Master Thesis

NewsPercept: A Simple Data Science Pipeline for Online News Perception Mining

University of Bergen
Information Science

Written by

Felipe Sepulveda

Suppervised by

Christoph Trattner

November 30, 2019

Contents

List of Figures

List of Tables

1 Acknowledgement

First of all I would like to express my gratitude to my supervisor, Christoph Trattner, whose involvement, patience, wisdom and motivation where vital for the research to succeed. I know I gave you some hard times, and I must thank you from the bottom of my hart for your patience and support. I know that without your support I wouldn't be where I am today.

I would like to thank my father that motivated me through his illness of cancer halfway through my master thesis. I must admit that his illness took the best of me, and I considered to drop out to focus on my family instead. I want to thank my mother for putting me back in line and to motivate me to never give up, even in the darkest moments.

I must also thank my friends for their support, help and motivation throughout the project.

Finally I would like to thank my spouse for her support and motivation through my hard times.

"No hay la peor cosa que la que no se intenta"
"There is no worst thing than what you don't try"

(Nany Baeza - Mom)

Felipe Sepulveda

Abstract

Sentiment analysis is a contextual minding of text that is used in various fields and organizations. Through sentimental analyzing it's possible to classify the polarity of a given paragraph or sentence. The aspect of the sentence is either positive, negative, or neutral, depending on the word's written emotions. Andranik Tumasjan (2005) used Twitter to show a correlation between German political parties and the public mood regarding the landscape of the election. In the health section, Korkontzelos et al. (2016) used the media as a resource to analyze the public reaction on a drug from Tweets. By using the automatic identification of *Adverse Drug Reaction* (ADR), the results show that most of the mentioned Tweets were labeled as negative towards the product. The process of sentiment analyzing is based on several factors, one in a particular, is collection of data and what kind of media channel is best suited for the task. Different media channels like Twitter, Facebook, and newspapers have a vast selection of information from the public. The thesis presents a system architecture which uses Norwegian Newspapers as data resource and transforms it by using Natural Language Processing (NLP) and presentation. The thesis will present the possibility of knowledge-driven sentiment analysis of Norwegian news articles and company identification. The focus will be identifying needs, and exploring the different aspects of the data, such as the XML and HTML. Furthermore, the thesis presents a methodology of the system architecture, the data pre-processing, and the analysis. The main goal is to identify corporations and see what sentiment derives from the newspapers towards a corporation. The thesis will also present the effect of the sentiment and how it changes during time.

2 Introduction

Today there exist a lot of sites compared to what it used to when the internet first was invented in 1983(ARPANET). The purpose was to use it on military communication. The research development of the network became the internet. It wasn't until 1991 that the groundbreaking scientist Tim Berners-Lee invented the World Wide Web[1](WWW) as we know it today, and published the worlds first website. Today we have more than billions of documents and sites on the web. Each and everyone having a purpose and holding on to a lot of data and information.

Unfortunately, not all the data is useful, but we still have a lot to learn from extracting knowledge from the data. The data mining field allows a big amount of data, as well as processing for further knowledge extraction. With the help of tools, we find patterns that usually aren't visible for human eyes, for example, to find correlations between patient records for further diagnosis or to find insight in business corporations. Data mining methodology has shown that many instances can exploit the possibility of knowledge extraction, among others, process optimization, quality control, and human factors.

The field of data mining is a field that that is expanding. The media is one of the biggest expanding mediums to this day, which generates a large amount of data. Among others, reviews, opinions, and emotions, are the factors that are shared among people and different platforms. With the growing popularity of opinion-rich resources on the web, such as Facebook, Twitter, online review sites, and personal blogs, allows using information technologies to seek and extract opinions of others (Pang and Lee, 2006). Such benefits adds to the human knowledge and help make decisions based on new knowledge, but again the information is presented in differents.

[1] `http://info.cern.ch/hypertext/WWW/TheProject.html`

For example, Barack Obama expressed emotions towards a particular subject.

> *Another good story worth sharing: From one "kid from Akron" to a new generation of Akron kids, some remarkable early achievements at @IPROMISESchool. Great work, @KingJames-and even better work by those students. Proud to be a witness to their success.*[2]
>
> (Barack Obama - Twitter 12. April 2019)

Opinion mining and sentiment are some of the most modern ways of handling new big data from the web. The term *Opinion minding* and *sentiment analysis* did first appear in the early 2000s, and has later grown with a focus towards the media and how to extract opinions from it (Liu, 2012). There has been done a lot of research in the field of sentiment analysis, among others, to create a better sentiment lexicons and use automation for reviews. There are still many challenges to sentiment analysis, such as to use it through different application domains, to handle negations, irony and to use it across different languages in the field. Cross-domain specified analytic is the field where one explores the difference in domain-based sentiment, for example, the difference in exploring the tweets from Newspapers. Both the sentences and expressions can have similarities but are different when it comes to expressing emotions. The text in a tweet can have higher sentiment polarity compare to the sentiment in newspapers because of the different platforms, implying that domains reveal different ways of emotions towards a topic. This example shows a paragraph from the same News related where Obama tweeteded from in the example above, but the domain is different.

> *The academic results are early, and at 240, the sample size of students is small, but the inaugural classes of third and fourth graders at I Promise posted extraordinary results in their first set of district assessments. Ninety percent met or exceeded individual growth goals in reading and math, outpacing their peers across the district.* [3]
>
> (Erika L. Green - New York Times 12. April 2019)

[2] `https://twitter.com/BarackObama/status/1116840506506514433`

[3] `https://www.nytimes.com/2019/04/12/education/lebron-james-school-ohio.html`

Even though the News related paragraph doesn't express a high emotion of sentiment compared to the tweet, the human mind interprets the sentences as positive because of the context, penalties, and words within the text. It makes it difficult for a computer to understand the domain-specific meaning and that's why it's challenging to use the same methods across different domains. Fang et al. explains that the overall task of the sentiment analysis is described by the orientation of the text, which can be a problem in supervised learning. The methodology used a labeled data that was not available and suggested a method for a given domain rather than labeling data for different domains (Fang et al., 2012, p. 12).

2.1 Objective

Many corporations use machine learning to analyze a lot of data because it supports insight and knowledge extraction for corporations. Such insights can be a collection of Norwegian news text about a specific company, topic, or organizational information to support decision-making processes. For example, when pricing a large corporation or evaluating the corporation, some data has to be evaluated beforehand. In toady's society, we have a lot of information to process, and there exist different instances to gather that information from. For instance, the weather cast has a big amount of data that gets distributed, but also stores it as well for history purposes, and for evaluating previous observations to new once. The factor is primarily focused on media to gain knowledge about a corporation, and might consider as a time-consuming approach since the corporations research methods builds on manual research and orientation for gathering new knowledge. I want to use the media as a tool and by analyzing the News through the sentimental analysis and mentions I hope to short down the searching process and add knowledge that might be difficult to discover manually. I'm going to use sentiment analysis as a tool to see how companies and organizations are being perceived in the media, as well as to identify a company through mentions in the text.

The importance of developing the information system is to be able to stay agile according to the requirements, as they can change over time, but the idea and topic will still be the same. The research elaborates on previous research, but little has been used in Norwegian companies and specifically, Norwegian financial articles. Even tho it's not groundbreaking research, it

supports the idea of knowledge extraction through different tools of analysis, as well as a support for other processes, people, and corporations. The thesis presents previous methods in extracting, transforming, and displaying the data. Similarly, the thesis does not explore automation, but rather a single timeline of extraction, transformation, analysis, in stages. Lastly, the method is illustrated so it can be scaled based on the requirements.

Problem Description. It is needed to address the sub-tasks before starting the process. As mentioned earlier, the idea comes from support towards extracting knowledge from News related content, such as the media. A company that is interested in reading and retrieving knowledge use different resources to obtain information from the press. Among others, Twitter, Facebook, or newspapers. To see how the media has described a corporation apart from reading the yearly report and extracting information from customers, the numerical data is the only data that gives of the "facts" about a corporation. Numbers don't lie on this occasion and usually give a good representation of a corporation based on earnings, sales, turnovers, etc.
Numbers and numerical data are easier for computers to process and are, in most cases, easier to relate to when reviewing a corporation. Compare to reading, as it depends on the individual to establish a *meaning* towards a company. Humans do this occasionally and subjectively, but each human is different. Either in experience, personality, or emotionally. In some cases, the media can help establish the last bit of information needed for evaluating a corporation, but the interpretation varies in different ways. With the newspapers, we also need to understand what papers present and what are the most trends. In this case, which papers can present companies.

2.2 Research Questions

1. *RQ1:* How can Norwegian companies be identified based on their names?

2. *RQ2:* How are Norwegian companies represented by the online news media in Norway?

2.3 Contribution

The thesis aims to present one aspect of the analysis of online news articles and use already existing tools to support this. Previous studies, such as Öztürk and Ayvaz and Bollen et al., already reveal the media's power to contribute to knowledge discovery. However, such investigations are based on Twitter as a news source. This thesis relies on large scale dataset form Norwegian News articles, primarily hegnar.no (NO). By cross-validating the data and using existing tools to analyze possible findings.

Previous studies in the field use Twitter or other types of microblogging channels to retrieve data, with a small amount of News articles, specifically Norwegian news articles. For the analysis of the thesis, different feature analysis has been used in the art of knowledge retrieval. Furthermore, a combination of the features in the art of analysis and presentation. Using a combination of pre-processing and translation, a machine learning algorithm from NLTK has been used to identify sentiment in the text.

This approach is based on identifying names based on their first capital letter, and total mentions in text, rather than named entity recognition. Previous research show that named entity recognition is one preferred approach of identifying companies in text. Still, instead of training different labels of data, the thesis presents the number of times a company is mentioned. Furthermore, the approach narrows it down to the highest score, a combination of total mentions, and the total number of articles during the period of extraction. The approach does not consider the articles with the lowest polarity score for further analysis, as these papers can be presented as false-positive. A review of the data is used to evaluate the approach based on the findings, as well as comparing and reviewing some text with the highest and lowest scores. Figure 1 illustrates the process in more detail.

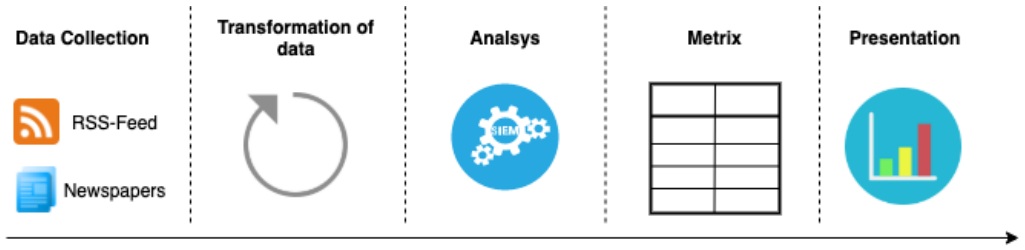

Figure 1: Schematic Illustration to knowledge extraction of Norwegian newspapers.

2.4 Thesis Outline

The thesis is separated into five main sections. The first section presents previous research done in the field of data mining and sentiment analysis. The second section presents the materials, tools, and technologies. The third section presents the methodology of the research. The fourth section presents the results and findings of the research, and the last sections present the summary, conclusion, and future work within the field.

3 Background

In the field of Nature Language Processing (NLP) and machine learning, there has been numerous accomplishment when it comes to transforming natural language into computer reading language and knowledge extraction. The explorations in the field have given root for various accomplishments, among others the power to analyze big data in the form of reviews, comments, news, and other micro-blogging areas. A significant amount of English corporations and organizations use machine learning for applying knowledge and real-time decision making, and has contributed to understand existing patterns. Among others, the power to plan for the future.

This review presents the background in three-part. The first section will present some known data mining methodologies used in the field of research, as well as the processes, the data extraction, and the knowledge. The second part will discuss the issue of applying NLP to new Norwegian corporations that do not not use machine learning in their process. It will address the challenges in the field of sentiment analysis. Later, a possible solution will be presented, where previous authors have addressed the challenges of natural language and dialect differences. I will also address the solution and into which extent they can work.

We live in an area where a lot of information is being distributed to the web and social media. According to the statistics from *ipsos* "Ipsos SoMetracker Q4'18" from January 2019 [4], 7 out of 10 women and 6 out of 10 men use Facebook on a daily basis. Only 400 000 uses Twitter on a daily basis. These numbers show that there a potential usage of such data when it comes to crawling data from services such as Twitter or newspapers.

3.1 The field

Most of the previous research done in the field of sentiment analysis has been done by analyzing text-based content. In later research, even emojis have been used to identify the sentiment of the text. For example, Felbo et al. used millions of texts with emojis from social media to pre-train models to learn emotional content in the text so it could be used to determine the sen-

[4] `https://www.ipsos.com/nb-no/ipsos-some-tracker-q418`

timent. The methods have given root for exploring public moods and have shown beneficial for corporations to use opinion minding methods.

It can give an insight into what customers might think about a product, brand, or customer support (Das et al., 2014), so the companies can make decisions accordingly and improve. It can also help to retrieve knowledge about their revenue and company, as well as to monitor other competitors, and can give an advantage when it comes to being first out in the market. Furthermore, Öztürk and Ayvaz demonstrated that language differentials across English and Turkish gave different results when comparing it to one another, and found that Turkish had a higher positive percentage compared to English (Öztürk and Ayvaz, 2018). For that reason, demonstrating that sentimental variation varies geographically and in topic-based importance.

