from text documents (Sarkar, 2016). It is fundamentally the same as Vector Space Model where weights are the frequencies of word occurrences in a document. Table 2.8 shows bag of words representation of the demo data set.

Table 2.8: BOW representation of the demo data set

Words	elevator	network	innovation	technology	escalator
D_{Nokia} weights	0	2	1	1	0
D_{KONE} weights	1	0	0	1	1

2.3.5 Feature Normalization

Tf-idf Model

Bag of words model uses absolute number of word occurrences. It may become a problem, as the words that would make the difference do not stand out because some more general word occur more often. To address the problem, *Term Frequency-Inverse Document Frequency* (tf-idf) normalization can be used (Sarkar, 2016). Tf-idf is the product of two metrics: term-frequency (tf) and inverse-document-frequency (idf). Tf is the absolute term frequencies in the certain document, formally defined as

$$tf(w, D) = f_{wD} \ ,$$

where w stands for a word and D for a document. The second part, Idf is defined as

$$idf(t) = 1 + \log(\frac{C}{1 + df(t)}) \ ,$$

where t is the term, C is the total number of documents, *log* is a natural logarithm, and $df(t)$ the number of documents where term t is present. One is added to denominator to prevent potential division-by-zero errors and smoothen the inverse document frequencies (Sarkar, 2016). This is also known as *additive smoothing* or *laplace smoothing*. With the formulas of tf and idf we can define tf-idf as

$$tfidf(t, D) = tf(t, d) \cdot idf(t) \ ,$$

where t is the term, d is the document, and D is the collection of documents.

Let us calculate idf for term "technology" in the demo set:

$$idf(\text{“}technology\text{”}) = 1 + \log(\frac{2}{1 + 2}) \approx 0.59 \ .$$

Then, tf-idf can be calculated as follows:

$$tf(\text{“}technology\text{”}, D_{Nokia}) \cdot idf(\text{“}technology\text{”}) = 1 \cdot 0.59 = 0.59 \ .$$

Note, that the tf is calculated for each term in the document (matrix cell) and idf for each term (matrix column). Finally Sarkar (2016) instructs to normalize values of tf-idf matrix by dividing values with Euclidean L2 norm of each document. According to Schaefer (1971), Euclidean L2 Norm is presented mathematically as follows (edited by the author)

$$\|tfidf(\mathbf{t}, d)\| = \sqrt{tfidf(t_1, d)^2 + tfidf(t_2, d)^2 + ... + tfidf(t_n, d)^2} \ ,$$

where $\mathbf{t}$ is all terms in the document d, and $tfidf$ is calculated as in the formula above. Let us calculate tf-idf Euclidean L2 norm for document D_{Nokia}:

$$\|tfidf(\mathbf{t_{Nokia}}, D_{Nokia})\| = \sqrt{0^2 + 2^2 + 1^2 + 0.59^2 + 0^2} \approx 2.31 \ .$$

To get the normalized tf-idf let us divide tf-idf with Euclidean L2 norm of the document:

$$\frac{tfidf(\text{“}technology\text{”}, D_{Nokia})}{\|tfidf(\mathbf{t_{Nokia}}, D_{Nokia})\|} = \frac{0.59}{2.31} \approx 0.25 \ .$$

Finally, the goal is to build the bag of words -style matrix but with normalized tf-idf values as weights. Idf part of the tf-idf downgrades the values of terms that occur in large portion of the documents which should make documents more easily separable when clustering them.

2.4 Unsupervised Learning: Clustering methods

Now that we know how to prepare natural language data for clustering, I will present clustering methods used in this thesis. Ultimately the goal is to design new industrial classification system that aims to classify companies to relevant industries. Traditionally classifications are done using methods like *naive bayes* or *support vector machine*. These algorithms represent branch of machine learning called *supervised learning* and require labelled training data (Sarkar, 2016). The problem is, that such data does not exist for this particular problem. However, *unsupervised learning* algorithms do not require labelled data. Clustering is an unsupervised learning method that can be used to find clusters in data. After the clusters are found, clusters need to be given descriptive names before they can be utilized as industrial classifications.

2.4.1 K-means Clustering

K-means clustering aims to divide data set into K non-overlapping clusters $\mathbf{C} = C_1, C_2, \ldots, C_K$ (James et al., 2021). Number of clusters K needs to specified in advance which can be considered as a disadvantage of the k-means clustering (Sarkar, 2016). James et al. (2021) describe the K-means clustering as an optimization problem: goal is to partition clusters in a way that Euclidean distances between samples within clusters are minimized. The standard algorithm to solve the optimization problem is Lloyd's algorithm which an iterative solution to find a local optimum for the problem (Lloyd, 1982). But how to tell what number of K clusters is optimal? One approach is to calculate *silhouette coefficient*, also known as *silhouette score*. Silhouette score tells us about “separation distance between

the resulting clusters" (Scikit-learn, Developers, 2021f). It has a range of $[-1, 1]$. If the score is near 1 it means sample is distant from the neighbour clusters. Score near 0 means that sample is near to the decision line between two clusters. Negative values indicate that sample could have been assigned to wrong cluster (Scikit-learn, Developers, 2021f). Silhouette score is calculated for each sample but the mean of silhouette scores of all clustered samples is often used to compare clusters with different Ks.

2.4.2 Hierarchical Clustering

Hierarchical clustering will results in a *dendrogram*, a hierarchical tree-model (James et al., 2021). The most used type of hierarchical clustering is called *agglomerative clustering*. In agglomerative clustering, as we move up in the hierarchy tree, leaves start to fuse into branches (James et al., 2021). The lower the fuse happen, the more similar observations are. When interpreting a dendrogram, only the distance of fuses with respect to x-axis is relevant. Y-axis does not have any significance in dendrogram. Hierarchical clustering for n samples is achieved with a two-stage algorithm. Tree is built literally bottoms-up which is also an another name for the agglomerative clustering. Below is the algorithm as presented by James et al. (2021):

1. Start by measuring dissimilarities (e.g. Euclidean distance) for all pairs. Treat each observation as its own cluster.
2. For $i = n, n-1, ..., 2$:
   (a) Examine all pairwise inter-cluster dissimilarities among the i clusters. Identify the most similar cluster pairs and fuse them. The dissimilarity between the two clusters corresponds to the height (x-axis) of the dendrogram. The bigger the dissimilarity is, the higher in x-axis fuse is placed.
   (b) Calculate new pairwise dissimilarities between clusters for the $i-1$ clusters left.

Calculating dissimilarity between two observations is very straightforward, for example using Euclidean distance. What if we want to calculate dissimilarity between two clusters containing two or more observations? To measure dissimilarity between clusters the concept of *linkage* is used. James et al. (2021) mentions that the four common types of linkage are *complete, average, single, and centroid*. Also *Ward's method* is a popular way to calculate linkages between clusters (Sarkar, 2016). Ward's method will be used in hierarchical clustering experiments presented in Chapter 3.

2.4.3 DBSCAN

DBSCAN stands for Density-based Spatial Clustering of Applications with Noise - the clustering method was presented by Ester et al. (1996). Advantages of DBSCAN is that it can detect non-convex clusters and it does not require number of clusters set beforehand. A set is said to be *convex*, if all the points between any two points of set, is included in the set (Klee, 1971). A set is said to be *non-convex* if it does not satisfy this condition. An example of a non-convex set could be a concave set (shape of a letter C) in two-dimensional Euclidean space. Additionally, DBSCAN categorizes a part of the samples as noise, meaning that they do not belong to any cluster. DBSCAN requires two parameters to be estimated beforehand: *epsilon* and *minPts* (Ester et al., 1996). Value for epsilon can be found by calculating k-nearest-neighbours of the feature matrix and plotting them from smallest to largest. Then we look for an "elbow" in the plot, which shows our desired epsilon value. For minPts, a convention is to choose a value twice the number of dimensions of the data set. The lower minPts value is set, the smaller the groups of samples are that DBSCAN considers clusters (Schubert et al., 2017).

This chapter discussed three different clustering methods: K-means, Hierarchical, and DBSCAN. I have used the Euclidean distance as an example similarity metric. There are other similarity metrics like the *coisine similarity* or correlation-based metrics like *pearson correlation* but the euclidean distance will be used through the study. Next chapter presents a clustering experiment on a natural language data set gathered from company websites.

3 Results

As mentioned earlier, construction of industrial classification system happens in two stages. First, the requirements are set and then process to produce such system is described rigorously. In Chapter 2, I described the theoretical background to find requirements and to produce industrial classification from natural language data using unsupervised learning methods. In this chapter customer interviews are analyzed and then the three clustering methods are applied to real-life data sets.

3.1 Interviews

Motivation behind customer interviews (kernel theory) is that customer-orientation will generally lead to better business success (Lukas and Ferrell, 2000). From design perspective, interviews are needed to define relevant meta-requirements for the industrial classifications system. I decided to interview three business professionals who have experienced the problems of TOL 2008 classifications in their everyday work. Interviews were conducted in Finnish and the key points are presented here in English as translated by the author. Table 3.1 describes the professions of interviewees as well as their use cases.

Table 3.1: Interviewees and their professions and uses cases

Interviewee	Profession	Use case
A	Sales professional	Customer segmentation
B	Business journalist	Key statistics of different industries
C	Administrative director	Decision-making

The need to develop a new industrial classification system emerged from the problems TOL 2008 classification causes. Interviewees were asked to describe problems TOL 2008 has caused in their work.

Question 1: What kind of problems TOL-2008 classification has caused in your work?

Interviewee **A** uses TOL 2008 to find new customers. **A**: "We have a lot of customers in certain industries. We have contacted 9 out of 10 of potential customers in those industries." **A** mentions that the IT industry is especially problematic because it is impossible to say who develops, who consults or who offers the software. All of those three are under the same classification in TOL 2008. **A** mentioned also that the TOL 2008 classifications are used extensively by the Finnish customs.

B says that "it is hard to define practical and easily understandable industrial classifications". Interviewee **B** told about a task where **B** was assigned to write a piece of news about the ten largest education companies in Finland. TOL 2008 classification provided a starting point but some major companies were missing because of their classification label was not inside the education top-class.

Interviewee **C** wants to discover retail and wholesale companies in Finland. **C**: "Nowadays companies do both retail and wholesale. TOL 2008 class is one or the other, not both. Also finding companies that run online stores is not possible. Third, some companies report some other industry related to their area of sales. But it does not indicate that they do retail or wholesale." **C** mentioned that authorities in Finland used TOL 2008 classifications to allocate financial support during COVID-19 pandemic to businesses. If a company reported certain industry, support was granted automatically. Otherwise company needed to apply for the support.

Summary: At the same time, TOL 2008 classifications is too general and too specific. Especially in fast developing IT industry, classifications are outdated. Second major problem is that companies report their industries by themselves, resulting varying classifications among the industries. Third, TOL 2008 is too stiff. Companies can report only one main industry while they do business in multiple industries. Also, TOL 2008 does not seem to always serve authorities' needs either.

Question 2: What kind of industrial classifications would suit your needs?

A: "To find potential customers we have not discovered yet, we would need to find very specific combinations, for example SaaS accounting companies. If you could split large industries to smaller and clearer groups it would benefit us."

B: “From the perspective of a journalist, I had to make a decision if a company is an education company. Generally, a pragmatic classification is what I would find useful. A pragmatic example could be university preparation course providers.”

C: “It would be useful, if the industrial classifications were updated regularly. As I mentioned before, we need to know if companies do retail, wholesale or both. On the other hand, we do not need that specific information about the companies. For example, do the store sell fish or meat. The most importantly, industrial classifications should benefit the majority.”

Summary: Interviewees mentioned very specific needs for the classification system. The new classification system could either be detailed at the expense of simplicity or fixed at the expense of accuracy. Major improvement compared to TOL 2008 would be multiple classifications. Interviewees described specific problems that could be solved by combining classes.

Question 3: How would you utilize classifications you described?

Interviewee **A** described a practical problem their organization have encountered: “Our sales organization is relatively small and we do not have resources to contact all potential customers. If we had a reliable list of those companies, we would assign an external contractor to do the initial work. Then we would contact the interested customers by ourselves.”

B: “In many cases you need to be familiar with the industry to find the right companies. You just need to know those. To make the decision if the company is an education company, I would need to know how large proportion of their sales comes from the education business.”

C mentioned that they would need better statistics to do well-founded decisions. First, **C**’s organization’s board of directors is formed relative to their members’ field of businesses. Second, they would need to target their communication more accurately to certain type of companies.

Summary: Interviewees described different use cases for the new classification system. Better classification would reduce the need of industry specific knowledge. It would clearly be useful to provide additional information about the companies, for example financial and contact information along with the classifications.

3.2 Unsupervised NLP experiments

Chapter 2.3 discussed number of techniques needed for natural language processing. In this Chapter those methods are put into practise. First, the data sets are presented, second the pre-processor is optimized, and at last three clustering methods are applied to data sets. The programming language used was Python 3.

3.2.1 Data Sets

To experiment with NLP, two data sets were collected: *toy data set* and *experiment data set*. The toy data includes 40 samples: company metadata and Finnish language from their websites. The idea of having such small data set was to test different combinations of pre-processor parameters. With 40 companies it is reasonable to do intuitive observations which can then be generalized to a larger data set.

The toy data set have two versions, short and long. The short version uses the language extracted only from front pages. The long version extends the data with maximum 20 HTML pages which links were found on the front page. Only links inside the company domain were used. To extract the language I wrote a custom scraper. The scraper parses the language from the HTML websites as well as randomly selects the links. Web scrapers are outside of the scope of this thesis but the code is provided in the code repository (Hyttinen, 2022).

Companies in the toy data set were selected from TOL categories “J - Information and communication” and “G - wholesale and retail”. I also labeled each sample with an industry to keep track if clusters were formed correctly. Labelling was done by intuition and its purpose is just to provide a reference point. Companies and their industries are discussed in detail together with the cluster analysis. For the experiment data set, I selected 805 companies from TOL top-classes “F - construction”, “J - Information and communication”, “G - wholesale and retail” and “P - education”. These industries are the ones interviewees mentioned.

3.2.2 Feature Extraction

Feature extraction were done as presented earlier in Chapter 2.3. First, bag of words (BOW) feature matrix was created which was then transformed into a tf-idf feature matrix.

Rows were mapped to a corresponding company and columns to a corresponding feature (word). BOW uses `CountVectorizer` class from the open source *Scikit-learn Library* (Scikit-learn, Developers, 2021b). Tf-idf matrix was created using `TfidfTransformer` class (Scikit-learn, Developers, 2021g). The code and data sets are provided in a Github code repository (Hyttinen, 2022).

3.2.3 Pre-processor and Parameter Optimization

Pre-processor implementation uses Voikko-library to lemmatize and POS tag Finnish language data. Voikko uses *Joukahainen* dictionary to match words during lemmatization process (Pitkänen, 2021). Pre-processor takes three arguments: data frame containing unprocessed natural language with company meta-data, Voikko POS codes to include desired parts-of-speeches, and a boolean flag to split or preserve word compounds. Voikko-library was used to restore word to its base form as well as to POS tag the word. With the help of POS-tags, it is possible to filter out all other words but nouns. Finally, the pre-processor writes pre-processed data into a .csv file along with company meta-data. To find out which combination of pre-processor parameters would be the best, silhouette scores were measured for both long and short versions of the toy data set. Three combinations of pre-processor parameters were tested: 1) all words with no compound splitting 2) only nouns with no compound splitting and 3) only nouns with compound splitting. To calculate silhouette scores, Scikit-learn library's silhouette score implementation were used (Scikit-learn, Developers, 2021f). Figure 3.1 shows measured silhouette scores of different pre-processor parameter combinations for different number of clusters.

The scores are close to zero which indicates that clusters are not separated well. However, it seems that using compound splitting improves the scores. Using only nouns and compound splitting, the short version of the toy data set has 1854 features (words) and the long version has 2655. After inspecting the feature matrices, it was found out that most of values are zero, meaning that the feature matrices are very sparse. With this low silhouette scores and high number of features compared to the number of rows, I assumed that I have encountered *the curse of dimensionality*. It is commonly known issue in machine learning which happens when a feature matrix is very sparse and has a large number of features. In other words, the feature matrix is high dimensional. The curse of dimensionality can be tackled with dimensionality reduction techniques, for example *Principal Component Analysis (PCA)* and *Singular Value Decomposition (SVD)* (James et al., 2021). I decided to use a special version of SVD called *Truncated SVD* which works well on term count and

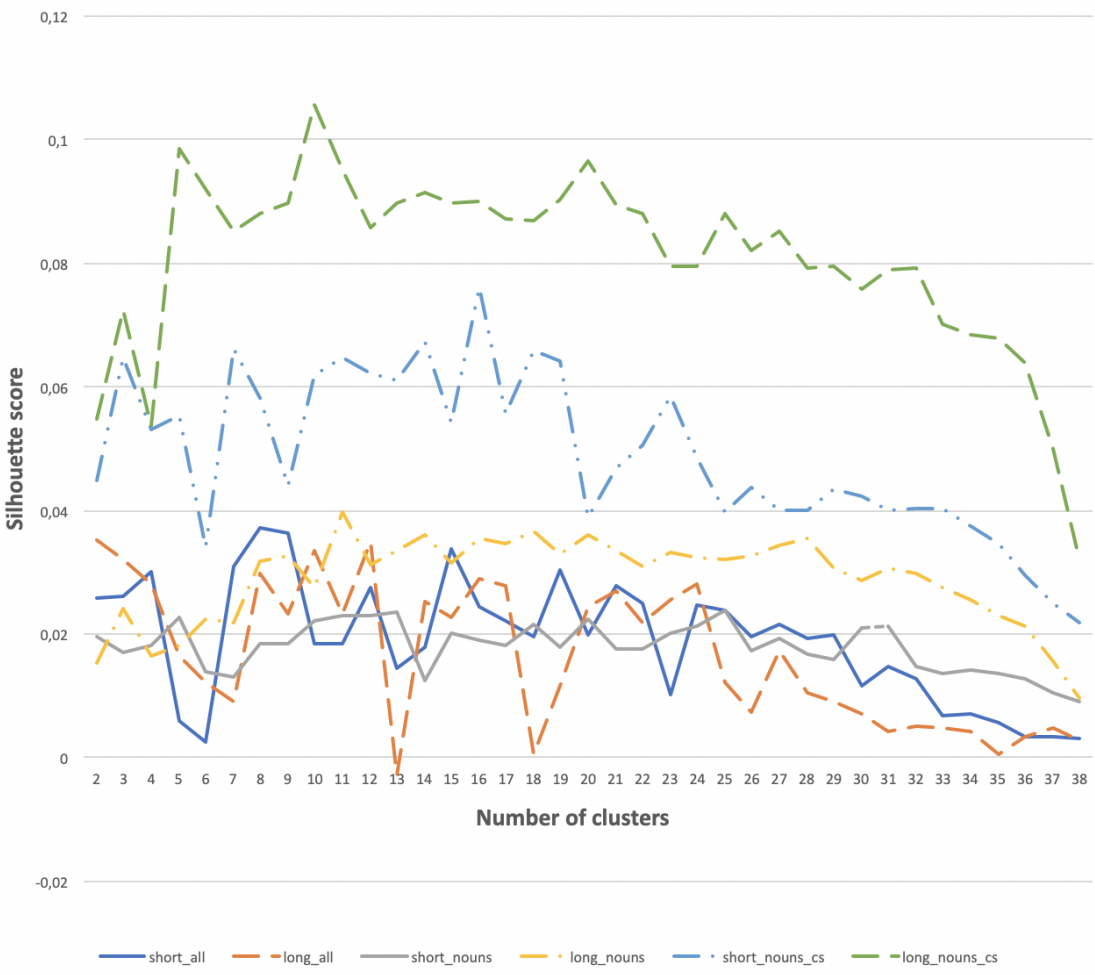

Figure 3.1: Silhouette scores for 3 different pre-processor parameter combinations using both long and short versions of the toy data set.

tf-idf matrices (Scikit-learn, Developers, 2021h). Before reducing the dimensions, number of desired components need to be decided. Scikit-learn library's `TruncatedSVD()` has an attribute `explained_variance_ratio` which tells percentage of variance explained by each of the selected components (Scikit-learn, Developers, 2021h). I reduced the toy data set to 100 components and inspected variance ratios of each component. Variance ratio values showed significant decrease after 5th component. Therefore, I decided to reduce the feature matrix of the toy data set to 5 components and run silhouette score measurements again. Figure 3.2 shows measured silhouette scores with the toy data set reduced to 5 components.