3.2 Data Mining Models

As one of the parts in the process of knowledge extraction, the data mining methodologies are presented in both corporations, industrial and private field of data extraction and pre-processing data. Data comes from various sizes and shapes and are collected in multiple ways. The term data mining refers to extracting patterns and useful information from a lot of data (Rohanizadeh and Bameni, 2009). Fayyad et al. described this field of research as *"two high-level goals of data mining as a practice and as a description"*. There are various types of disciplines in how to extract and process the data. Each mining methodology can suit a corporation or organization in different ways, while others focus on data extraction and not on how to implement the methods in new systems. One of the most common data mining methodologies is CRISP-DM(Cross-Industry Standards Process for Data Mining, KDD (Knowledge Discovery in Databases), and SEEMA (Sample, Explore, Modify, Model, and Access). I will discuss the methodologies involving KDD, CRISP-DM, and a proposed data mining methodology from Rohanizadeh and Bameni. In data mining methodology, we can include several fields affecting the data mining process, where the area describes the importance of the process and what kind of results it give. The fields related to data mining are: data, sources, tasks and techniques, goals, tools, and procedures. Data

mining includes extraction, collecting, analyzing, and statistics. It's useful to find information in the data in a logical process. In big business corpora-

tions, data mining can be applied to safety, cost reduction, quality control, and others. They describe techniques that can be used in corporations. Authors propose a scheme of approach when it comes to knowledge discovery, and a process of data mining(Rogalewicz and Sika, 2016). The first stage, defined by the others, describes the preparation of the data, regarding the missing values, removing theoretical values. The second stage correlates to the results in step one but depends on the complexity. This stage is usually automated and depends on automated tools like RAM, CPU, and HDD storage. The third stage involves interpretation of the data, where Rohanizadeh and Bameni (2009) presents that it's essential to include experts in the analysis of data as it requires expert knowledge in statistical methods and Data mining methods. CRISP-DM has roots in the consortium of companies

founded by the European Commission(Rohanizadeh and Bameni, 2009). The methodology has a 6 step cycle where the phases are not rigid. It explores the business understanding, data understanding, and data preparation, modeling, evaluation, and deployment. Rohanizadeh and Bameni proposes a data mining methodology for technical procedures, such as to how better their process. The framework consists of several phases, among others, where an organization is a focus during the whole process. The model consists of five significant steps: analyze the organization, structure the work, develop data model, implement model, ongoing support. In the first stage, the organization has to know what data is needed for the project to succeed. The project has to be formulated, and one has to identify the goals and objectives of the project. Equally important, tools and techniques has to be selected and will be determined by the target. For example, if the goal is opinion mining, then the needs are to identify what tools are needed to collect the data.

Likewise, the tools er equally essential to identify the type of data we wish to process. Collecting data from a survey is different from collecting data from the web, as well as how to store the data. For example, locally, a local or external server or a database. The end needs justifies the tools needed for the process. The Knowledge Discovery in Databases Process (KDD) uses similar approaches when understanding the problem (Cios et al., 2007), but does not focus on the industrial process, rather about the data itself, and how to process it.

It has been used in several medical and software development areas, as it has been used for analyzing data or image-based classification. KDD relates

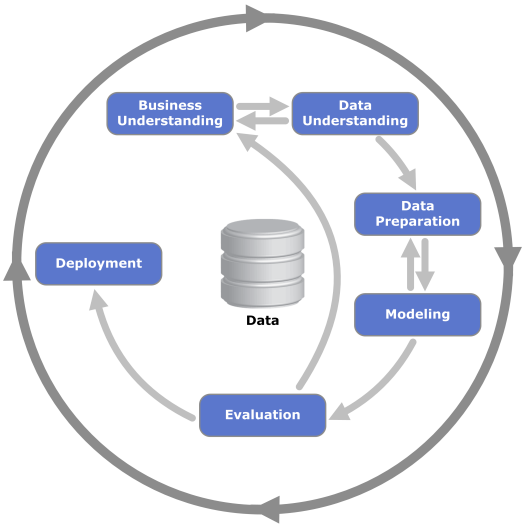

Figure 2: CRISP-DM Process.

to pattern recognition, databases, statistics, data visualization, and similarities when it comes to iterating over the process of data mining. CCRISP-DM, on the other hand, can be related to Industrial Processing, as it uses similar approaches. It focuses on understanding the business and mapping business objectives and business perspective(Azevedo and Santos, 2008). Rather than focusing directly on the data business requirements, KDD focuses on creating a dataset and extract what deemed to be knowledge.

Furthermore, Rohanizadeh and Bameni presents the data as a dynamic, and that it changes over time. One factor that is considered an essential aspect of the information is the cost, the owners, and regulation with the collection. For example, collecting online newspapers is fundamentally different from collecting data from a survey. Additionally, the data needs to be prepared, which involves running tests and removing noise from the data. Cios et al. describes this process as an input for data mining methods, which includes completeness of data, and removing noise and taking care of missing values. This stage is essential where the storage of unstructured data is processed to structured data or machine-readable data(Nirmal and Amalarethinam, 2015), similar to a data preparation or normalization of text in Sentiment Analysis. In CRISP-DM the data extraction has to be

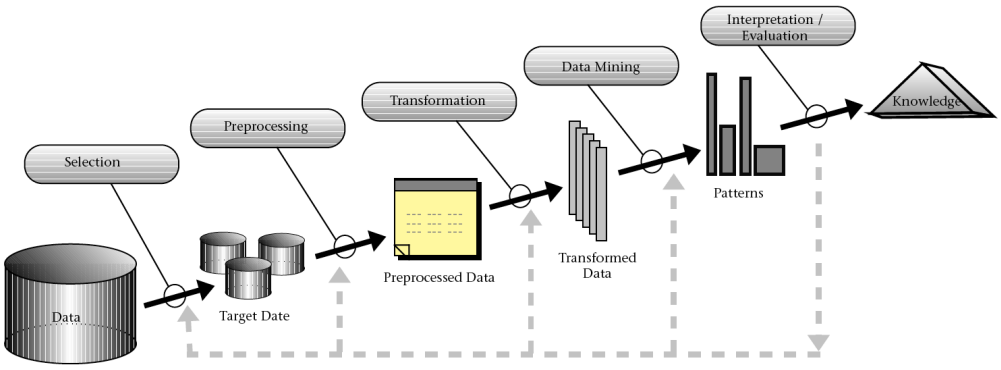

Figure 3: KDD Process.

monitored to see if it support requirements of the goal. It usually requires some iteration between the business, stakeholder, and having an insight into the data before preparing it. Rohanizadeh and Bameni explains that even though the data model can create insight into the data, in many cases, the data can consist of possible inconsistencies. The data preparation is meant to reduce inaccuracies in the collection, such as irregularities, eliminate duplicates, or correct missing values. In sentiment analysis, this would correlate the evaluation of the collection. For example, having too many duplicates would require removal, or the results would be misleading according to reality. Furthermore, Rohanizadeh and Bameni proposes the following, should there appear some missing values. The records are either ignored, new values are applied, or a particular category for this instance is added. Depending the what type data appears in the dataset one has to consider which solution is best suited for the best outcome. Ignoring the values can damage the result, but is a measure that can be accounted for as long as one is aware of it. Even though the data seems consistent, it has to be monitored for further

validation. Process control can be used for exploring the irregularities and define if additional actions need to be made before moving on to the next step. The process requires checking algorithms, hardware, or software. If the proposed model doesn't serve the purpose of the requirements or that previous measures were unaccounted for, then further actions are made on

behalf of this — for example, storage, code, wrong algorithms, or other instances. If the data is kept in its original form, then one has to consider this as something that can manifest further on in the process. Nirmal and Amalarethinam explains that most of the algorithms today support the data mining structure and unstructured data, as well as how to re-process the data, as its advantages can be found in many different mining methodologies. For example, the semi-structured nature of the RSS content has a lot of information to it, which can be beneficial to present for the end-user(O'Shea and Levene, 2011). By pre-processing details, one can prepare for analytical reviews, discover trends, or mining numerical data, but understanding the use of the data has to be at mind at all times.

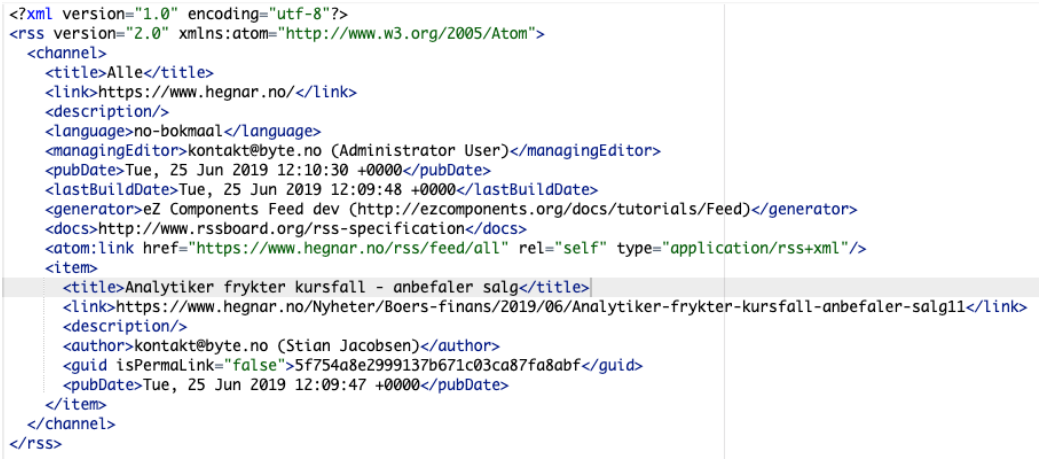

Figure 4: RSS Feed from hegnar.no.

As identified by Nirmal and Amalarethinam, the unstructured data has to be understood before moving on to the next step. The selection of attributes, values, and information is essential. The data is then selected according to goals and features represented in the first stages. Thereby, remove unnecessary value noise. For example, if we intend to use RSS as our data source, we would first review our data source and see what kind of values are needed for further transformation. Either all of the RSS-feed is extracted or parts of it. Wanner et al. used DataMiner sub-system of myDS to extract the data from RSS feeds. The extraction consisted of two separated mining techniques: the Occurrence Mining, which extracted values according to certain strings, and the Value Mining, which records values to a certain topic or movement, such

as sports, politician or financial. After deploying the model, quality control

needs to be made to validate the model. Rohanizadeh and Bameni presents the model validation to determine whether the model correctly predicts the behavior of the variables. To solve the thresholds, the model can be assigned to each project for validation. Similarly, CRISP-DM methodology creates a model for later evaluation. Rogalewicz and Sika defines the results as a result of the companies translation into real conditions that describes the companies functionality. In comparison, KDD uses an interpretation and evaluation of the mined patterns.

3.3 Issues in the Field of Sentimental Analysis

Sentiment Analysis, also known as opinion mining, refers to the use of natural language processing (NLP) and text analysis to extract and identify personal information, as well as sentiment in the text. For example, "I like running", is a positive loaded sentence, while "I dislike running" is a negative loaded sentence. In both cases, the subject *running* is loaded with either negative or positive words. It's easy for humans to understand the meaning and the written words or the sentiment written in them. The machine needs an interpreter, or else it wouldn't understand "this is a cat" or "I love you" is a romantic sentence. The problem with sentiment analysis is that the machine needs to be taught how to understand natural language.

For the machine to understand, the text needs a set of interpretation rules. The interpretation means that we consider the program to return a piece of given information after applying the following paragraph. In this case, the sentiment of the text. One of the tasks in sentiment analysis is classifying the polarity of the given text, whether the sentence or entity feature is either positive, negative, or neutral loaded. This is a fundamental task in sentiment analysis, where the negative and positive words are compared to the words in a sentence. For each positive loaded word, the weight of the sentence increases, and for each negative, the weight of the sentence decreases. The total sentiment of the text is positive if the total amount is positive, and vise Versa. Even though the task is essential, it is also an inadequate approach to extracting sentiment or meaning from the text. A text can be positively loaded without including the most apparent sentiment words or sentences. In relation, one of the problems in sentiment analysis is dealing with sarcasm

within the text or the negations in the text. For example, "wouldn't" or "isn't" are examples of negated words. There are different ways of avoiding sarcasm and handling negation, though getting a 100% validation is challenging to achieve, the goal is to aim as close to a 100%.

There are different ways of performing sentiment analysis on text (Thelwall and Prabowo, 2009). One of the popular ways is to generate a list of predetermining sentiment, also known as a classification in rule-based. The purpose is the reviewed text upon the classification, which is close to equal positive and negative sentiment. Depending on the values it's possible to see if the word is negative loaded or positive loaded. The overall sentiment polarity is based on an average of all the scores, also represented as polarity score.

As to extracting knowledge from data, one of the problems derives from exploring different domain in sentiment analysis (Raina, 2013). Previous research has shown the difficulty in creating a mix domain analysis for text and has shown that it changes the sentiment. This makes sense since replacing the domain makes it challenging to identify fine-grained sentiment in the text compared to other domain specified fields. As defined by Raina, analyzing text-based newspapers or articles can be challenging because of the difference in which emotions and opinions are expressed. News related content is different from public mood as it relates to actual news, facts, and not direct mood and attitudes. However, despite not having substantial fluctuations in sentiment and neutral appearance, news-related content can still have polarity. It does not exclude the content from being "happy" or "sad". Soelistio and Surendra established a system flow detecting and re-processing the data before testing. The proposition focuses on five modules: reader and parser, cleanser, helper, analyzer, and display.

Unlike news, other media channels such as Twitter or Google+ has shown to be a good domain to explore public mood, as the content usually is either emotion or opinion-based. As stated by Bollen et al.(Bollen et al., 2011a, page 450): *"tweets normally tend to fall in one of two different content camps: users that microblog about themselves and those that use microblogging to share information"*(Naaman et al., 2010). The tweets tend to convey information and often mood states of the user, such as "I'm glad", "I feel sad about it", or "I'm not excited". In these cases, moods are explicitly

being used to refer to the topic emotionally. This makes the domain suited for sentiment analysis as it supports the definition by Oxford dictionaries "a view of opinion, a feeling of emotion, or the expression of view". Liu defines the problem as follows, *"sentiment analysis mainly studies opinions which express or imply positive, negative or neutral sentiment."*(Liu, 2012, page 17). In practice making it easier when categorizing the content into positive, negative, and neutral compared to newspapers.

In comparison, Amolik et al. concludes that Twitter domain can be challenging to identify emotional words due to the presence of repeating characters, slang, white space, or misspelling (Amolik et al., 2016). Similarly, Kiritchenko et al. supports this statement adding, that short text messages compose a challenge in the sentiment because of the limitations in length and words. Sarcasm and misspelling are one of the more common issues in the field and describes the word in a similar way, but gives another meaning to the context. For instance, if a person wrote, "I just lost my job, that's just perfect!", or "OMG! The football match was gr8!". A human would know that losing their job isn't positive even though the sentence "that's just perfect" is written afterward, a computer wouldn't understand the sarcasm. The words "OMG" and "gr8" is not introduced as negative or positive loaded word, because it is not presented in natural language, and will not be interpreted because of misspelling. It's preferably introduced as an abbreviation, elongation, or misspellings.