Reducing feature matrix to 5 components improved silhouette scores significantly. Interestingly, the short version of the toy data set performed the best. The scores are at their highest when the number of clusters is 8 or 9. This indicates that the optimal pre-

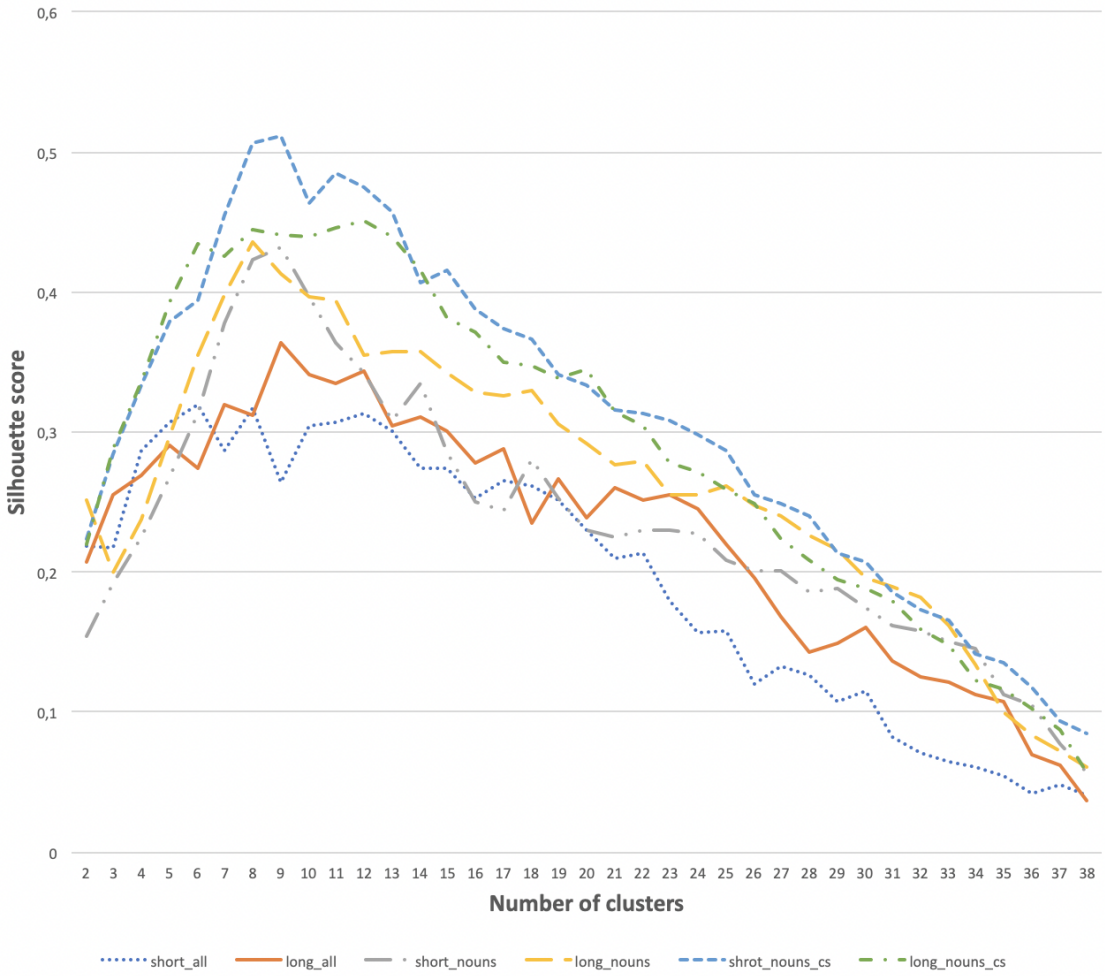

Figure 3.2: Silhouette scores for 3 different pre-processor parameter combinations using Truncated SVD (5 components): long and short versions of the toy data set.

processor parameter set up is to use only front pages, filter out all but nouns, and to split compounds.

Next, I calculated silhouette scores for the experiment data set using the best pre-processor parameters found by experimenting with toy data set. I collected 805 front pages of the companies selected for the experiment data set. Pre-processor was set to filter out all other words but nouns and to split compounds. Feature matrix was reduced to 8 dimensions. Interestingly, with the experiment data set there was a drop of about 0.20 in silhouette scores from 2 to 3 clusters, 0.423 being the highest score with two clusters. Figure 3.3 shows measured silhouette scores for the experiment data set.

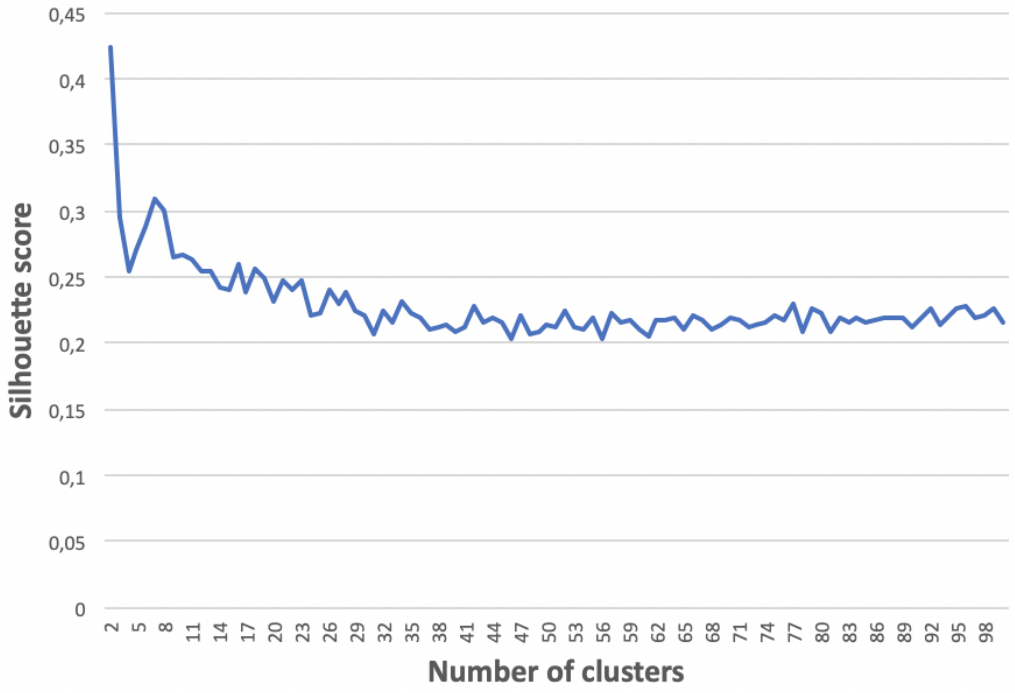

Figure 3.3: Silhouette scores for the experiment data set: only nouns, compound splitting and Truncated SVD with 5 components.

3.2.4 Clustering Experiments

Silhouette score were used as a measure to find out the optimal parameters for the preprocessor. Measurements also indicated that more data (natural language) does not necessarily improve the results. It was also showed that reducing the dimensionality of the feature matrix should help to separate clusters. Next, the three clustering methods presented earlier was applied to the data sets.

Hierarchical clustering

For start, I decided to apply hierarchical clustering method to the short version of toy data set. Ward's method was used to calculate linkage with Euclidean distance as a distance metric (The SciPy community, 2021). Figure 3.4 shows a dendrogram produced. Five smaller clusters seem to be meaningful having only single mistakes. The largest green colored cluster consists of software and news companies but it is not necessarily formed because it is something meaningful but instead it is not something. Higher level clusters do not seem to make any sense.

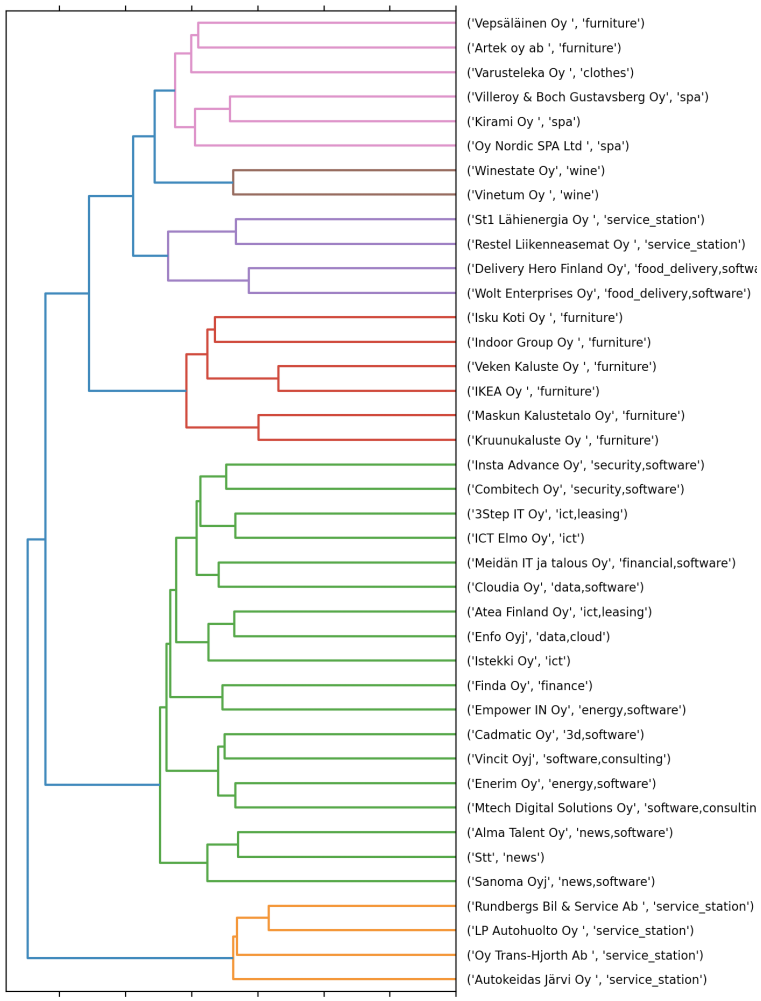

Figure 3.4: Hierarchically clustered toy data set using Ward's method and Euclidean distance metric.

I applied the same clustering method to experiment data set. Figure 3.5 shows the dendrogram produced. Of of three higher lever clusters, only the orange cluster seem to be well-defined consisting only car dealership companies. The red and green clusters do not seem to make any sense. However, some of the lower level clusters seem to be very accurate, for example construction companies and education institutions. It seems that hierarchical clustering is not able to construct meaningful hierarchical industrial classification system, at least on the experiment data set. It still may be useful to detect clusters but the dendrogram needs to be cut at the right fuse. Even the hierarchical clustering does not require number of clusters defined beforehand at clustering stage, the number of clusters must be set to extract clusters. Therefore, in this particular problem the advantages of hierarchical clustering are merely theoretical.

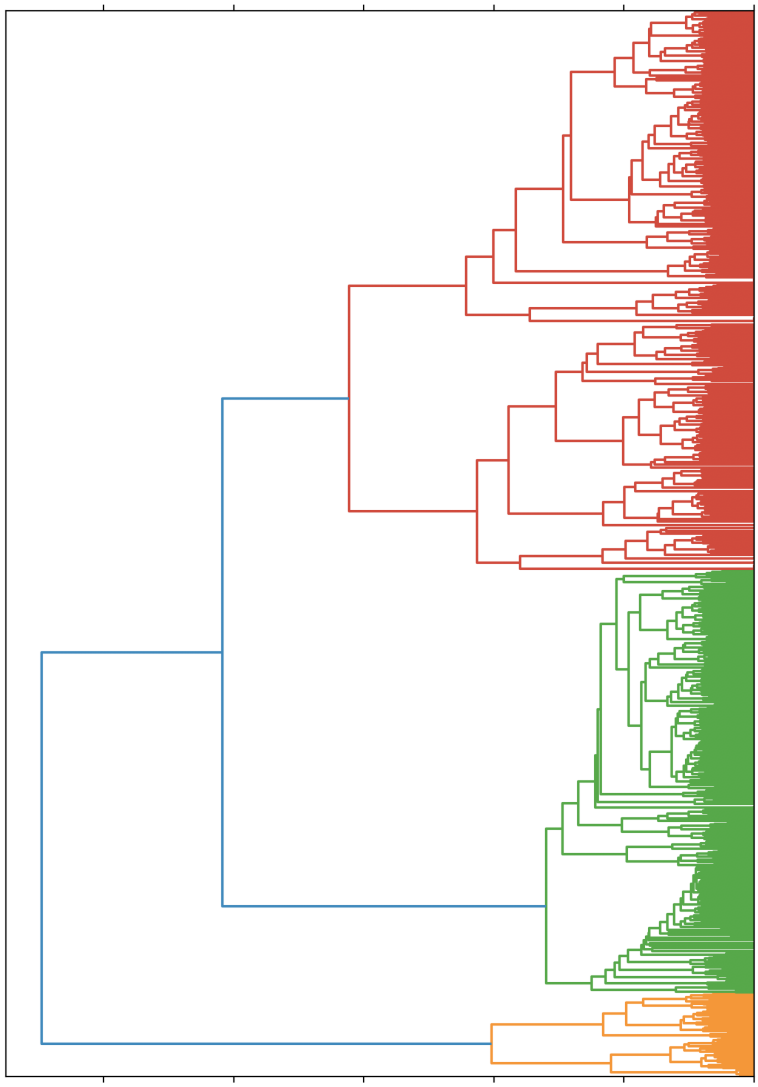

Figure 3.5: Hierarchically clustered experiment data set using Ward's method and Euclidean distance metric. X-axis expresses dissimilarity between clusters.

K-means clustering

As mentioned in Chapter 2, when using K-means clustering, the number of clusters needs to be defined beforehand. Previously measured silhouette scores indicate that for the toy data set 8 or 9 clusters would be optimal, while for the experiment data set, 2 clusters would the best fit. For the experiment, I used `KMeans` implementation of Scikit-learn library (Scikit-learn, Developers, 2021d). First, I used K-means to cluster the toy data set to 8 clusters. Because the data set was hand picked and labelled, I could track if a company was assigned to a relevant cluster. Most of the companies were assigned to meaningful clusters like service stations, media companies ar wine wholesalers. However,

these result are most likely biased because the data set is so small and was constructed for this purpose. Next, I applied K-means to cluster the 805 companies of the experiment data set. As the silhouette score measurements advised, I set the number of clusters to two. The clusters found are car dealerships and others. Figure 3.6 shows the two clusters: yellow dots mark the car dealership clusters and lilac dots mark the other. I went through the companies in the car dealership cluster and there was no mistakes except one sample which website address pointed to a wrong company website. So this error was not due the K-means clustering, but would be definitely seen as a mistake in a real-life application and is an important concern of the design. At this point, the car dealership cluster looks well-defined and could be labelled. But what to do with the other cluster? Obviously, I could calculate silhouette scores again and apply K-means clustering to it. As mentioned earlier in Chapter 2, K-means clustering is able to recognise only certain types of clusters (Ester et al., 1996). To detect non-convex clusters, DBSCAN can be used.

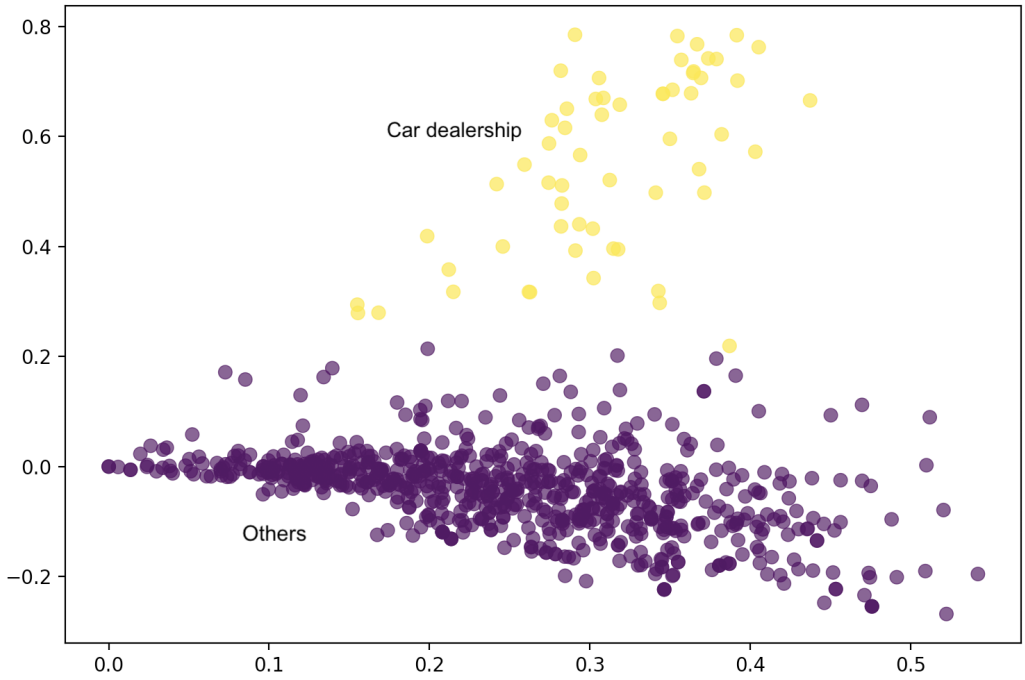

Figure 3.6: K-means clustering: the experiment data set with 2 clusters

DBSCAN

I wanted to see if DBSCAN can detect clusters that K-means cannot. I applied Scikit-learn `DBSCAN` implementation to both toy and experiment data sets (Scikit-learn, Developers, 2021c). First, I needed to find an optimal value for epsilon. I used Scikit-learn

`NearestNeighbors` implementation to find the value for epsilon (Scikit-learn, Developers, 2021e). For the toy data set, no clear "elbow" was seen in the plot. Eventually DBSCAN categorized all of the samples as noise for different values of epsilon. I suppose the data set was too small for the DBSCAN. However, for the experiment data set, an elbow could be seen around value 0.09. I set minPts parameter to 5, which simply sets the smallest allowed cluster size. DBSCAN found 10 clusters which can be seen in Figure 3.7. Samples considered as noise is marked with black dots. After inspecting the samples in the clusters, I found out that smaller clusters are relatively well-defined while the largest cluster (red) does not make sense. The largest meaningful cluster (orange) consists of car dealerships, however the noise (black dots) around it are also car dealerships. Other smaller clusters contain car or construction related companies which indicates that at this point the clusters found with DBSCAN are too fine-grained. I tried to set minPts parameter to 10. This time DBSCAN found three clusters: construction companies, car dealerships and the large clusters similar that can be seen in Figure 3.7 plus noise. It seems that DBSCAN detects more fine-grained clusters compared to K-means.