3.4 Misspelling

Amolik et al. demonstrated a solution by using a feature vector in two phasesAmolik et al. (2016). First, the extraction of feature twitter specific words are removed from the text. Then the extracted feature vector is transformed into text. Second, features are extracted from the standard text without slang or hashtags. These features will then form the feature vector for later classification. Similarly, Kiritchenko et al. used a general-purpose sentimental lexicon for detecting misspelling. The implementation covered tweet-specific sentiment from 2.5 million tweets and marked each word to correlate to either abbreviations, elongations, or misspellings. Words that usually don't express emotion where also included in the lexicon. This was done to estimate the polarity of the sentiment. In the research where Kiritchenko et al. wanted to estimate the reaction of Adverse Drug Reaction(ADR), presented

that by retrieving the tweets and DailyStenght post for manually annotating the ADR improved the ADR identifications in Tweets.

3.5 Sarcasm

People tend to express themselves by using facial expressions, certain tone gestures, or hand movements to indicate sarcasm to simplify meaning (Jena et al., 2016). Gestures are usually missing in the raw text, making it difficult for the machine to detect certain sarcasm based expressions. Jena et al. used a different approach for detecting irony in text and proposed a Hadoop based framework. By passing the text through the MapReduce function, the tweets are classified as either positive, negative, or neutral. In general, the tweet is used as an input to the sentiment. They are presented as an output with either real positive, sarcastic, or negative or sarcastic(Jena et al., 2016). A similar approach was presented by Bharti et al., but focused on the tweets that started with interjection word. The identified text that abrupt in sentences of the total of 50000 tweets with hashtag #sarcasm (Bharti et al., 2015).

Felbo et al. used a different approach for detecting sarcasm in text. They used sarcasm from two different datasets from Internet Argument Corpus (Walker et al., 2012). They used a modeling-base features, among others, trigrams and unigrams with Support Vector Machine(SVM), and later using GoogleNews word2vec for embedding-based features. For cross-validation, a hyper parameter search for regulation was used. In light of detecting sarcasm in sentences, the solution is based on two factors. The first represents dataset and domain explicitly. For example, a pattern-based approach might seem suited if an external corpus is not used (Davidov et al., 2010). On the contrary, using an estimator for the word frequency worked well for the training set. Davidov et al. found that the pattern-based framework was sufficient for detecting sarcasm in tweets since each feature vector was based on tweet. They allowed to approximate the pattern in the matching for learning. Additionally, the number of generic features where focused on as well, such as "!", "?" and quotes.

Additionally, Korkontzelos et al. presents the style of which the public people usually tend to write in the media to indicate strong or soft emotions towards a topic. The style not common in the News or newspapers, but

rather in other types of news channels, such as Twitter or Facebook. It's not necessary towards capturing sarcasm, but rather words that are written with a capitalized letter. They usually tend to indicate a strong feeling of emotions towards a topic, for example, anger, joy, sadness, etc. (Korkontzelos et al., 2016). Furthermore, elongated words (e.g. "riiight", or "yeess"), and punctuation marks are usually used to describe emotions. To identify the sentiment in sentences Korkontzelos et al. used capitalization, elongated words, and punctuation's as the source of information.

3.6 Negation

The negative loaded word can be represented by the negation of the word. One assumption is that the polarity will be determined by the words(e.g. "not", "no", and "less") in one sentence. If it appears one or more times, the polarity will be changed as many times as the negation appears in the sentence(Soelistio and Surendra, 2015). Soelistio and Surendra proposed a model for detecting negations in sentences. The research found that articles and news towards politician gave good results after testing, but had yet to be tested towards more significant and more complex data set, such as slang. In addition, the model had difficulties in identifying "who" and "whom" in articles.

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)}$$

Figure 5: Naïve Bayes.

Another model that is used a lot in sentiment analysis the *Naïve Bayes* model for simple probability classifier models, based on bayes rule and independent assumptions. Each class is classified as positive, negative, or neutral. The problem occurs if the classifier encounters words that have not been trained, resulting in that the classes would become zero (Swati et al., 2015). Based on the ideas of Swati et al., the problem can be solved by *laplacian smoothing*. Aside from handling negations, the Bernoulli Naïve Bayes can be used to handling duplication's as well, but handling negated words was one of the main factors that Swati et al. preferred to use this classifier (Swati et al., 2015). Information is generally conveyed by one or

more adjectives or a combination of them in sentences, where the features can be captured by the pair of words, called bigrams or triplets of words, called trigrams (Swati et al., 2015, page 116). The easiest way of addressing negations is by changing their values from s to -s (Alistair and Diana, 2005). For example, "this food is good" to "this food is ***not*** good", but the previous study has shown that this solution not sufficient for handling negations.

Kiritchenko et al. presented through experiment that positive terms tend to reverse their polarity when they were negated, while negative words only were changed to negative. By building two lexicons, one for negated context and one for affirmative (non-negated) context (Kiritchenko et al., 2014, page 724). By following the work from Pang et al., the negated words were handled according to punctuation marks (Kiritchenko et al., 2014, page 733), and words where created from Christopher Potts tutorial[5]. The result from the negated content were added to the Negated Context Corpus, and the rest is added to the Affirmative Context Corpus. Similar to the research of Alistair and Diana(Alistair and Diana, 2005), Pang et al. used the same approach of handling negated values, by adding a `_NOT` tag between a negation word and the first punctuation followed by the negation word. Previous experiments have shown that removing the negation tag has a small effect but a harmful impact on the performance.

Korkontzelos et al. used a similar approach for capturing negated phrases, by using Christopher Potts sentiment tutorial (Korkontzelos et al., 2016, page 151). The process focuses on a sequence where a word starts the negation and ends with a punctuation. Because of the variation of typing errors in the text, the increase of variation and process in machine learning tools leads to sparsity. To reduce sparsity Korkontzelos et al. used *Twitter Word Cluster*(Korkontzelos et al., 2016, page 151). The cluster included a set of 1000 clusters of similarly spelled words. Subsequently, creating a tagger by applying *the Brown Clustering Algorithm* on 56 million English tweets. This resulted in a cluster containing 216,856 different words. Thus, making it possible to map the amount of correctly spelled and misspelled words to smaller clusters and making links between frequently and less used words.

[5]`http://sentiment.christopherpotts.net/lingstruc.html`

3.7 Algorithms and Classifications

Previous work has also shown that there has to be a variable that can confirm the analysis when testing and validating the results comparing it to different hypothesis. For example, Andranik Tumasjan proposed sentimental analysis on tweets for predicting the German national election results (Andranik Tumasjan, 2005). By examining 100,000 political tweet messages between August 13th, 2009, and September 19th, 2009. The collection consisted of the six parties in the German parliament, were 70.000 tweets mentioned one of the major parties and 35.000 tweets referring to the politicians. For extracting the sentiment, they used a LIWC2007(Linguistic Inquiry and Word Count), which is a tool for assessing components of text samples. The research showed that the platform was indeed used to express sentiment towards the political election. The results showed the tweets came close to the election poll, but had an average prediction error of 1.65% (Andranik Tumasjan, 2005, p. 183). Similarly, Öztürk and Ayvaz used a time-frame from March 29th to April 30th, 2019, when collecting tweets for the research.

Another similar approach was done by Bollen et al., where the primary purpose was to predict the stock market based on tweets from the public. The method is identical when it came to narrow the time-frame startpoint to endpoint. In addition, the research carried on from February 28 to December 19, 2008 (Bollen et al., 2011b, page 2). The advantages by using a bigger time-frame is that the collection is bigger, meaning that the data is consistent and can be more precise when being analyzed afterwards, -more importantly, used to predict the probability of changes in the stock marked within the same time-frame based on data from the stock marked. The collection consisted of a total of 9.853.498 tweets, where the tweets provided an identifier, date of submission, type, and text of the tweet. The tweets were later processed through two mood assessment tools: *OpinionFinder*[6], which is public software for sentiment analysis that can determine the sentiment in a text(positive, negative or neutral), and *GPOMS*, which measures six different sides of mood in text. Each text was measured on a given day giving, as well as extracting, DJIA values from Yahoo! Finance. In total, seven measurements were used to determine the sentiment of the text. By extracting the financial values and measure them up against sentiment results. The results where later used to pinpoint if it is possible to determine

[6] `http://www.cs.pitt.edu/mpqa/opinionfinderrelease/`

the outcome over time.

In both scenarios described by Andranik Tumasjan and Bollen et al., a time-frame was set in which the data was collected. The time-frame correlates to the amount of data collection within that time-frame but has different uses of what to compare that data to. Rohanizadeh and Bameni already presented the challenges of changes in the data, and that it tend to change over time a period of time. However, this might be true, it's not an issue as long as we are aware of them when and can account for changes in the process.

There are many types of classification models for processing data. Among others, *Naïve Bayes* classification model has been used in different research when classifying the data. Several other types of research have used different classification models on the same dataset to verify which model works best on the dataset.

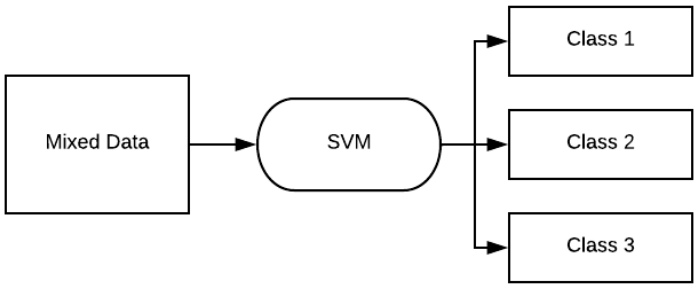

Figure 6: Simplified process of SVM Classifier.

The research conducted by Alomari et al. *Naïve Bayes* classification model was applied to the dataset. Similarly, the SVM was applied to the dataset as well. The results conclude that the SVM classifier using a stemmer with weighing schemes through N-grams, such as Unigram, Bigram, and Trigrams, gave the best results. Only the Bigram showed promising results with an accuracy of 88.72% (Alomari et al., 2017, p. 608). Likewise, Kim and Hovy presents the usage of the SVM classifier, but the purpose was mainly to binary classify (eg. WIN or LOSE) for the final classification later in the process. The assumption was that generalized patterns would present a better relationship between the parties. They generalized the patterns of

words so it would be possible to predict opinion and instead skipped the different trigrams, such as *"Liberals will win*, *"NDP will win"*, and *Conservative will win*, they generalized the data to *PARTY will win*. In contrast to Alomari et al. the experiments investigate the aspect of predictive opinion and comparing them to judgment opinion.

3.8 Preparation and Normalization

3.9 Natural Language Processing (NLP)

NLP is the process of manipulating the text. By natural language, we consider the communication language between humans, such as English, Spanish, Korean, etc. In contrast to artificial intelligence, human language can be difficult to define unless we have a set of rules. Previous research and explorations withing the process of natural language have shown that it's process is important within computer science and artificial intelligence. Even though the process has been in the field for years, we can find work of NLP back to the 40s and 50s. Alan Turing proposed an evaluation of artificial intelligence, which today is called the *Turing Test*(Turing, 2009).

Development of the NLP applications is a challenge since computer work with precise and structured languages, such as programming languages. However, human words are often ambiguous, and the structure depends on many variables, such as slang, regional dialects, social context, domain. With the ongoing growth of the world wide web, there is a drastic increase in data and information. As the data increases, the amount of unstructured data does as well. Nirmal and Amalarethinam says the following: *Due to the enormous volume, the highly unstructured nature of the data and the vast rate at which it is being generated, parallel implementations of pre-processing algorithms becomes important.*(Nirmal and Amalarethinam, 2015, p. 149). Before we can retrieve knowledge from the data, the unstructured data has to be processed in a way that is suited for the purpose. Previous research has shown that there are different ways of re-processing the data, and one of the most commong tasks within natural language processes are: tokenizing, stemming, lemmatizing, etc.

3.9.1 Tokenize and Part-of-Speech (POS)

The NLP field is a subsection in computer science and artificial intelligence. One of the reason we wish to transform the data is that it's easier to analyze the text when it's normalized. The purpose of the analysis determines the way of how the text is normalized. One of the more popular tools for preparing and normalize the data is *Natural Language Tool Kit*[7] (NLTK). It's a platform developed by Steven Bird and Edward Loper. It that was build in the Department of Computer and Information Science at the University of Pennsylvania. NLTK supports the research and teaching in NLP and its close areas, such as information extraction, machine learning, cognitive science, and artificial intelligence. As of 2019, the tool kit supports several languages but is a strong supporter of the English language when it comes to analysis, as it's already integrated into the software. Bharti et al. used NLTK to discover *Part-of-Speech* (POS) tags in the text and present them in in a tree (see figure 7). POS tags are also known as classing the words or lexical categorizing them. For the POS tags to be recognized. Also, the text needs to be tokenized, meaning that each word in a sentence needs to be separated into its own instance. Next, the POS tag method in the NLTK module will classify the words into tags of verbs, adjectives, conjunction, and so on.

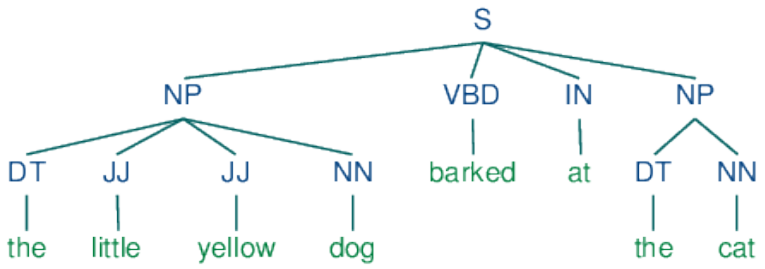

Figure 7: Part-of-speech (POS) Taggs.
NP = Proper Noun, DT = Determiner, JJ = Adjective, NN = Noun, VBD = Verb, Past Tense, IN = Proposition

[7] `https://www.nltk.org/`

3.9.2 Stemming and Lemmatization

Nasukawa T describes their method as a shallow method for NLP, but they focused more on stemming and the analysis of the POS tags. Stemming is the process of producing morphological variants of a word. By reducing the inflected words to their stem word, base, or root form. Figure 8 illustrate the outcome of stemming the part of speech tags. The process only operates with single words without the knowledge of the context (Fang et al., 2012). There are several other tools that support the morphological variant of words and transform them to their root form. We've already mentioned NLKT as a reliable tool for re-processing the text. In addition, so does *Spacy*[8], which is a platform that supports several programming languages. Among other Python, Java, and C++.