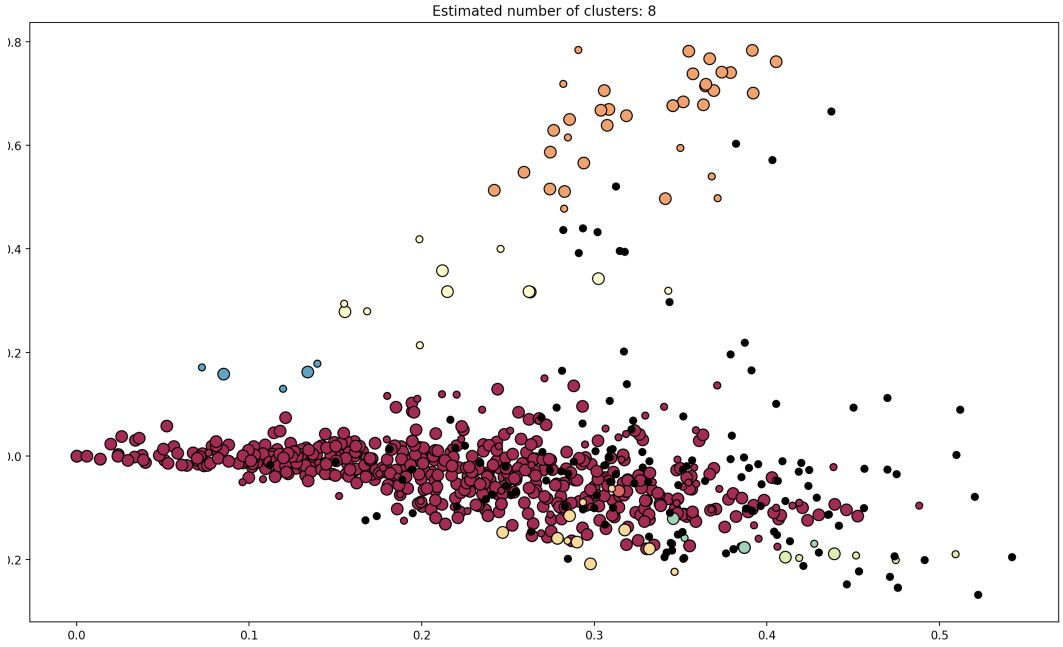

Figure 3.7: The experiment data set clustered with DBSCAN: 7 smaller and 1 large cluster (red). Noise is marked by black dots.

Interactive Clustering

Different clustering methods have different advantages: seems that clustering methods are able to find meaningful clusters from natural language data gathered from company websites. However, with tested clustering methods, fully automatic clustering does not lead to accurate enough results in this particular problem. It seemed logical that the accuracy could be improved if a human could make the decision if a cluster should be formed or not. This method is actually called *interactive clustering*. It can be useful when the problem is too complex to be solved fully automatically using clustering methods (Bae et al., 2020).

To experiment with interactive clustering on the experiment data set, I programmed two interactive clustering tools: the first using K-means together with DBSCAN and the second using only Hierarchical Clustering. K-means and DBSCAN implementations used were the same Scikit-learn implementations as mentioned in the previous section. Hierarchical Clustering implementation used was Scikit-learn implementation `AgglomerativeClustering()` which can be used to label n clusters (Scikit-learn, Developers, 2021a). I wanted to see if a hybrid of two clustering methods can outperform a single clustering method. Chen et al. (2005) proposed a hybrid of hierarchical and K-means clustering to cluster gene expression data set. They mention that clusters found using the hybrid method provided more meaningful results. Therefore, I wanted to try similar method.

Interactive clustering happened in 7 steps:

1. Form a Tf-idf feature matrix from the pre-processed samples.
2. Reduce the feature matrix to 5 components with Truncated SVD.
3. Calculate silhouette scores for a sufficient number of clusters.
4. Set the K number of clusters as the highest silhouette score.
5. Cluster data to K clusters with selected clustering method. Also, DBSCAN requires estimating Epsilon value at this step.
6. Examine clusters and label meaningful clusters. If no more meaningful clusters are appearing, save labelled clusters and stop.
7. Store labelled clusters, remove those from the data set and go back to step 3.

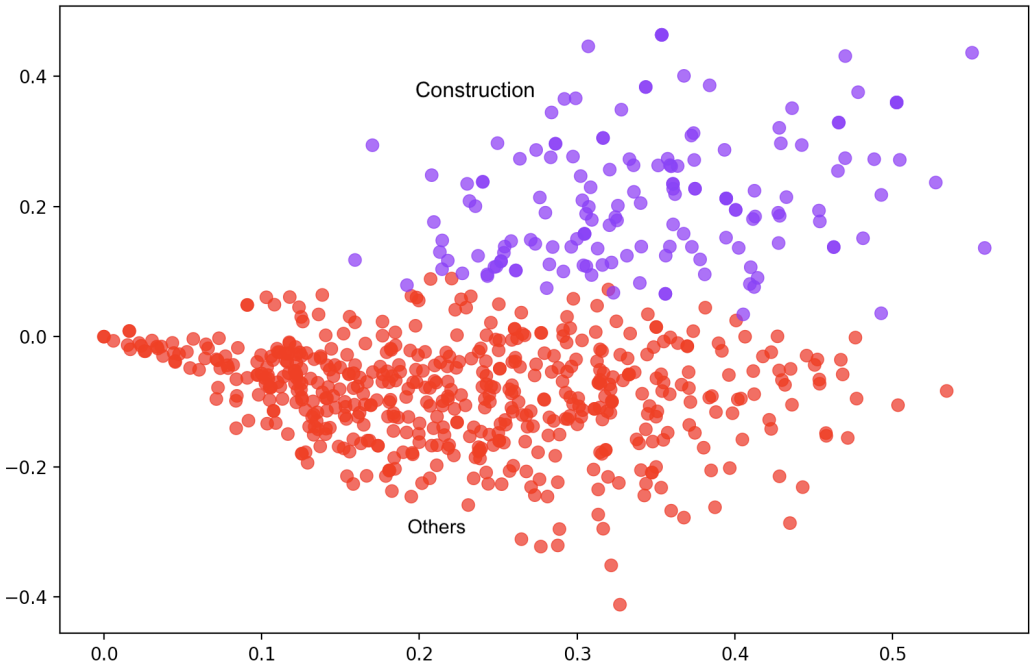

Figure 3.8: K-means clustering: car dealerships are removed from the data set. K-means detects a new cluster of construction companies.

For example, Figure 3.8 shows how K-means detects very well formed clusters of construction companies after car dealerships were extracted from the experiment data set. Also DBSCAN found new clusters as the interactive clustering proceeded. The second method using only hierarchical clustering produced also meaningful clusters. Both methods seemed to find very similar clusters from the data. Both methods reduced the number of classes significantly compared to the number of distinct TOL classes, 182 in the data experiment data set. Table 3.2 shows the key results for both clustering methods. It appears that K-means combined with DBSCAN yields better results considering higher percentage of clustered samples and higher number of clusters.

3.3 Analysis

In previous sections of this chapter I interviewed customers, optimized pre-processor and tried clustering methods on natural language data sets. Customer interviews revealed valuable insights about their needs on industrial classifications. In the interviews, it was found out that customers would need both more detailed and more simpler industrial classification system. As number of clusters extracted from the experiment data set was

Table 3.2: Results of the interactive clustering for the experiment data set

	K-means and DBSCAN	Hierarchical Clustering
Clustered - Not clustered	613 - 192	500 - 305
Clustered percentage	76%	62%
Number of clusters	48	22
Relative change in number of classes compared to TOL classes	-77%	-88%

77-88% lower than the number of distinct TOL codes, I assume the industrial classification system created with clustering would be much simpler. Interviewees also mentioned that it would be valuable if the new industrial classification system would be able to classify companies into multiple classes. However hierarchical clustering, the only potential clustering method tested able to classify companies into multiple (hierarchical) classes, turned out to be useless. The hierarchies method detected made simply no sense. Two more requirements were found out during the interviews: the classification system should be updated regularly and it should be able to correct wrongly assigned TOL classes. The clustering process can be performed again at anytime, but it requires the data has changed to see any kind of change in the classifications. When it comes to industrial classifications, it most likely takes some time to see changes in big picture, possibly years. Labelling a newly founded company would be more frequently occurring task. With clustering, wrongly assigned TOL classes will not be a problem: the method outlined in this thesis is completely independent from TOL classes and can correct false TOL classes.

In Chapter 1, three research questions were defined. RQ1 asked, what are requirements for the new industrial classification system while RQ3 asked if a such system can be created using unsupervised NLP techniques (clustering). Table 3.3 summarizes the requirements as well as the answer to RQ3 considering each requirement. Altogether, the experiments showed that an industrial classification system created using clustering, will most likely be much simpler compared to existing TOL 2008 industrial classification system.

RQ2 asked how to clusters natural language data set in practise. Natural language processing starts by identifying relevant data sources. I experimented trying different clustering methods on Finnish language data extracted from company websites. It was discovered, that using more pages inside the company web domain does not necessarily improve the

Table 3.3: Requirements of the new industrial classification system reviewed from the perspective of clustering experiments.

Requirement	Can the requirement be satisfied?
The system should be more detailed	No
The system should be simpler	Yes
Classification of the system should be updated regularly	Yes
The system should be able to classify companies to many classes	No
The system should be able to correct wrongly assigned TOL codes	Yes

results, compared using just the front page. Also, the pre-processing of language data will affect the results. With Finnish language, filtering out all but nouns and splitting the compounds produced the highest silhouette scores. After pre-processing documents (websites), they are represented as a feature matrix. Normalized Tf-idf feature matrix is state of the art transformation used in NLP. It will give more weight to words that differentiate documents from each other by lowering the weights of the words appearing frequently in the collection of documents. When applying clustering methods to the feature matrix, dimensionality reduction techniques, like SVD, are beneficial for a couple of reasons. First, it will improve the performance of clustering process as it reduces the number of columns significantly. Second, dimensionality reduction can solve the curse of dimensionality which was seen especially with K-means clustering as it was not able to form meaningful clusters before the reducing the feature matrix to five components.

I tried three different clustering methods on both toy and experiment data sets. Clustering methods were not able cluster data into meaningful clusters fully automatically. However, using clustering methods in parallel by extracting clusters one by one, I was able to cluster most of the samples to meaningful clusters. A hybrid of method using K-means and DBSCAN parallel produced slightly better results than Hierarchical Clustering alone. Table 3.4 summarizes recommended NLP techniques on each step of the NLP pipeline. In the next chapter, the whole study will be evaluated from perspective of design science which was introduced Chapter 2.

Table 3.4: Recommendations for the steps of NLP pipeline

Step	Recommendation
Data collection	It seems that preferring quality over quantity will give better results, for example use only front pages of company websites.
Pre-processor	Use only nouns and split compounds (applies to Finnish language).
Feature engineering	Use Tf-idf transformation and dimensionality reduction technique, for example SVD.
Clustering methods	K-means used together with DBSCAN can produce higher percentage of clustered samples.
Clustering strategy	Use interactive clustering to extract meaningful clusters.

4 Discussion

Let us recall the most important terms of the design science process. First, the design science method defines 7 guidelines for the design process. Design process should produce a design product which components are requirements, design, kernel theory and testable design hypothesis. As a reminder, design component is consisted of four artifacts: construct, model, method and instantiation. This chapter walks through the design process concluded in the thesis in relation to the seven guidelines. Also, the artifacts produced are pointed out. Finally, the design hypothesis is evaluated based on the requirements and results found in Chapter 3.

The first guideline Hevner et al. (2004) mentions is that design process should produce an applicable artifact. The most practical artifact (instantiation) produced, is the proof of concept (POC) clustering experiment that resides in a public code repository (Hyttinen, 2022). Additionally, this study introduces key terms (construct) to understand the instantiation, provides a diagram describing the structure of NLP pipeline (model) and walks through the NLP process in detail (method).

Guideline number two states that the design science process should solve real business problem with new technology. This thesis was written as commission to Alma Media Oy as the topic was introduced by the head of product. Hence, I can say that the problem is strongly connected to a real business problem. To make sure that the design process would focus on the real problem, customers who had experienced problems with TOL 2008 industrial classifications were interviewed. By analyzing the interviews, meta-requirements of the new industrial classification system were set. The motivation for interviews (kernel theory), was that a customer-oriented design will lead to a higher business success. To implement NLP proof of concept, Sickit-Learn unsupervised machine learning library was used which is relatively new technology at the time of writing.

Third, design artifact should be evaluated rigorously. Hevner et al. (2004) mentioned five types of evaluation methods which all was used. Customers were interviewed to find out typical problems with TOL 2008 industrial classifications (observational). Experiment with real Finnish language text data collected from company websites, was conducted. The experiment and interviews can be seen as an hybrid of case study and controlled experiment (observational and experimental). During the experiment, number of clusters, length of

data set and pre-processor parameters were optimized by comparing the silhouette scores (analytical). Finally clustering algorithms were tested to see how classifications relate to real life. This can be seen as white box testing, because it can be tracked which words determined certain cluster. The design process is documented in this thesis which can be seen as one extensive informed argument (descriptive). Also, performance of the clustering methods was analytically evaluated.

Design science process should contribute at least in some of the following fields: design artifact, knowledge or evaluation. This thesis contributed mostly in the fields of design artifact and knowledge. As mentioned previously, the design artifact should solve a real-life business problem. The customer interviews helped to define the business problem associated with industrial classifications (knowledge). As this study acts as the first step of the design process, it is too early to evaluate the customer success of the design artifact. Clearly, unsupervised learning can be used to discover industrial classifications from the natural language data, but it does not alone provide solutions to complicated customer issues. As the clustering experiment showed, unsupervised learning can help to produce a much more simpler industrial classification system.

The fifth guideline instructs that rigorous methods should be used in the evaluation of the design artifact. As mentioned in Chapter 2, there are five types of design evaluation methods: observational, analytical, experimental, testing and descriptive. First, the meta-requirements of the design artifact were set using observational method, qualitative customer interviews. Pre-processor parameters were optimized analytically comparing silhouette scores on the toy data. Next, a clustering experiment was conducted using Finnish language on front pages of 805 company websites. Clustering strategies were compared with different metrics, for example percentage of samples clustered and number of clusters formed. Overall, design evaluation methods were used in a versatile manner. Additionally, the design artifact is described both in programming language and in English is this thesis.

The sixth guideline reminds that the design process is an iterative process. This study acts as the first cycle of this process. From business perspective the results of this thesis are crucial. The further design process proceeds, the more resources it requires. As the designer of new method presented in this study, it is my responsibly to present the result objectively so the stakeholders can make the best business decisions about the future design work.

The last guideline mentions that results of the design science process should be reported understandably to both technical- and business-oriented audiences. As an author, I have

tried to write this thesis to be understandable both intuitively and formally. For business-oriented audiences, I have prepared a presentation that summarizes the design process on a high-level. Technical audiences are taken into account by providing the source code of the clustering experiment and referencing the documentation of the programming libraries used in the thesis.

5 Conclusions

The research problem introduced in Chapter 1.4 stated the following: *"What is needed from a useful industrial classification system and what kind of process can automatize the creation of it?"* This thesis outlined the process of producing industrial classifications from Finnish language data. I briefly demonstrated how to prepare Finnish language for unsupervised learning (clustering) and then experimented with different clustering methods. The clustering experiment carried out gave insights about the clustering: industrial classifications extracted from the experiment data set were mostly relevant and the number of classes much lower compared to TOL 2008 classes. On the other hand, experiments showed that some level of inaccuracy is expected when clustering natural language data set gathered from company websites. Some portion of the samples (24-38% in my experiments) did not end up in meaningful clusters and some samples ended up in wrong clusters. Luckily, the number of misclassifications was very small. If we look at the problem from customer's perspective, false information is far more harmful than knowing the information is missing. That is the reason why applying clustering methods without human interaction seemed problematic: automatic clustering produces too many misclassifications. Therefore, I suggested interactive clustering strategy where the quality of clusters is supervised by a human. One drawback of the interactive clustering is that the clustering algorithms need to be run multiple times which can be time consuming. On the other hand, clustering algorithms would no be needed to run very often.

Software design is a combination of strategical business decisions and formal problem solving skills: commonly programming but in the case of this study, also mathematics, statistics and business skills. Two important advantages of intangible software products are that they can be scaled with low costs and can be delivered to almost anywhere in the world with almost no costs. However, different languages introduce some problems. The clustering experiment was carried out with Finnish language data but in reality, different languages need to be taken into account. For example, if pre-processed language data sets are in different languages, it is not possible to use them together. It would require a translation process before the data could be used together. This is a good example of practical problems can affect the technical solutions greatly. Nevertheless, the results of the clustering experiment are still valid despite of the language used. But for future work, the language of the data sets should be taken into account. In 2022, programmatic

translation is relatively cheap and the quality of translations have improved during last years. All of the data could be for example, translated into English. That would enable the use of English language pre-processing libraries.

Clustering proved to be helpful unsupervised learning method to produce simpler classification system. The customer interviews indicated that there is a need for multiple industrial classifications per company. Based on the clustering experiments conducted in this study, unsupervised learning methods are not able to detect multiple classes at least from natural language data extracted from company websites. It might be possible with supervised learning methods but these methods require labelled data. Therefore, I leave this question open for future research.

Bibliography

Alasuutari, P. (2011). *Laadullinen tutkimus 2.0.* 4th ed. Vastapaino.

Bae, J., Helldin, T., Riveiro, M., Nowaczyk, S., Bouguelia, M.-R., and Falkman, G. (Feb. 2020). "Interactive Clustering: A Comprehensive Review". In: *ACM Computing Surveys* 53.1, pp. 01–39. DOI: 10.1145/3340960.

Bird, S., Klein, E., and Loper, E. (2009). *Natural language processing with Python: analyzing text with the natural language toolkit.* 1st ed. O'Reilly Media, Inc.

Cantamessa, M., Gatteschi, V., Perboli, G., and Rosano, M. (July 2018). "Startups' Roads to Failure". In: *Sustainability* 10.7, p. 2346. DOI: 10.3390/su10072346.

Chen, B., Tai, P., Harrison, R., and Pan, Y. (2005). "Novel hybrid hierarchical-K-means clustering method (H-K-means) for microarray analysis". In: *2005 IEEE Computational Systems Bioinformatics Conference - Workshops (CSBW'05)*, pp. 105–108. DOI: 10.1109/CSBW.2005.98.

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise". In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining.* KDD'96. Portland, Oregon: AAAI Press, pp. 226–231.

Hevner, A., R, A., March, S., T, S., Park, Park, J., Ram, and Sudha (2004). "Design Science in Information Systems Research". In: *MIS Quarterly* 28, pp. 75–105.

Hyttinen, M. (2022). *Clustering experiments code repository.* URL: https://github.com/miikahyttinen/masters_thesis_2021 (visited on 08/31/2022).

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2021). *An introduction to statistical learning.* 2nd ed. Vol. 112. Springer.

Kettunen, K. (2006). "Developing an automatic linguistic truncation operator for best-match retrieval of Finnish in inflected word form text database indexes". In: *J. Information Science* 32, pp. 465–479. DOI: 10.1177/0165551506066057.

Klee, V. (1971). "What is a convex set?" In: *The American Mathematical Monthly* 78.6, pp. 616–631.

Lloyd, S. (1982). "Least squares quantization in PCM". In: *IEEE Transactions on Information Theory* 28.2, pp. 129–137. DOI: 10.1109/TIT.1982.1056489.

Lukas, B. and Ferrell, O. (2000). "The Effect of Market Orientation on Product Innovation". In: *Journal of the Academy of Marketing Science* 28, pp. 239–247. DOI: 10.1177/0092070300282005.

March, S. and Smith, G. (Dec. 1995). "Design and Natural Science Research on Information Technology". In: *Decision Support Systems* 15, pp. 251–266. DOI: 10.1016/0167-9236(94)00041-2.

NLTK Project (2021). *NLTK 3.6.3 documentation*. URL: https://www.nltk.org/ (visited on 11/12/2021).

Pitkänen, H. (2021). *Voikko 4.3 documentation*. URL: https://voikko.puimula.org/ (visited on 10/26/2021).

Porter, M. F. (1980). "An algorithm for suffix stripping". In: *Program* 14.3, pp. 130–137.

Python Software Foundation (2021). *Python 3.9.7 documentation*. https://docs.python.org/3/. (Visited on 11/12/2021).

Richardson, L. (2022). *Beautiful Soup 4.9.0 documentation*. URL: https://www.crummy.com/software/BeautifulSoup/bs4/doc/ (visited on 03/23/2022).

Sarkar, D. (2016). *Text Analytics with Python*. Springer.

Schaefer, H. H. (1971). "Topological vector spaces". In: *Graduate texts in mathematics*. Vol. 3. New York: Springer.

Schubert, E., Sander, J., Ester, M., Kriegel, H. P., and Xu, X. (2017). "DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN". In: *ACM Trans. Database Syst.* 42.3, pp. 01–21. DOI: 10.1145/3068335.

Scikit-learn, Developers (2021a). *Scikit Agglomerative Clustering*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html (visited on 12/14/2021).