In correlation to Nasukawa T, Fong et al. used stemming for feature extraction and later implemented a sentiment analysis with MALLET (Fong et al., 2013, p. 302). As a result of this, they removed capitalization, punctuation, and stop words from the text. According to Fong et al., this method makes it easier when training the data set. Likewise, Lau et al. applied a similar approach for text pre-processing (Lau et al., 2009). The main purpose of the research was to do a sentiment analysis after extracting the TF-IDF (Frequency-inverse Document Frequency) weighing the most. Compare to TF-IDF, TF (Term Frequency) doesn't care if a word is common or not. Swati et al. presents the WEKA tool for normalizing the newspaper's data. Where the main goals was to retrieve sentences, tokenize, remove punctuations, and detect POS tags (Swati et al., 2015, p. 115). In addition, Alomari et al. presents the different aspects of using stemming in the data set. Among others, applied different stemming techniques, such as no stemmer, stemmer, and light stemmer.

Alomari et al. used RapidMiner software to provide text tokenization, filtration, and Arabic Stemming in their research for sentiment analysis. After pre-processing the text, each document was analyzed by applying TF-IDF or TF weighting schemes and generated eighteen different analysis process for each weighting scheme per classifier(Alomari et al., 2017, p. 605). In addition to using RapidMiner, the *ASAP Utilities*[9] was used to clean repeated characters, filter text and remove links from the text. The research found

[8] `https://spacy.io/`

[9] `https://www.asap-utilities.com/`

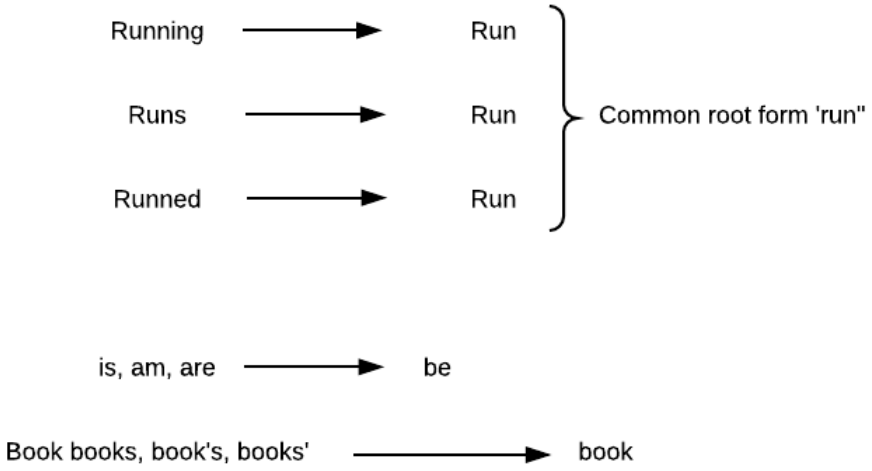

Figure 8: Stemming and Lemmatization.

that, by applying a new public Arabic tweet corpus, which consisted of 1800 annotated tweets written in the Jordanian dialect, gave promising results when analyzing the data. There existed little ML supervised approach for Arabic sentiment, so a new corpus contributes to the Arabic sentiment analysis in domain limited resources.

In comparison to Lau et al., the pre-processing process showed that the data normalization is based on the domain-specific orientation of the data. For example, tweets present a key factor and importance to establish an overview of the collected data. Similarly, Rohanizadeh and Bameni emphasizes the need for understanding what is needed for the analysis. In this case, the understanding of how natural language in microblogging is build up. Likewise, Swati et al. presents another domain of extracting data, such as news, which is a different domain and also a different approach to extracting and transforming the data. Swati et al. presents crawling data extraction is divided into different steps. Since the articles are presented in HTML, a second step needs to be applied to remove the text from the HTML(Swati et al., 2015, p. 115). This is in many ways another step of the re-processing of the data,

given that the extraction also included some HTML tag, like %¡h1%¿ or ¡p¿. The removal of these tags is in itself a process of unwanted removing noise.

Lemmatizing in NLP refers to the morphological analysis of the words. To be able to analyze each word, there needs to be a detailed dictionary, so the algorithm can be able to look through and link the words of the form back to the lemma word(s). In comparison to stemming, a lemma is the base form of all its inflectional forms, meaning that all the list of lemmas are dictionaries. In NLTK, we get a connection to *WordNet*[10], which is a database of English. The database provides a conceptual relationship between words such as hyponyms, synonyms, antonyms, etc.

Lemmatizing the text has proven to be a practical resource when transforming the data and providing some meaning to it. Previous research presents the effectiveness in reducing the inflected forms to the base form, and a reduction in the sparseness of the data (Joachims, 1998). Fang et al. used the *Standfor Core Natural Language Process toolkit*[11] to pre-process the labeled data from different source domains, and the test data from the target domain (Fang et al., 2012, p. 7). The results showed that by lemmatizing the documents, they were able to determine the contextual POS context within the text, meaning that they were able to specify a more appropriate classification model for their purpose.

Nielsen used the wordlist towards ANEW (Affective Norms of English Words) and found that the scoring without normalization was the highest. The results showed that there where little to no difference in the tweets sentiment with the given range of 0.522-0.526 (Nielsen, 2011). By looking at the intersection between the two word lists, which a consisted of a direct 299 words, they could evaluate the performance between the application of the ANEW.

In light of the pre-processing procedure, different approaches can be utilized for different purposes based on the needs of the analysis. A negative side of it is that it can damage the results to some extent. For example, if we remove capitalized letters from the text in some cases, the analysis of name

[10] `https://wordnet.princeton.edu/`

[11] `https://stanfordnlp.github.io/stanfordnlp/`

recognition would fail, given that the test was based on capitalized letters resulting in different names. For that reason removing capitalized letters from the text may include a side effect of the natural language process. On the other hand, it's not common to only use one NLP technique to process the data or extract meaning from it. For example, if a research was plainly based on stemming and then analyze, the results would have been differently if we consider the different factors support the analysis at the end. Another example could be to remove noise and then analyze, given it's domain-specific, then we haven't accounted for other factors such as stemming or other preprocessing techniques. Previous research shows that different methods are acquired to pre-process the data. These factors contribute to better end results.

3.10 Translation

Analyzing the translated text is in its concept, an easy task. The process involves only to translate the text to be classified by the machine (e.g. googletrans4.1, google translate), and then pre-processing the text or running sentiment analysis on the translated text. Different methods have been applied for translating the words and sentences to other languages. In some cases, the translation can make for a different sentence or word, which means that the strength of the word or sentence can have a totally different sentiment from the original language. One of the standard processes in sentiment analysis is to use machine translation to translate the text into another language. Hammer et al. used this method to translate the English sentiment AFINN lexicon to Norwegian by using Google translate(Hammer et al., 2014). They created a second lexicon and double-checked for errors manually. 13.296 product reviewed from the Norwegian online shopping site komplett.no where used to validate the quality of the sentiment lexicon. Also, 4149 movie reviews from filmweb.no were used as well. The results show that the state-of-the-art graph methods do not outperform the translated AFINN-list, meaning that machine translation of linguistic resources in English can be used for the Norwegian Language. They also discovered challenges in the translation of the text. Among others, the strong sentiment words were challenging to translate. For example, fuck, motherfucker, son of a bitch, which AFINN suffers from.

Limitation. The process of success is plainly depending on the text that is being translated. If we assume that the text is structured in a professional way, we can assume that the translation of the text will be as close to a success. A few examples of the this can be text from serious newspapers, published books or previous research. Other media channels, such as Twitter, Facebook, can have high amount of slang, smileys, spelling errors, inconsistent punctuation's that can be influence the translation. These factors will influence on how the text is being translated, and leave parts that comprehensible to the translator untranslated. In other words, the chance of a good analysis decreases depending on the translation. We know that the text needs to be consistent for the sentiment analysis to be a success, if we lose words along the way, this can harm the results. Considering the limitations of text translation is regarding the amount of text that needs to be translated. This adds another step to the process making it a reliable part of the process. For smaller set of text the translation process is fast, and slow for bigger set of text.

Advantages. On the other side, the method of translating the text makes it possible for the program to remain unchanged as soon as the language is discovered. When the language is discovered, it's a simple task to translate the text(s) into the program default language (English), normalizing the text and then running an analysis. Furthermore, the method of translating the text prevents a lot of implementation in different languages. Instead of creating different approaches and algorithms for different languages, one can utilize one approach for one language only. Thereby using the same approach for all the other languages. The advantages of using this method is minimization of tools. For example, no need to create a dictionary each time. On the other hand, a new domain-based dictionary has to be implemented either way.

3.11 Summary & Differences to Previous Research

We see that there different aspects in both the data mining field and different solutions in pre-processing and analyzing the data. The biggest trends in the field are usually pointed towards tweets, online comments or other microblogging communities that presents a quantity of text. Each field of researches correlates to bigger proportions of emotions, such as in the media. Also, little research is done within the newspaper domain. A combination of

mix-domain-specific research is usually done and often concludes a separation approach to a more specific domain approach. Another key aspect of the research is the data mining methodology. It has been consistent over time and has shown that there are room for readjustments depending on the purpose of the process. For example, by having an agile approach of mining and then analyzing, gives more room for improvement if the approach is not optimized.

Previous research has also shown that there exist different libraries for pre-processing and different tools for analyzing the data. This implies that there will not be much need of reinventing the wheel when it comes to algorithms within programming, but rather using the key aspect of data mining methodology and use simple, powerful pre-processing tools for research. Different approaches have been done to analyze and optimize the analysis. However, it's still little known on domain specific analysis, such as Norwegian Financial Newspapers, and how companies are being perceived. Little research focus on the Norwegian language to be processed, and combine a translation tool on Newspapers. The thesis aims to use some the aspects from data mining, translation, re-processing and analysis to try to solve the task of acquiring knowledge and identify companies in newspapers.

4 Materials

4.1 Tools

The material section present the tools that where used during the research. The section describes each tool and technology in simple detail, and elaborates its functionalities. A good quantity of existing tools where used for the thesis. Most of all, Python was used on a large scale for system developing and analysis. Numbers from Apple, was used to iterate over the data and present it in a good graphical way.

Pycharm[12] is an integrated development environment (IDE), specifically for Python language. The environment is developed by the Czech company JetBrains, and provides different features, such as code analysis, graphical representation, and support web development, as well as Data Science.

Numbers is an application developed by Apple Inc and helps build beautiful spreadsheets for the user. The application uses a free form "canvas" approach the transforms the table to one or more tables on one single page. It presents the data through the chart and graphical text. Like a tradition spreadsheet, such as Excel, that uses the spreadsheet as one single container.

Python is a programming language generated for high-level general purposes. The program has a wide variety of modules and applications that can support the programmer. It can further be extended to C or C++ for better computing speed on bigger tasks and provides a strong structured system that enables a logical application to process small or large tasks.

Python is used as the programming language due to the focus on simple yet powerful syntax, as well as a variety of libraries(Python Software Foundation).

[12] `https://www.jetbrains.com/pycharm/`

Virtual Machine (VM) is an emulation of a computer system. VM is based on computer architectures that give functional and physical computer environment in computing. The emulated system allows for the execution of software application and operating systems for another CPU or architecture. The VM's usually have a terminal that allows for navigation within the system and, depending on the operating system, will have the same shell navigating system of either a Linux, Macintosh, or Windows operating system.

Crontab[13] is a time-based utility software in Unix operative systems. It lets the user set up and maintain software environments to schedule jobs. Users can set up and maintain schedule jobs so that the software runs at fixed times, dates, and intervals. The process is usually referred to as automated operations, though its general-purpose can make it useful for downloading files or downloading email at intervals.

Comma-Separated Values (CSV) is a delimited text file that uses comma to separate values. The CSV store data in plain text using character set, such as ASCII and various UNICODE (UTF-8) character sets. Each line of the data is a data record that consists of one or more fields, separated by commas.

Key Modules are the modules that are used in Python. The modules were used in different stages of the process and contributed to either extract, process, or review the data.

BeautifulSoup[14] (BS) is a powerful python library that helps pull data from HTML and XML files. This includes having malformed markup, for example, non-closed tags. It works for parsing the data and creates a parse tree for parsed pages that can be used to extract data from an HTML file. This is usually referred to as web scraping.

Element Tree[15] (ET) is a Python library which provides an element type, that enables flexibility within the object. The design is made to store hier-

[13] `https://www.adminschoice.com/crontab-quick-reference`
[14] `https://www.crummy.com/software/BeautifulSoup/bs4/doc/`
[15] `https://docs.python.org/3.6/library/xml.etree.elementtree.html`

archical data structures in simplified XML. Element Tree also provides for reading and write functionality with the element structures in XML.

Pandas[16] is an open-source Python module that provides high-performance and easy use data structures and data analysis tool (three-cluase BSD license 1999). The module provides data structures and operations for manipulating numerical tables and time series. The name Panda comes from the term "panel data", a condition that describes a dataset that includes observation over different time periods for the same individual. The library has several features, among the most popular once are:

- Fast and reliable.
- Auto language detection
- Customized service
- Connection pooling

Python googleTrans is a free an unlimited python library that is implemented in the Google Translate API. The module uses the Google Translate Ajax API[17] to make calls to methods such as to detect the language and translate it. The module gives a variety of advantages in translation because of its features.

- DataFrame object for data manipulation with integrated index.
- Tools for reading and writing in-memory structures.
- Data alignment and handling missing data.
- Data filtration.
- Time series functionality: Date range, statistics, linear regression, date shifting and lagging.
- Data structure column insertion and deletion.
- Data set merging and joining.

[16] `https://pandas.pydata.org/pandas-docs/stable`
[17] `http://info.cern.ch/hypertext/WWW/TheProject.html`

Natural Language Tool Kit (NLTK) is a powerful Python library in Python programming language. The tool kit includes graphical demonstrations and different sample data. The NLTK is intended to support the different arts within NLP or closely related areas such as artificial intelligence, machine learning, and information retrieval.

It provides easy to use interfaces and has over 50 different corpora and lexical resources the data file. Among others, WordNet and sentiment and tools that helps for pre-processing the data, such as classification tools, tokenizers, stemming, tagging, and parsing.

5 Methods

The development of the system is flexible in that way that it can be scaled to other proportions depending on the requirement of the system. The methodology is divided into three parts. The first part presents the selection of data, and it's limitations. The second part describes the requirement of the system, and two main aspects of the methodology, newspapers, and RSS feeds that are divided into paragraphs. The last part presents the development, the selection of companies, and the iteration in detail.

5.1 Selection of Data and it's Limitations

In the early stages of the research, the main goal was to analyze Norwegian Tweets. The main goal was based on extracting different data from where a corporation might have been mentioned. The sole problem while reviewing the data is the amount of collections that would have been presented. Therefore this stands out as a limitation to the methodology. Likewise, the amount of potential organizations in the data does present as a limitation to the methodology as well. As mentioned by Rohanizadeh and Bameni, the first part of extracting knowledge from the data is to understand what the data consists of and what part of the data needs to be extracted. Furthermore, the methodology proposes an understanding of the data, which leads to the preparation of the data.

The target data can, in some cases, be a lot bigger for bigger corporations, so an understanding of the hardware limitation, such as HDD or RAM, is important and can be a potential limitation to the process. The scale of the data and pre-processing will determine what hardware technologies are suited for the process. For the methodology this is not an issue as we're not going to extract big quantities of data.

Requirements. Requirements are identified as intended products that act as a specification on how the system architecture should behave or how it should perform (Preece et al., 2015). The requirements should be clear to understand, in such a way that it isn't possible to misinterpretation on any part involved in the process. Establishing the requirements for the architecture will lay the core foundation of what it is to expect from the system. Preece et al. presents two types of requirements that have proved to be

traditional in system development. The functional requirement, that detail the specifics on what the system should do, and non-functional requirements that describe the restriction for the system.

The methodology of the system used for this project was early established, but it has to be point out that a few parts of the development was changed due to restrictions and limitation of the data. This is later presented under *Selection of data and limitation*. While the requirements for the system architecture were established in the early stages, they were checked later to reassure it's consistent of the system.

Functional Requirements where set based on another Norwegian company. The requirements is basically an approach of the architecture, most important the understanding of the data, its collection attributes, and its re-processing procedure.

Newspapers are described as a paper publication that is issued regularly, usually once a day or once a week. The papers are meant for public information about current news or events. People like to stay updated and informed about their local city, state, or country. Newspapers tend to have different topics to them, for example, crime, business, or sports. Many newspapers include daily or weekly weather news. The publications are usually sectioned based on the subject and the content of the paper. The most important news will be displayed on the front page of the publication.

The Internet has changed the way we acquire knowledge and stay updated on previous and latest trends. Based on the knowledge that the paper is daily updated, we can use this as a news source for extracting knowledge. The other factor is that we know that the newspaper we are collecting data from is a financial newspaper and tend to write other corporations.

RSS Feeds RSS feeds or *Really Simple Syndication* is a type of web feed, where the basic idea of restructuring information about information dates back to the 90s. Which can help to keep track of the latest news, views, or podcast in a single news generator.

Different websites use RSS to pass updated information, for example, blogs, news headlines, or podcasts. As soon as a content is published, the feed will be updated according to the content and attributes to the content. The news generator will check for RSS feeds for new content, that allows the content to be passed from a website to website or from website to the user. The RSS feed has a summarized content in text and metadata, for example, the author's name or the date of publication.

The standard format for the RSS is in XML, which is compatible with different programs. Different applications use RSS to presents the data, where RSS also benefits timely updates from favorite websites. This means that the user can store their favorites media in one place and create a feed based on their favorite sites.

```
<rss xmlns:ns0="http://www.w3.org/2005/Atom" version="2.0">
  <channel>
    <title>Alle</title>
    <link>https://www.hegnar.no/</link>
    <description />
    <language>no-bokmaal</language>
    <managingEditor>kontakt@byte.no (Administrator User)</
    managingEditor>
    <pubDate>Thu, 10 Jan 2019 12:43:34 +0000</pubDate>
    <lastBuildDate>Thu, 10 Jan 2019 12:42:43 +0000</
    lastBuildDate>
    <generator>eZ Components Feed dev (http://ezcomponents.
    org/docs/tutorials/Feed)</generator>
    <docs>http://www.rssboard.org/rss-specification</docs>
    <ns0:link href="https://www.hegnar.no/rss/feed/all" rel="
    self" type="application/rss+xml" />
    <item>
      <title>Omsetningsrekord for Oslo S Shopping</title>
      <link>https://www.hegnar.no/Nyheter/Boers-finans
    /2019/01/Omsetningsrekord-for-Oslo-S-Shopping</link>
      <description>&lt;p&gt;Oslo S Shopping &#248;ker med&amp
    ;nbsp;8,2 prosent fra 2017.&lt;/p&gt;</description>
      <author>kontakt@byte.no (Marianne  Loland)</author>
      <guid isPermaLink="false">2180470
    b806b85faf10f3867148b7d94</guid>
      <pubDate>Thu, 10 Jan 2019 12:42:07 +0000</pubDate>
    </item>
    </channel>
</rss>
```

Listing 1: Hegnar XML structure.

In Listing 5.1, the raw material of the RSS feed is presented. All the data presented can, in theory, be used, but can be unnecessary since all the data isn't aquired for process. For example, if we wish to list a database with only publication dates and titles, the ideal thing to do is to extract the raw text from both $< pubDate >$, and $< title >$, and later apply them to a schema or database. If we later in the process which to apply another variable to the data, such as $< description >$, another iteration has to be made to verify that the correct raw-data-description is added to both the publication and the title values. Even though the data is understood, the requirements are not set before the process has started, implying that either the requirements have been changed along the process, a misunderstanding of the data has

been made, or the requirements are not fully established.

In addition, the collection of data from newspapers can, in some cases, turn out to be presented as limitations to the methodology, such as pay to read articles or metadata that are not presented in the data itself. Two factors need to be taken into account in that case, for example, if a small company is not mentioned compared to a bigger company. This is something that needs to be accounted for when reviewing the data later on. Depending on the demographics of a corporation and the newspaper, there is a higher possibility that the small companies out in the districts are not mentioned as much compared to bigger and more popular companies. On the other side, there is a higher possibility that smaller companies get mentioned if the newspaper is from the same district.

5.2 Development

The idea behind the research is to review online newspapers, so we which to select a publication that can be substantial for the study. We know that a company can potentially be written about in a paper. As presented by Rohanizadeh and Bameni, the ideal is to formulate the goal of the extraction(Rohanizadeh and Bameni, 2009). In our case, we which to review papers that might contain organizations. By this, we don't wish to extract documents that are known for writing about sport or fashion. adressa.no, dn.no, and hegnar.no are all excellent potential candidates for obtaining data, but for the research, the scope is narrowed down to one newspaper. Hegnar was used since its well known for presenting financial news for to public as well as mentioning different companies.

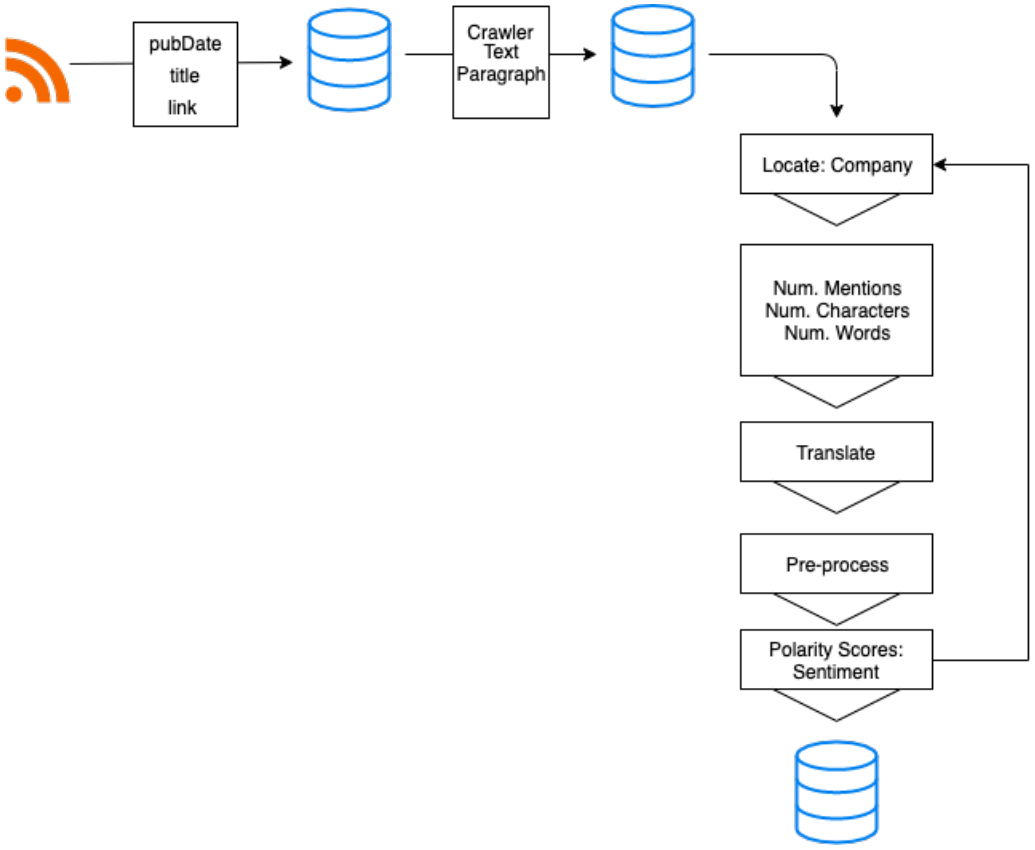

Figure 9: Research Process. Note: The database indicates the CSV file.

The first interval was to extract the data using the RSS feed from the newspaper and store them. For this, a virtual machine is used to run each 10 minutes from *January 2019 to July 2019*. Figure 10 presents the command that was used to run the code and extract the XML format from the RSS and store them. A.2 offers the code that was initialized each 10 minutes.

```
ubuntu@news-data-machine */10 * * * * /usr/local/bin/python3
/home/ubuntu/Programs/rss_feed_crawler >> crontab.log 2>&1
~
~
~
~
```

Figure 10: The Crontab Command from the Terminal.

All the XML files were relocated to the local machine for further processing. After reviewing the RSS feed, a choice was made on what element to extract from RSS feeds. The most important part of the research is to review the raw text that is displayed on the diffent webpages. To optain this result we need to focus on the *link* tags in the XML. Table 1 presents the different elements that are represented inside the Item elements. The *text* is an important part because it contains the raw material that is needed for further re-processing. Furthermore, the *title*, *description* and the *pubDate* is extracted as well to give a timestamp to each article and to identify the article.

Table 1: A representation of each Item element in the XML.

XML Tag	**Tag Description**
<title>	The Title of the Item
<description>	The description of the Item which can be either text or HTML
<link>	The URL of the full content on the original website
<guid>	A unique identifier, often the URL of the full content
<pubDate>	the Item's publication date
<category>	The category of the item given by the publisher

Each of the $< item >$ elements were extracted from the XML files and placed in a CSV file. Each tag where placed with their respective values. Figure 11 presents an overview over all the values in the CSV file. At this stage, the text was not extracted, but the CSV was reviewed to check that the values were valid for the next stage in the process. An important factor that needs to be commented on is that the dates were labeled differently for the RSS to be read. Most of the modules run on a different datestamp compared to the Norwegian date stamp that presents dates in day, month, and year. This was converted to year, month, and day so that the data could be presented in different graphs if needed, but did not have any impact on the analysis when placing them in Numbers program for Mac.

	pubdate	date	title	description	link	paper
0	Tue, 05 Feb 20...	05-02-2019	Andreassen: P...	Eika-økonom J...	https://www.he...	Alle
1	Wed, 06 Feb 2...	06-02-2019	Aker BP skuffer		https://www.he...	Alle
2	Wed, 06 Feb 2...	06-02-2019	Orkla: 4. kvartal	Orkla leverer t...	https://www.he...	Alle
3	Tue, 26 Mar 20...	26-03-2019	Uber kjøper riv...	Rekord-deal i...	https://www.he...	Alle
4	Mon, 18 Feb 2...	18-02-2019	Avinor har delt...	Avinor har i løp...	https://www.he...	Alle
5	Mon, 18 Feb 2...	18-02-2019	Saudi-Arabias...	Saudi-Arabias...	https://www.he...	Alle
6	Mon, 20 May 2...	20-05-2019	Stig Myrseth a...		https://www.he...	Alle
7	Thu, 28 Mar 2...	28-03-2019	Svenske storb...	Analytiker tolk...	https://www.he...	Alle
8	Thu, 21 Mar 20...	21-03-2019	Dette skjer i da...	Alt dette er på...	https://www.he...	Alle
9	Wed, 10 Apr 2...	10-04-2019	Bilsalget falt fo...	Bilsalget i Kina...	https://www.he...	Alle
10	Tue, 15 Jan 20...	15-01-2019	Det mest attra...	Analytiker Ole-...	https://www.he...	Alle
11	Thu, 14 Feb 20...	14-02-2019	SalMar tar nytt...		https://www.he...	Alle
12	Tue, 28 May 2...	28-05-2019	Thinfilm øker u...	Enda rødere fo...	https://www.he...	Alle
13	Tue, 28 May 2...	28-05-2019	Streikesmell fo...		https://www.he...	Alle
14	Tue, 28 May 2...	28-05-2019	Styreleder kjøp...	Øker eksponeri...	https://www.he...	Alle

Figure 11: Screenshot of the CSV after extracting data from the XML. Note: Rows present each paper, column Presents the item tags in the XML.

Furthermore, the text was extracted from each of the URLs in the rows. The HTML (Hyper Text Markup Language) was reviewed to locate what part of the text was needed for crawling. BeautifulSoup was used for this task and to identify the text within the HTML.

Table 2 represents the most basic approach of locating values in the HTML tags. The idea for this task is to evaluate the importance of the

Table 2: A representation of HTML Tags.

HTML Tag	**Tag Description**
<h1>	The headline tag of the text
<p>	The paragraph tag of the text
	div defines the division or a section in a document
id attribute	id attribute defines a unique value of an element
class attribute	The class attribute defines one or more class names for an element

text and the purpose of the extracting, as well as what the end goal is. How much do we need and how much is sufficient for the task? If we would crawl several newspapers at the same time, then each of the HTML architecture had to be reviewed, since each newspaper has its own way of representing the HTML architecture and how they name the attributes.