– (2021b). *Scikit CountVectorizer*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html (visited on 12/10/2021).

– (2021c). *Scikit DBSCAN*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html (visited on 12/14/2021).

– (2021d). *Scikit K-means*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html (visited on 12/14/2021).

– (2021e). *Scikit Nearest Neighbours*. URL: https://scikit-learn.org/stable/modules/neighbors.html (visited on 12/14/2021).

– (2021f). *Scikit Silhouette Analaysis*. URL: https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html (visited on 12/13/2021).
– (2021g). *Scikit TfidfTransforme*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfTransformer.html (visited on 11/26/2021).
– (2021h). *Scikit TruncatedSVD*. URL: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html (visited on 12/14/2021).

Simon, H. A. (1996). *The sciences of the artificial*. eng. 3rd ed. Cambridge (MA): MIT Press. ISBN: 978-0-262-69191-8.

The SciPy community (2021). *Sicpy Ward's Linkage*. URL: https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.ward.html.

Tilastokeskus (2021). *TOL 2008*. URL: https://www2.stat.fi/fi/luokitukset/toimiala/ (visited on 11/04/2021).

Unicode, I. (2021). *Announcing The Unicode Standard, Version 14.0*. URL: https://home.unicode.org/announcing-the-unicode-standard-version-14-0/ (visited on 11/12/2021).

Walls, J., G., J., Widmeyer, R., G., Sawy, O., and A., O. (1992). "Building an Information System Design Theory for Vigilant EIS". In: *Information Systems Research* 3, pp. 36–59. DOI: 10.1287/isre.3.1.36.

Appendix A Interview A

Author:

Kun haastattelut on tehty, tarkoitus on lähteä tutkimaan datalähteitä ja miettimään minkälaista tekoälyä päättelyyn saisi tehtyä.

Interviewee A1:

Teillä on Almalla yhteistyötä ja Inderesin kanssa, joka tekee syvempiä toimialakatsauksia. Sieltä voisi käyttää datalähteitä. Tämmöinen kohderyhmä (asiakkaan oma kohderyhmä), et saa TOL 2008 –toimialaluokituksella millään vaan se vaatii kaivamista ja raporttien lukemista.

Author:

Niin kuin vastasit jo, ensimmäinen kysymys siis kuului, että minkälaisia ongelmia TOL-2008 luokitus aiheuttaa liiketoiminnalle. Mainitsit esimerkiksi, että SaaS firmoja ei löydä. Tuleeko muita mieleen?

Interviewee A1:

Juurikin näin. Meillä on tietyt toimialat, joilla olemme vahvoja. (tarkoittanee liiketoimintaa) Näistä yhdeksän kymmenestä on koluttu, mutta kun täytyy ruveta etsimään tietyn toimialan sisältä, tässä tapauksessa IT-firmoista SaaS-tilitoimisto-comboja, niin se vaatii Inderes-raportteja, joissa otetaan kantaa myös toimialan kehitykseen. Ja sieltä löytyy poimintoja, että ketkä tämän toimijan kilpailijat ovat.

Author:

Eli nämä eivät ole tarpeeksi tarkkoja.

Interviewee A2:

Jos mennään IT-kategoriaan, niin palveluntarjoaja ja ohjelmistontarjoaja ovat samassa kategoriassa. Tämä on se ongelma. Toinen on ostaja, joka käyttää toisen ohjelmistoa ja toinen on ohjelmiston hyödyntäjä. Sitä TOL-2008 erota. Mediassa taas mitä mediaa sä tuotat? Sähköinen media, printtimedia, aikakauslehti. Samassa kategoriassa kaikki.

Author:

Tuntuu, että TOL-2008 luokitus ei ole pysynyt digitaalisen kehityksen vauhdissa. Jos mietitään esim. maataloustoimialaa, niin siellä luokittelut löytyvät melkeinpä viljelykasvia

myöten.

Interviewee A2:

TOL-2008 luokituksen suurin käyttäjä on Tulli. Siis ulkomaankauppa. Sieltä tulee tullinimikkeet ja IT ei maksa tulleja. Sen takia se on jäänyt jälkeen. Jos te saatte esim. Inderesraportteja mukaan te pystytte pilkkomaan isoja toimiala pienemmiksi ja selkeämmiksi ryhmiksi. Mikä on se ryhmä ison TOL-luokituksen sisällä.

Interviewee A3:

Nostaisin keskusteluun esimerkkinä isot yritykset, jotka toimivat usealla toimialalla. Näillä on vain yksi TOL-luokitus. Esimerkiksi meidän TOL on "Sanomalehtien kustantaminen", mutta tehdäänhän me ihan valtavasti muutakin.

Author:

Jep, pitäisi olla useita

Interviewee A3:

Toimialakuvauksessa vähän laajemmin kerrottu, mutta ei sekään välttämättä kerro ihan kaikkea.

Interviewee A2:

Ja siellä voi olla sanottu "ja muut tähän toimialaan liittyvät toiminnot". Se tulee taas PRHstä.

Author:

Sitten kakkoskysymys, jota jo sivuttiinkin eli minkälaisia luokituksia pitäisi olla.

Interviewee A1:

Eli segmentoitua tietoa toimialan sisältä. Niitä ei saa virallisesta rekisteristä vaan vaikkapa taloushallintoliitolta ja Inderesin raportteja tutkimalla.

Author:

Ja kun tuon prosessin saisi automatisoitua, eikä tarvitsisi käsin penkoa.

Interviewee A1:

Teidän pitäisi jutella Inderesin kanssa.

Author:

Tästäkin jo puhuttiin, eli kolmas kysymys oli, että miten hyödyntäisitte tuollaista toimi-

alaluokitusta. Segmentointiin ainakin.

Interviewee A1:

Ideana on se, että soitetaan meidän myynnistä suoraan 10 tai 15, mutta laajemmin sitten ostettaisi ostopalveluna soitot pienemmille yrityksille. Julkiselta sektorilta kartoitetaan milloin on kilpailutus, kuka on nykyinen toimittaja ja minkälaiset volyymit. Pitäisi lähteä soitto, sitten toimialaspesifi esite ja toimialaspesifi sähköposti. Eli kerrotaan miten voidaan olla hyödyksi teille.

Interviewee A2:

Mitä kollega mainitsi. Kannasta pysytään poimimaan sanotan 300 yritystä. Oikeasti meidän kohderyhmä on 130. Halutaan antaa tämä jollekin soitettavaksi, että etsivät sen oikean henkilön. Eli tekevät sen kartoituksen ja me jatkamme sitten siitä. Eli tässä korostuu se henkilön tehtävä ja toimiala. Meidän kohderyhmä on Suomessa satoja yrityksiä, ei tuhansia yrityksiä. Me tiedetään tarkkaan ketä ne ovat, mutta ei pystytä neljän hengen voimin soittamaan niitä.

Appendix B Interview B

Author:

Q1: Minkälaista käytännön ongelmia TOL-2008 luokka on aiheuttanut sinun työssäsi?

Interviewee B:

Omasta kokemuksesta ongelmaa tulee siitä, kun halutaan muodostaa helposti ymmärrettäviä ja käytännönläheisiä toimialoja. Esimerkiksi ohjelmistojen suunnittelu ja valmistus ei ole ehkä sellainen. Ajatellaan, että halutaan tehdä juttu, joka tuottaa lisäarvoa esimerkiksi koulutusyrityksistä; sen sisällä (TOL 2008: P Koulutus) on vaikkapa julkisia tahoja, kuten peruskoulut, lukiot jne. Saman toimialan sisällä on myös yksityiset toimijat, kuten valmennuskurssit ja vaikkapa pörssiyritys Soprano Oyj. Voisin kuvitella, että jos Seduo olisi oma firmansa niin se olisi toimialan sisällä, mutta eipä ole. Alma Talentin (omistajan) toimiala on sanomalehtien kustantaminen.

Author:

Eli tässäkin Seduo on Alma Talentin sisällä.

Interviewee B:

Tällaisiahan asioita TOL 2008 ei kerro. Eli ne pitää vaan tietää. Olisi tosi hyvä, jos tässä olisi joku työkalu auttamassa. Suomessahan asiantuntija/analyytikko/yrittäjä voi tuntea toimialan ja kilpailijat melko hyvin, mutta heti jos halutaan tehdä Euroopan kokoinen katsaus aiheesta, niin harva enää tietää mitä Ruotsissa tapahtuu. Monien yritysten verkkosivut ovat englanniksi. Eli jos sen (luokittelun) pystyy tekemään verkkosivuilta, niin (Euroopankin kontekstissa?) luulisi, että sitä tietoa olisi saatavilla.

Author:

Tuo oli mielenkiintoinen, kun otit Alman puheeksi. Almakin on konserni ja konsernin sisällä olevia palasia tulisi usein käsitellä omina ”yrityksinä”.

Interviewee B:

Hyvä kysymys on, luokittelisitko Alma Talentin koulutusyritykseksi?

Author:

Pitäisi varmaan luokitella.

Interviewee B:

Niin ei se varmaan helppoa ole, jos pitäisi löytää vain yksi toimiala.

Author:

Tämän uuden toimialanluokituksen ajatus on, että niitä voisi olla useita yhdellä yrityksellä.

Interviewee B:

Siinä tapauksessa, jos nyt katsotaan, vaikka Alma Talentia. Eli siellä on toimialakuvaus, joka voi pitää sisällään kaiken (useita toimialoja). Päätoimiala (TOL 2008) on siis sanomalehtien kustantaminen. Tässä toimialakuvauksessa on kyllä kaikenlaista mm. konsultointi, tutkimuspalvelut, markkinointi eli firma voisi tehdä melkein mitä vaan tämän perusteella. Nyt jos tekisin untuvikkona selvitystä koulutusyrityksistä, niin tekisin varmaan poiminnan, joka on lähtökohtaisesti TOL-pohjainen. Nythän sitten Alma Talent jäisi pois tästä, koska päätoimiala (TOL 2008) ei viittaa koulutukseen. Että löytää Alman tuohon, niin tarvitsee sen tiedon jostain muualta.

Author:

Toimialakuvaushan voi olla ihan mielenkiintoinen datalähde.