After reviewing the HTML, the crawler is set up accordingly. Appendix A.5 represents the crawler for all the URLs in Hegnar newspaper. It is scoped down to extracting the text and not focusing so much on the headline. The headline was already extracted in the first step of the process, so the headline doesn't need to be focused on. By extracting the text-only and not with the tags, we save time in removing them afterward. The text needs to be re-processed either way, through NLP, but that's a task that can be shortened down if possible. Some headlines might contain tags, but they're not the critical part. The most crucial part of the HTML is paragraphs (e.g $< p >$). These tags are the ones we wish to extract. The headlines can be used as identifiers for the main text if we wish to use them as identifier.

```html
<!DOCTYPE html>
</html>
<head></head>
<body>
<h1>Headline</h1>
<div class="main_text">
    <p id="first">This is the first paragraph</p>
    <p id="second">This is the second paragraph</p>
    <p id="third">This is the third paragraph</p>
</div>
</body>
</html>
```

Listing 2: HTML example structure.

As presented on Figure 12, the process of crawling is done with iterating over all the *link*'s and retrieving the text from the respective site.

	pubdate	date	title	description	link	paper	text
0	Tue, 05 Feb 20...	05-02-2019	Andreassen: P...	Eika-økonom J...	https://www.he...	Alle	Eika-økonom J...
1	Wed, 06 Feb 2...	06-02-2019	Aker BP skuffer		https://www.he...	Alle	Aker BP ga tall...
2	Wed, 06 Feb 2...	06-02-2019	Orkla: 4. kvartal	Orkla leverer t...	https://www.he...	Alle	Orkla leverer t...
3	Tue, 26 Mar 20...	26-03-2019	Uber kjøper riv...	Rekord-deal i...	https://www.he...	Alle	Rekord-deal i...
4	Mon, 18 Feb 2...	18-02-2019	Avinor har delt...	Avinor har i løp...	https://www.he...	Alle	Avinor har i løp...
5	Mon, 18 Feb 2...	18-02-2019	Saudi-Arabias...	Saudi-Arabias...	https://www.he...	Alle	Saudi-Arabias...
6	Mon, 20 May 2...	20-05-2019	Stig Myrseth a...		https://www.he...	Alle	I den siste uker...
7	Thu, 28 Mar 2...	28-03-2019	Svenske storb...	Analytiker tolk...	https://www.he...	Alle	Analytiker tolk...
8	Thu, 21 Mar 20...	21-03-2019	Dette skjer i da...	Alt dette er på...	https://www.he...	Alle	Alt dette er på...
9	Wed, 10 Apr 2...	10-04-2019	Bilsalget falt fo...	Bilsalget i Kina...	https://www.he...	Alle	Bilsalget i Kina...
10	Tue, 15 Jan 20...	15-01-2019	Det mest attra...	Analytiker Ole-...	https://www.he...	Alle	Analytiker Ole-...
11	Thu, 14 Feb 20...	14-02-2019	SalMar tar nytt...		https://www.he...	Alle	SalMar, som to...
12	Tue, 28 May 2...	28-05-2019	Thinfilm øker u...	Enda rødere fo...	https://www.he...	Alle	Enda rødere fo...
13	Tue, 28 May 2...	28-05-2019	Streikesmell fo...		https://www.he...	Alle	SAS melder o...
14	Tue, 28 May 2...	28-05-2019	Styreleder kjøp...	Øker eksponeri...	https://www.he...	Alle	Øker eksponeri...

Figure 12: Screenshot of the CSV file after retrieving the text. Note: Rows present the papers. Last column "text" present the text.

5.3 Selection of Companies

The selection of companies is based on the largest companies in Norway by their revenue[18]. For the research, I wish to review the data that are based on the top 15 companies. The list of companies is used to check if its a correlation between the trends in the news and the revenue list.

[18] `https://en.wikipedia.org/wiki/List_of_the_largest_companies_of_Norway`

Table 3: Wikipedia ranking of largest Company in Norway by their revenue from 2006.

Company	**Acronym**	**Revenue**
Equinor	EQ	431,112
Norsk Hydro asa	NHA	196,234
Telenor asa	TA	91,077
Aker asa	AA	79,892
Orkla asa	OA	52,683
Aker Kværner asa	AKA	50,592
Total E&P Norge as	TEN	50,577
ExxonMobil Exploration and Production Norway as	EEPN	49,680
Yara International asa	YIA	48,261
Esso Norge as	ENA	45,408
Kommunal Landspensjonskasse	KL	43,581
NorgesGruppen asa	NA	36,631
Storebrand asa	SA	34,074
Norske Skogindustrier asa	NSA	28,812
DnB NOR asa	DNA	28,439

Table 3 presents the popular companies in Norway during 2006 and can be used as an indicator if the index of each company follows the same pattern of popularity in the papers.

5.4 Iterations

The iteration of retrieving knowledge from the text consists of different steps, one of which is to retrieve the amount of data that each article consists of. The companies are primarily located based on their names as a simplified approach. A company name consists of a capital letter which identifies the company in the text. The same approach can be used for names or addresses. As long as the company is written with the capital letter, we can assume that either the article is about a company or is mentioned in some sort of context. Appendix A.6 presents the process of retrieving the companies in the text. By this stage, no pre-processing of the text is made. On the example presented on listing 3, the example show how a company can be located through a simple *contains()* method in Pandas.

In the documentation[19] of the method is presented with a few other options. A set of rules is set on line 1, such as na-values (NaN) and case (case sensitive) values. NaN values are presented as null or empty values in the data so that it skips the empty values if there is some, and the case sensitive rule is set so that we can locate the specific word in the data.

```
1 df = df[df['text'].str.contains("Equinor", na=False, case=
    True)]
2 df['mentioned'] = df.text.str.count("Equinor")
```

Listing 3: An example of how Equinor is located.

Consider the text below. Widerøe is being mentioned up to three times during the text, but the context of the text is not being accounted for. The human understanding of the text can consider the text as being negative and will know that its' a company that is being talked about, and not just mentioned in some other context.

> *"Over half of the employees in* ***Widerøe*** *at Bergen flight station had to take a leave because of water leak in their premises.* ***Widerøe's*** *spokesperson said the following "****Widerøe*** *apologizes for the circumcises and believes they will be at full operation tomorrow morning."*

[19] `https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.Series.str.contains.html`

Next, we wish to extract more data from each article. Appendix A.6 also presents the script that was used to identify the amount of time the name of a company is identified in the text. Each of this identifications are labeled as *mentioned*, and are added to their own column. The amount of *total words* and *characters* are extracted as well. As presented in figure 13, a new CSV file is created, or the previous one is updated with update methods in the pandas module.

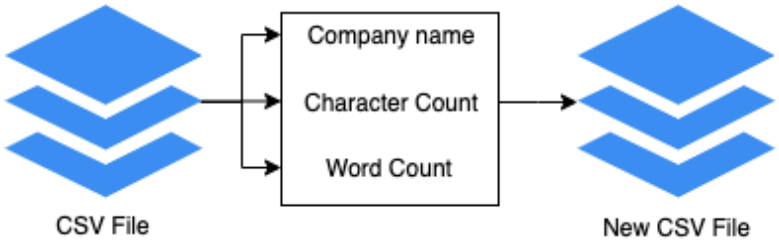

Figure 13: Locating the Companies and creating a new CSV File.

Translation is a crucial part of the analysis. Since we wish to do further analysis of the text later in the process. For this task to succeed we need to use a translation tool that works for this purpose. As presented in Section 3.10, there are rich resources in English that can be applied to the Norwegian language. Previous research, such as Hammer et al. shows that translation can be applied to rich Norwegian text.

Python Google Trans Module is used for the task of translating the Norwegian text to English. One of the reasons that this module is used is that it's a free tool that provides a variety of languages from the google API, among others, from Norwegian to English translation and vice versa. By using a free translating tool, the limitation of economy boundaries does not become a reality. As of 2019, Google already provides Cloud Translations through its services but has a limitation when it comes to the maximum amount of free characters that can be translated before the user has to pay.

Each of the rows in the data where the Norwegian text is placed, the text is being translated and placed in a new column for the next step of re-processing. At the end of the process, the table should be looking like table

4. It is important to emphasize the problems that occurred during this part of the process. By using the free module to much, google might block the IP for the request to the translation tool. This was solved by either waiting 24 hours before translating the text or by using a Virtual Private Network (VPN). Another solution can be is to remove the rows where the data is collected and run the script for translation again. Run the script for those rows that are not being translated and passing the English text to the new CSV file.

Table 4: The headers this far in the process.

index	main index	date	pubdate	description	link	paper	text	total words	mentioned	count	english text

Table 4 presents the head of how the table will be seen after adding the translated text to the new table.

Re-processing. The text is processed in one step instead of dividing it into several small steps. Appendix A.8 presents the script of cleaning the text and adding the sentiment score to the new column. Previous research present re-processing in several steps, such as removing characters, or removing words. In theory, it's processed in on step. The only module that is used in this case is the NTLK for natural language transformation. The part that takes the text and removes special characters and splits the stopwords is presented in listing 4.

```
df['paper_text_processed'] = df['english'].apply(lambda x: re
    .sub('[,\.!?]', '', x))

df['paper_text_processed'] = df['paper_text_processed'].apply
    (lambda x: ' '.join([word for word in x.split() if word
    not in (stop_words)]))
```

Listing 4: Re-process the data.

Sentiment column is added with the sentiment method, presented in listing 5. The NLTK module has a Sentiment Intensity Analyzer that reviews the sentiment of the text and adds the polarity score to the new column. With this score, we can verify if the total amount of the text is either positive or negative during the time period. Adding the sentiment score does not take into account how many articles there are in total. If there is only one, then that one article represents the sentiment of the company for the time of mining RSS feeds. This is later taken into account when filtering.

```
from nltk.sentiment import SentimentIntensityAnalyzer

def sentiment(x):
    sentiment = sia.polarity_scores(x)['compound']
    return sentiment
```

Listing 5: Method for adding Sentiment Score.

Narrowing down the research to some companies based on their score is the next step of the analysis. All the data are put in a different matrix for review. There are two factors that need to be taken into consideration when analyzing the data. One, there needs to enough articles to be analyzed. The higher the number of materials, the better the analysis. The second part represents the number of mentions the article has. This is why the larger the amount of articles there is, the bigger is the possibility of a company being mentioned several times in an article can become. As mentioned before, the previous step does not take into account that there might be only one article being labeled with sentiment or four hundred. The equation below represents how the articles we wish to analyze in the next step are identified.

$$Company\ To\ Analyze = num.Articles + Highest\ mean\ Score\ of\ Mentions \tag{1}$$

In the art of speech, the probability of an organization being mentioned increases, the more the organization is being mentioned in a particular text. Given that we don't know the context of the text, we wish to identify the organization by counting how many times the organization's name is written in the text. Another factor that we need to take into account is the style of how the paper writes it's papers. After reviewing some of the papers, we know that hegnar.no usually presents statistical organizational information, such as how their revenue is if something is being bought or sold. In light of that, we know that a company can be mentioned one time, but we don't know in what context.

6 Results

This chapter presents the results in detail from the research. The first section presents the data and the results from identifying companies in the articles. The analysis is based on an empirical evaluation of the data. The second section presents the results based on how the selected organizations are presented in the media, and to what knowledge we get from the media. Unfortunately, the analyzed text can not be presented as it breaks copyright of the newspaper, but some documents are being referred to, based on their polarity score.

6.1 Identifying Companies(RQ 1)

The first section presents the results of identifying companies in the dataset. Figure 14 shows the total percentage of the articles that were located in the data based on their names. Out of all the articles, 11% of the article were identified based on their names. That concludes a total of 1305 articles out of 10652, where 15 companies were mentioned during the mining period.

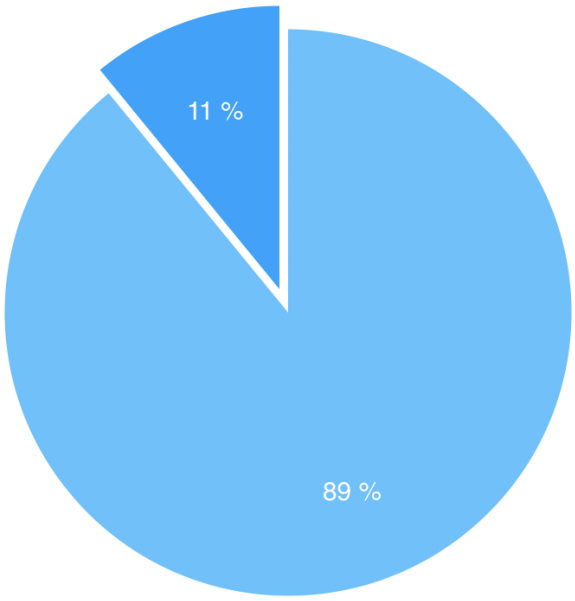

Figure 14: Number of articles where all companies are mentioned compared to the total of articles.

Table 5: Top-15 companies ranked based on number of articles.

Company	**Num. articles**
Equinor	423
Telenor	243
Hydro	228
Storebrand	122
Esso	115
Orkla	110
ExxonMobil	35
Yara	11
Aker	10
NorgesGruppen	3
Kommunal	2
NorskeSkogIndustrier	1
Total	1
DnB	1
AkerKverner	0

As presented in table 6, The top 3 companies that were written mostly during the period where Equinor, Telenor, and Hydro. Equinor with a total of 423 articles, Telenor, with 243 articles and Hydro, with 228 articles. These companies are also being labeled as the top 3 companies in Wikipedia based on their revenue, presented on table 3, on page 50. Even though their ranking is not parallel to what table 6 presents, it's close enough to justify a correlation to what table 6 presents in the top 3.

The mean presented in table 6 also shows the average amount of words written in each article during the period of extraction. It's important to emphasize the amount is based on the Norwegian language. We can see that the amount written can have variation depending on the context, which in this case is unknown, but can represent as a trend for the paper. Furthermore, there is no correlation between the amount written in an article and the number of papers collected. As presented in 6, a larger amount of papers does not give more significant probabilities for a larger amount of words. For example, Equinor comes out with the top of 557 number of words written on 423 articles, while the next Hydro has a total 439.11 number of words, with 228 articles. Even though Telenor has 243 articles, which is 15 more articles than Hydro, the number of written words are slightly less where Telenor is mentioned. Another example is the difference between Orkla, with 411.77

Table 6: Top-15 companies based on their revenue. Note: ∅ indicates mean values.

Company	Revenue	Num. articles	∅ Num. words	∅ Num. Char
Equinor	431,112	423	557	3532.17
Hydro	196,234	228	439.11	2790.26
Telenor	91,077	243	423.97	2699.38
Aker	79,892	10	215	1368.9
Orkla	52,683	110	411.77	2606.02
AkerKverner	50,592	0	0	0
Total	50,577	1	90	616
ExxonMobil	49,680	35	311,25	2004.88
Yara	48,261	11	375.1	2411.75
Esso	45,408	115	188	1140.33
Kommunal	43,581	2	287.83	1756
NorgesGruppen	36,631	3	257	1650.33
Storebrand	34,074	122	395.42	2523.65
NorskeSkogIndustrier	28,812	1	135	888
DnB	28,439	1	547	3460

words on 110 articles, and Esso, with 188 words on 115 articles. 15 shows a visual representation of the number of articles collected.

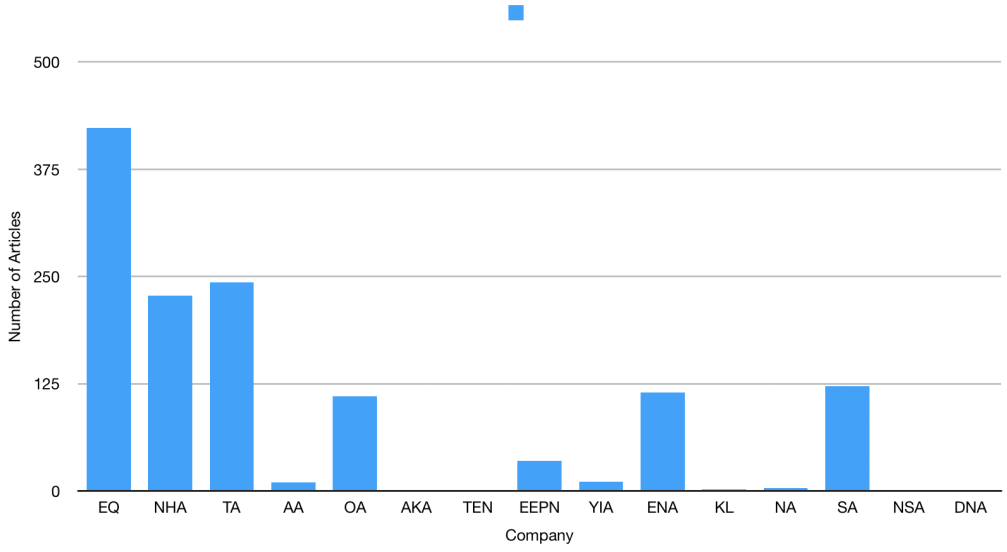

Figure 15: Total Number of articles of each company.

The identification of companies based on the methodology are represented in the data to a certain degree, but do not represent the context. This is the reason why the text is identified on their mentioned or name in the text. Table 7 presents all the mentioned of a company during the time period. The matrix represents the mentions from 1 to 5, where the higher the number of mentions is, the higher is the probability that a company actually is being mentioned, based on the company name in the text. This is why text, where a company is mentioned 1 or 2 times, can in some cases be false-positive mentions. Also, knowing that the papers tend to write about financial facts from time to time, the paper can not present much sentiment about the company. The methodology uses an approach where more than three can be assigned as validity for further research.

As the matrix presents, the bigger the amount of papers recorded implies that the possibility for a company being mentioned gets bigger. However, this is not true in all cases. In table 6 Orkla has a total of 110 newspapers and is ranked as number 6 on the same table. In table 7 Orkla is the company ranked highest on average mentions with an average of 2.56 mentions on fewer papers compares to the rest of the other companies. Moreover, Storebrand is mentioned 2.04 on average, with a total of 122 papers, and Telenor third, with 1.98 on average with 243 papers.

6.2 The Representation of the Company in the Media(RQ 2)

The representation of companies based on the sentiment in the text. The method used in the research is based on a translation of words and later applied an overall sentiment score: -1 (negative sentiment), 0 (neutral sentiment), and 1 (positive sentiment). Table 7 present the average polarity score of a company based on the papers. The table does also mentions how many papers are being written about each company each day. The data mining process lasted 181 days. Based on this, the newspaper wrote, on average 2.33 papers each day about Equinor, 1.25 papers on average about Hydro, and 1.35 papers about Telenor.

It's important to identify the false positives as one text is not considered enough validity for the analysis. For example, Total, Kommunal, Norske

Table 7: Number of times a Company is mentioned in an article. Note: ∅ indicates mean values.

Company	**1**	**2**	**3**	4	**5⩽n**	∅
Equinor	361	74	41	21	26	1.66
Hydro	186	37	5	0	0	1.2
Telenor	145	42	27	12	18	**1.98**
Aker	9	1	0	0	0	1.1
Orkla asa	60	14	8	19	9	**2.56**
AkerKverner	0	0	0	0	0	0
Total	1	0	0	0	0	1
ExxonMobil	29	1	2	1	2	1.48
Yara	10	2	0	0	0	1.16
Esso	82	24	2	6	1	1.43
Kommunal	2	0	0	0	0	1
NorgesGruppen	1	1	0	1	0	2.3
Storebrand	70	15	14	16	8	**2.04**
NorskeSkogindustrier	1	0	0	0	0	1
DnB	1	0	0	0	0	1

Skogindustrier, and DnB are all depicted as false positives. This means that even though they have a high polarity score it's only related to one article. This doesn't describe the reality, on the other hand, it can describe the article on day spesific day. In table 6 all the companies have less than ten papers written about them. Furthermore, Aker, Yara, ExxonMobil have all less than 40 papers written about them, giving a doubtful validity of the poliarity score. On the other hand, the sentiment score gives a hint of how a company is being represented by the media regardless of how many mentions they have and how many papers are being written about the company.

In light of this , Norges Gruppen has three papers written about the company but has over average mentions compared to the other companies. The company got a polarity score of 0.73, which gives a positive mention in the media but does not give enough data for how the company is being perceived by the media during a period of time. On the other hand, the result can pinpoint how the company was perceived in those few days.

Table 8: Average number of articles over the past 181 days. Note: ∅ indicates mean values.

Companies	**∅ Num. articles**	**∅ Num. words**	**∅ Num. Char**	**∅ Num. Sentiment**
Equinor	2.33	3.077	19.51	0.8
Hydro	1.25	2.42	15.41	0.69
Telenor	1.35	1.18	7.56	0.74
Aker	0.05	1.18	7.56	0.96
Orkla	0.60	2.27	14.39	0.79
AkerKværner	0	0	0	0
Total	0.005	0.49	3.40	0.49
ExxonMobil	0.19	1.71	11.07	0.70
Yara	0.06	2.07	13.32	0.84
Esso	0.63	1.96	6.30	0.50
Kommunal	0.01	1.59	9.70	0.002
NorgesGruppen	0.016	1.41	9.11	0.73
Storebrand	0.67	2.18	13.94	0.82
NorskeSkogindustrier	0.005	0.74	4.90	-0.38
DnB	0.005	3.02	19.11	0.99

The only company that got a negative score by the media was Norske Skogindustrier, with a polarity score of -0.38, but only one paper was extracted. In spite of this, it also represents a possible validity, but only 1 mentioned has to be accounted for as it may be false-positive, represented in on table 7.

In companies got on average positive sentiment in the text, as represented on figure 16 and table 8. Based on this analysis, DnB scored the highest score of 0.99 but did also only have one mention on one paper. Next came Aker with a score of 0.96 on ten papers, and an average mention of 1.1.

For further analysis, there where extracted three companies with the highest score based on mentions and papers were analyzed further. As can be seen in table 7, the top mentions where Telenor, Orkla asa, and Storebrand. These companies are most likely to have a high probability that the company being mentioned and that the paper actually is about the company. Similarly, there are enough papers to narrow the papers to 3 or more mentions for all the three companies. In the hope that we wish to review the media's perception of a particular company, these three companies are the companies that stand out based on sentiment and mentions.

Companies can in several occasions be identified based on their names. But

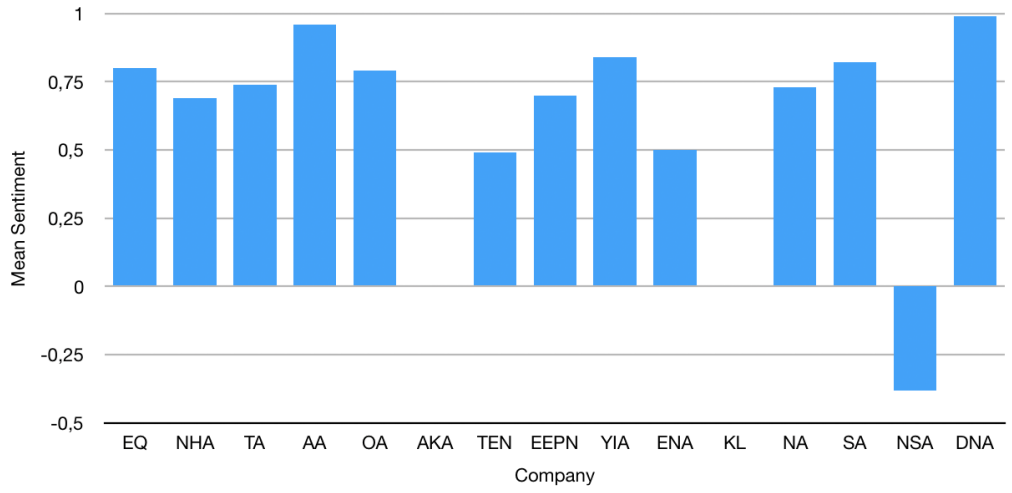

Figure 16: The average sentiment of all the articles during the time period.

as it will be presented further on the results, there are several factors that have to be taken into account when analyzing the text. As has been stated before, reading the newspaper make a high impact on what kind of text that is being analyzed, such as financial factors, numerical analysis, or just facts presented by the press about a company, such as hegnar.no. When it comes to a company poorly based on name, yes, a company extraction can identified only based names with capital first letters, but we have to accompany another iteration that can narrow down the false positives and the "just mentions", when we don't have any context about what the text is about.

Table 9: Number of articles based on 3 or more mentions. Note: ∅ indicates mean values.

Company	**Num. Articles**	**∅ Num. Mentions**	**∅ Num. Sentiment**
Telenor	55	5	0.68
Orkla asa	36	5	0.78
Storebrand	36	4	0.93

Table 9 presents the result of the top 3 companies after extracting the papers based on 3 or more mention. Likewise, the table presents the average mentions of the papers, as well as the average sentiment of the paper. As illustrated in figure 17, 19% of the mentioned papers are extracting, meaning that 55 papers have an average of 5 mentions in them. Telenor is the company that has the lowest score on sentiment in the papers, with an average of 0.68, while Orkla comes at second place, with an average of 0.78 and Storebrand with 0.93.

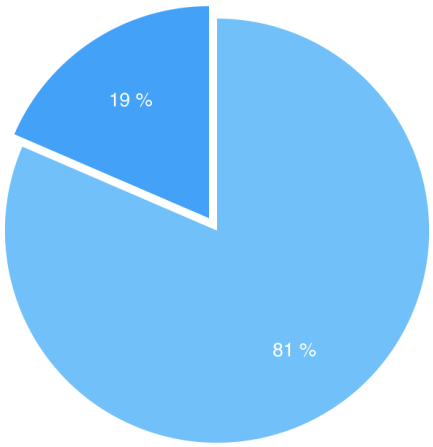

Figure 17: 19% of articles where Telenor is mentioned 3 or more times. 81% of articles where Telenor is mentioned 2 or less times.

Positive loaded papers of Telenor. Telenor is being represented with an average sentiment of 0.68 on table 18. As the figure presents, the overall sentiment of Telenor is mostly above the average score, with some negative sparks.

After reviewing papers the results indicate that most of the papers that were positive loaded mostly contained facts about how well or poorly Telenor had done compared to other times (e.g., a week ago, a month ago, a year ago). In most cases, the papers were about the stock market and also included other

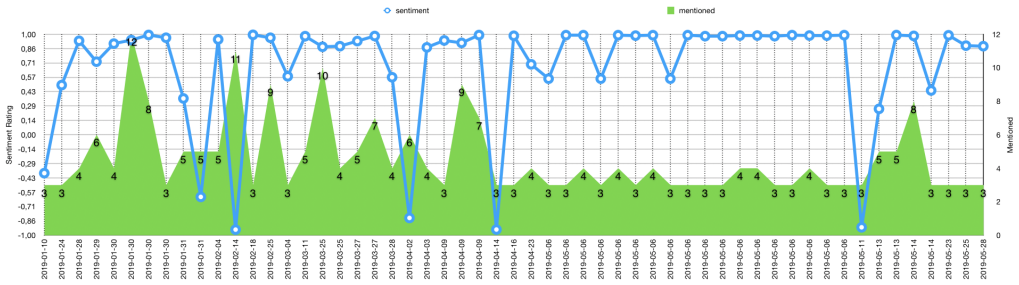

Figure 18: The sentiment and total Telenor mentions in an article where the company is mentioned 3 or more times.

companies. On the other side, the negatively loaded papers showed that the analysis picked up the negative loaded sentiment in the text. Furthermore, it also indicates that there is no correlation between how many times Telenor is mentioned and the negative loaded sentiment in the text. Figure 18 presents that the company was mentioned three times in a paper written in 2019-05-11, but has almost the same the polarity score on another paper written earlier that year in 2019-02-14.

As table 10 presents, the Sentiment Intensity Analyser from NLTK (page 54) managed to pick up the negative sentiment of 6 papers from Telenor. Almost all the articles were written about the Company Telenor, except for the first paper. The first paper written in 2019-01-10 referred to a sub-company in Thailand where the stock went down and had a negative impact on Telenor.

Table 10: Telenor's negative loaded articles with date and content.