Interviewee B:

Niin kyllä tuohon toimialakuvaukseen on houkutus kuvailla kaikki mahdollinen. Toki kannattaa varmasti tutkia. Jos taas mietitään verkkosivuja, niin yrityksillä on motiivi kirjata tiettyjä asioita. Asiakas ei tietenkään halua lukea toimialakatsausta. Ja se (verkkosivu) on myös markkinointiteksti.

Author:

Jep. Nämä toimialakuvaukset ovat aika paljon tuota ”kaikki laillinen liiketoiminta” -osastoa.

Interviewee B:

Yksittäisiä yrityksiä tarkasteltaessa tuntuu et siihen on lyöty ihan kaikki ja vähän ohikin.

Author:

Tuntuisi, että markkinointimateriaali (kotisivut) voi olla paljon kuvaavampaakin, koska sen funtsimiseen laitetaan aikaa.

Interviewee B:

Se on sitten tavallaan tavoitteellista tekstiä, eli yrityksellä on joku tarkoitus kun se sanoo asian juuri sillä tavalla. Tosin esim. pörssiyrityksillä on velvoitteita kertoa asioista tietyllä

tavalla ja tiettynä aikana. Mutta jos mennään perinteisiin Pk-yrityksiin niin sääntely ei ole enää niin tiukkaa. Tuskin on esim. mitään sijoittajasuhdesivuja.

Author:

Tuo on hyvä pointti ja voisi ajatella, että rajaa kokeilusta pörssiyritykset pois (tai kokeilee niitä erikseen).

Interviewee B:

Niin en ehkä pois jättäisi, mutta pitää miettiä haluaako käyttää sijoittajasuhdesivuja. Usein pörssiyrityksillä nyt on paremmat sivut, kuin perus Pk-yrityksillä.

Jos saat muodostettua lopulta jotain järkeviä toimialaluokkia, niin voin kuvitella, että siitä olisi paljon apua. Relevantin lähdedatan löytäminen voi olla se haaste.

Author:

Q2: Minkälaisia luokkia olisit työssäsi tarvinnut, jos saisit toivoa?

Interviewee B:

Toimituksen näkökulmasta sellaisia mistä tehdään juttuja esim. nämä koulutusyritykset. Jos haluaisin (toimittajana) esitellä kaikkien kaupallisten koulutusyritysten tunnuslukuja. Niin se ei TOL 2008-luokituksella onnistu. Yliopistoillakin on erilaisia yritysjohdolle suunnattuja kaupallisia koulutuksia. Jos taas vain 5 % yrityksen liikevaihdosta koostuu koulutuspalveluista, onko se koulutusyritys? Minun mielestäni ei ole, koska tuo on niin marginaalinen osuus. Toisaalta sitten jonkun yrityksen liikevaihdosta 60–70 % muodostuu koulutuspalveluista, mutta sitten sen TOL-Luokka voi silti olla ohjelmistojen suunnittelu ja valmistus.

Käytännönläheisyys on se oleellinen juttu tässä. Jos haluaisin tietää kaikki valmennuskurssien tarjoajat, niin se (valmennuskurssit) olisi hyvä käytännönläheinen toimiala.

Author:

Se on hyvä kysymys mihin raja vedetään. Jos otetaan vaan tämän uuden luokituksen tarjoamat kolme todennäköisintä voi olla, että eka on ok, mutta kaksi seuraavaa on täysin hakoteillä. Pitäisikö sitten seurata mistä liikevaihto tulee?

Interviewee B:

Sehän riippuu käyttötarkoituksesta. Jos seurataan talouslukuja koulutustoimialalla, ei ole kovin mielekästä ottaa tarkasteluun yrityksiä, joiden liikevaihdosta 5

Author:

Q3: Miten olisit hyödyntänyt niitä luokkia?

Haen tässä nyt sellaista käytännön työn perustelua tälle tutkimukselleni. Esimerkiksi jos sinulla olisi ollut haluamasi luokitus käytössä olisiko se suoraviivaistanut työtäsi jollain tavalla?

Interviewee B:

Jos haluat tietää, että Alma Talent tekee koulutusta, niin se vaan ”pitää tietää”. Jos tehdään vaikka juttu ”Suomen kymmenen suurinta B2B-koulutksen tarjoajaa”, niin TOL 2008-luokituksen avulla se ei ole mitenkään mahdollista. Jos otetaan Alma Talent siihen, niin miten määrität, että paljonko sen liikevaihdosta on koulutusta? Niin vaikka katsoisit TOL:ia tai nettisivuja, niin silti taloudellinen realiteetti on olemassa. (Yritys voi sanoa tekevänsä vaikka mitä, mutta jos myyntiä ei ole tekeekö se sitä?)

Author:

Onko tilinpäätöksissä eroteltu mistä liikevaihto tulee? Voidaanko niistä päätellä mitään toimialasta?

Interviewee B:

Toimintakertomuksesta ainakin voi päätellä jotakin. Tietysti kun mennään pienempiin yrityksiin, niin tuo informaatio vähenee.

Author:

Missä se rajaa muuten menee, että toimintakertomus pitää tehdä? (Tarkistuksen jälkeen) Se oli 7,3 liikevaihtoa tai taseen loppusumma 3,6 miljoonaa. Voisikin tutkia, että monta prosenttia Suomen yrityksistä on velvollisia antamaan toimintakertomuksen.

Interviewee B:

Tuli mieleen IT-yritykset, että voisi olla mielekästä jaotella niitä eri kategorioihin. Se on toimialana kuitenkin niin iso ja moniulotteinen. (t)

Author:

TOL taitaa olla vuodelta 2008. IT-alalla on tapahtunut aika paljon 13 vuodessa.

Appendix C Interview C

Author:

Tuottaako TOL jotain ongelmia?

Interviewee C:

Olemme pyrkineet luokittelemaan meidän jäsenyrityksiä luokkiin TOL 46 ja 47, autopuoli ei ole suoraan meidän jäseniä. Eli onko tukkukauppaa (46) vai vähittäiskauppa (47). Toistuvasti huomataan se, että raja on kadonnut. Yritys joka on ilmoittanut tekevänsä vähittäiskauppaa, tekeekin yritysmyyntiä (tukkukauppaa) ja toisin päin. Ero on hämärtynyt.

Verkkokauppa.com on hyvä esimerkki siitä, että se tekee kuluttaja- ja yrityskauppaa. Toisena esimerkkinä uusi yritys Oda, joka myy ruokaa ihmisille kotiovelle. Sen TOL-koodi on ”Muu päivittäistavaroiden erikoisvähittäiskauppa”. Siinä ei puhuta mitään verkkokaupasta. Meille iso ongelma on vähittäis- ja tukkukaupan ero sekä se, että kaikki tekevät verkkokauppaa, mutta meidän jäsenistä alle 10 mainitsee verkkokaupan toimialakseen. TOL ei anna niitä vastauksia meille.

Kolmas asia on se, että kaikki yritykset tekevät kauppaa. Meillä on paljon IT-alan tai teknisen tukkukaupan yrityksiä, joilla koodina on joku teollisuuden koodi. Oikeasti ne tekevät maahantuontia ja kauppaa. Yrityshän valitsee itse sen koodin koko repertuaarista. TOL-koodien valinta on todella kirjavaa, kun ihmiset perustavat yrityksiä. Lisäksi TOL-koodin ongelma on se, että se on liian yksinkertainen. Eikä kukaan edes valvo niitä. Kun meille hakee jäseniä, niin joudutaan selvittämään, että mitä ne oikein tekevät.

Tämä koronakriisi oli mielenkiintoinen. Julkisen sektorin päättäjät lähtivät liikkeelle TOL-luokituksesta; ne joilla oli tietty koodi, saivat tukea automaattisesti, kun muiden piti sitä erikseen hakea. Nykyään kaupat ja ravintolat ovat aivan sekaisin, esimerkiksi huoltoasema, jossa on ravintola samassa. Arvioisin, että puhutaan kymmenistä prosenteista, joissa TOL-koodi ei kerro mitä yritys oikeasti tekee. Lisäksi se on liian staattinen. Se on ja pysyy. Voi tosin olla, että TOL-koodeja vaihdettiin tämän tukiasian takia, jotta yritys saisi paremmin tukia.

Author:

Minkälaisia luokkia toivoisitte?

Interviewee C:

Hyvä olisi, että toimialaluokitus eläisi ajassa. Jos mietitään tukku- ja vähittäiskauppaa niin tarve olisi, että olisi joko vain toinen tai sitten molemmat. Sitten jos luokitus on väärä, niin se aiheuttaa enemmän ongelmia kuin se, että sitä ei olisi ollenkaan.

Sinänsä toimialaluokituksen ei kannata olla liian tarkka, myydäänkö vaikka lihaa tai kalaa, ne kuitenkin muuttuvat niin nopeasti. Tärkeää, että luokitus palvelee enemmistön tarpeita.

Author:

Miten hyödyntäisitte luokkia?

Interviewee C:

Jäsenhankintaan sekä koulutusmarkkinoinnin kohdentamiseen, kun tavoitetaan yrityksiä jotka eivät olet jäseniä. Olemassa olevien jäsenten kohdalla olisi tärkeää myös kohdentaa viestintää yrityksille, joita tietty viesti koskee. Meillä tehdään esimerkiksi B2B-myyntikoulutusta, niin ajattelimme ensin, että kohdennetaan se vain tukkukaupalle. Sitten huomattiin, että koulutukseen ilmoittautuu yrityksiä, joilla on vähittäiskaupan koodi. Ja kun tutkittiin näitä yrityksiä, niin ne tekevät kuitenkin myös tukkukauppaa.

Virheellinen toimialaluokitus vaikuttaa myös tilastoinnin laatuun. Vainun (kilpailija) luokitusta käytetään myös, mutta en tiedä tarkkaan, käytetäänkö sitä julkistettaviin tilastoihin vai onko se enemmän vain taustoittavaa työtä.

Toimialaluokitukset vaikuttavat myös taustalla oleviin asioihin esim. meidän hallituksen muodostamiseen. Ja nyt kun ero hämärtynyt, on käytäntöä tarvinnut muuttaa.