Date	About
2019-01-10	Stock went down[20]
2019-01-31	Downsizing in 2018[21]
2019-02-14	Telenor breaks rules in sales[22]
2019-04-02	Telenor is downsizing[23]
2019-04-14	Technical problem[24]
2019-05-11	Telenor is monitoring their employees computers[25]

Negative loaded papers of Telenor. Table 19 presents the sentiment and mentions of the articles where Telenor was mentioned 2 or less times. The sentiment score does also tracks negative sentiment in the text. Still, after reading the negative articles, most of the papers were marked as financial facts, or that the company was mentioned with other companies. The table presents no correlation between mentions and the sentiment in the text but shows that Telenor has a higher number of articles where it is mentioned only one time. Compared to figure 18, figure 19 has a higher score of sentiment, but a lower validity referring to Telenor as a company alone. The article[26] written 2019-03-28 present such case where several companies, as well as Telenor, is being mentioned with stock details or financial facts. Never the less, the article[27] written the 2019-01-25 writes about downgrading Telenor because of the cases that are happening in Malaysia and Thailand. The

[20] https://www.hegnar.no/Nyheter/Boers-finans/2019/01/Telenor-analytiker-Dette-var-en-negativ-overraskelse

[21] https://www.hegnar.no/Nyheter/Boers-finans/2019/01/1.800-faerre-ansatte-i-Telenor-i-2018

[22] https://www.hegnar.no/Nyheter/Boers-finans/2019/02/Telenor-faar-smekk-for-telefonsalg

[23] https://www.hegnar.no/Nyheter/Boers-finans/2019/04/Telenor-vil-kutte-konsernstaben-med-500-millioner

[24] https://www.hegnar.no/Nyheter/Boers-finans/2019/04/Problemer-hos-Telenor-rammer-trygghetsalarmer

[25] https://www.hegnar.no/Nyheter/Boers-finans/2019/05/Telenor-maatte-stoppe-ulovlig-overvaaking-av-ansattes-PC-er

[26] https://www.hegnar.no/Nyheter/Boers-finans/2019/03/Oslo-Boers-aapner-svakt-opp

[27] https://www.hegnar.no/Nyheter/Boers-finans/2019/01/Nedgraderer-Telenor

papers present a fall in the stock market based on the situation Telenor had around January 2019.

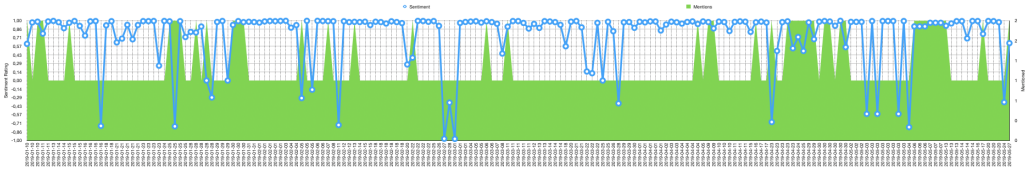

Figure 19: The sentiment and mentions in an article where the Telenor is mentioned 2 or less times.

Positive loaded papers of Okla. Orkla had 36 papers after filtering out where the papers mentioned Orkla 2 or fewer times. As presented in figure 20, 34% of the total amount got filtered out and had an average of 0.78 positive loaded sentiment on average. Compared to Telenor, Orkla had 14 papers less but had the same mentions on average. Figure 21 shows that there is no correlation between the sentiment loaded papers and the number of mentions in the paper.

The analysis shows that one article written 27/03/2019 was negatively loaded. The paper presents a negative loaded text where the Stock went down. Furthermore, papers that were positively loaded, but were close to the neutral polarity score zero had a negative impression. For example, the paper written in 2/5/2019 wrote that Orkla had to pay for half of the expenses of two broken satellites[28]. An evaluation of a paper can be subjectively but is identified as unfavorable for the company, thereby negative presented in this case. The paper from 7/5/2019 had the most mentions, with a positive polarity score at 0,99. Even though the papers describes financial facts about the firms quarterly earnings the analysis managed to identify the positive loaded text. The papers mostly presents numbers and results about the firm and Orkla's subsidiaries [29].

[28] `https://www.hegnar.no/Nyheter/Politikk/2019/05/Nasa-tapte-seks-milliarder-Hydro-selskap-maa-punge-ut`

[29] `https://www.hegnar.no/Nyheter/Boers-finans/2019/05/Orkla-legger-frem-gode-kvartalstall`

[30] `https://www.hegnar.no/Nyheter/Boers-finans/2019/03/Jekker-ned-Orklae`

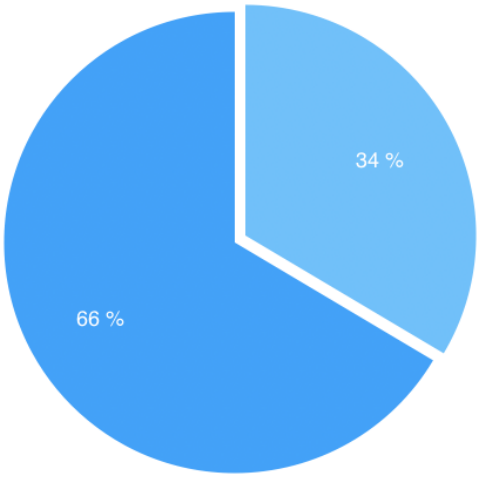

Figure 20: 34% of articles where Orkla is mentioned 3 or more times. 66% of articles where Orkla is mentioned 2 or less times.

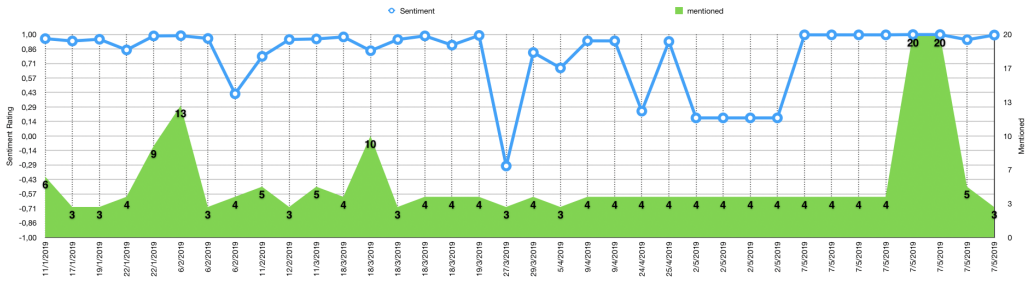

Figure 21: The sentiment and mentions in an article where the Orkla is mentioned 3 or more times.

Table 11: Orkla's negative loaded article with date and content.

Date	**About**
27/3/2019	Stock went down. Are not being recommended[30]

Negative loaded papers of Orkla. Figure 22 shows the results where Orkla was filtered based on papers where Orkla was mentioned two or fewer times. Most of the papers had only one mentioned in them, figure 20 presents this case as well. After reading the papers based on the lowers polarity score, the papers present financial facts, stock details and comparisons in the

market based on previous and new data. Furthermore, the papers present subsidiaries and not specifically Orkla. In those cases where a subsidiary is mentioned, financial facts are represented. Turnovers and mentions of other companies are presented in them as well. The paper[31] written 6/2/2019, where Orkla is mentioned two times, describes a downward adjustment compared to previous numbers that same year.

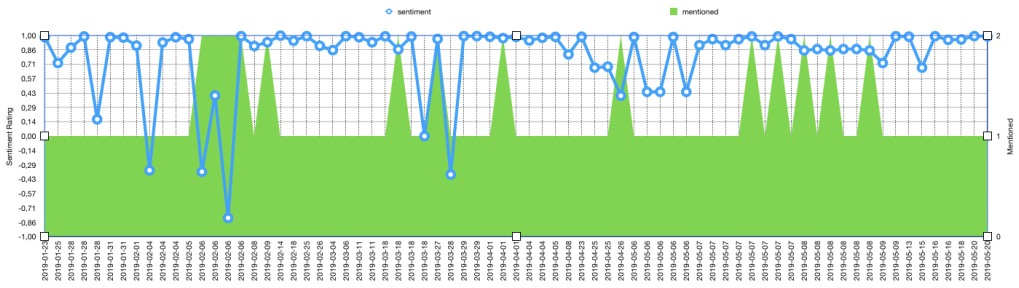

Figure 22: The sentiment and mentions in an article where the Orkla is mentioned 2 or les times.

Positive loaded papers of Storebrand. Storebrand had 36 papers after filtering by three or more mentions. Table 9 show that Storebrand had the highest score among all three companies, with a polarity score of 0.93, but had the lowest mentions among all three, with an average mention of 4. Figure 24 presents no negative sentiment in the papers. The paper that was marked with the lowest polarity score was written in 2019-05-09. This paper describes adjustment in price target and does not present any form of sentiment[32]. As figure 24 shows, three papers had the highest polarity score of 7 mentions in them. Two papers where written 2019-05-08, with both having the same polarity score of 0,99. The data shows that *description* tag was added to the XML in the RSS after being posted, which duplicated the row in the data. Rohanizadeh and Bameni presented this as falsification to what describes the reality, but is ignored as we are aware of it. The last paper from 2019-05-27 had the polarity score of 0,99 as well but presents, among other things, financial data and unrelated information to Storebrand.

[31] `https://www.hegnar.no/Nyheter/Boers-finans/2019/02/Meglerhus-justerer-ned-etter-svake-tall`

[32] `https://www.hegnar.no/Nyheter/Boers-finans/2019/05/Analytikere-tar-grep-etter-gaarsdagens-skuffelse`

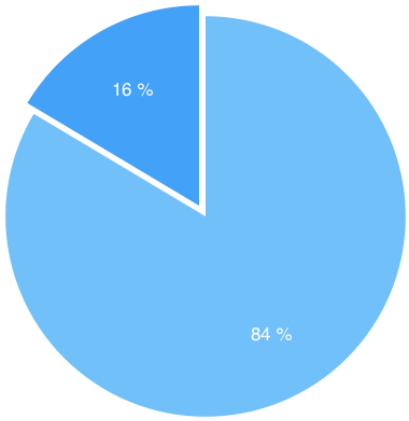

Figure 23: 16% of articles where Storebrand is mentioned 3 or more times. 84% of articles where Storebrand is mentioned 2 or less times.

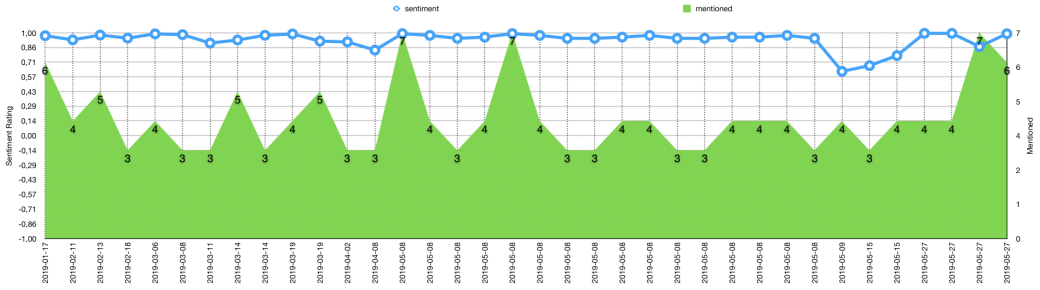

Figure 24: The sentiment and mentions in an article where the Storebrand is mentioned 3 or more times.

Negative loaded papers of Storebrand. Figure 25 presents the papers where the mentions on Storebrand where two or lower. In comparison to figure 24, figure 25 shows negative loaded sentiment on some papers. All of them are presented with one mention of the company, while 15 papers are presented with two mentions, as shown in table 7(page60) as well. The negatively loaded papers present results of the company compared to other companies and are mostly represented by the stock exchange in Norway. Several other companies and firms are presented in the texts as well. The result of this makes the low text validity for representing only one company when it comes to extracting knowledge for sentiment analysis.

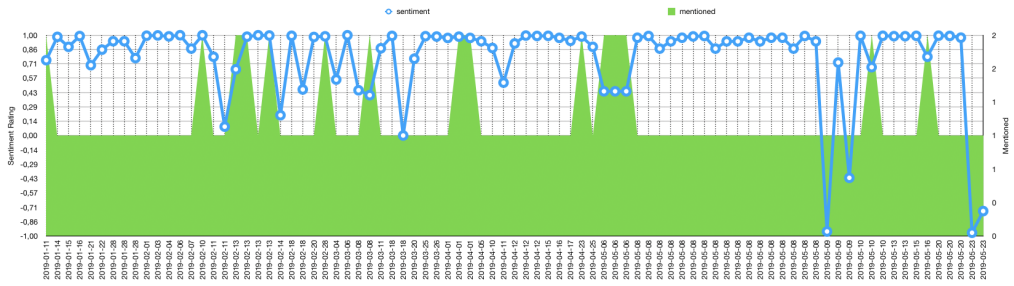

Figure 25: The sentiment and mentions in an article where the Storebrand is mentioned 2 or less times.

7 Summary, Conclusion & Future Work

This chapter concludes the thesis by summarizing the findings and discusses the possible limitations of the approach. The chapter also presents a possible improvement and follow up studies.

Summary. The thesis has explored the possibilities of data mining and analysis with already existing tools to answer the question of the media's perception of a company. This is the question that other organizations are asking themselves when trying to apply new knowledge to existing knowledge by using the media as a tool for knowledge extraction. The thesis investigates the support of using free newspapers from a specific newspaper, such as hegnar.no, to extract knowledge in an efficient way. Furthermore, the thesis investigates the support from NLTK for pre-processing text, as well as data crawling possibilities for one specific newspaper. I've have evaluated the approach of identifying companies through mentions and applied a translation sequence based on previous research, as well as applying a polarity score from NLTK to see how the media perceive a company through sentiment analysis. The open-source NLTK, BeautifulSoup, and Pandas were used to implement the components and the analysis. The advantage of NLTK is that its easy to use by a novice developer. The performance of the system architecture has similar to what previous research has done before. An advantage of this approach is that it's easy to add other companies that we want to be analyzed through sentiment analysis and mentions. Furthermore, it's possible to identify other papers and companies and see what papers are tend to mention a company the most. Thereby, narrowing down those papers to the source of

extraction. The only difference is if we wish to add another newspaper, then a new crawling algorithm needs to be added to identify the text on the web for that given newspaper.

Conclusion. The identification of the company has indicated that the system was appealing in terms of identifying a paper based on a company name and counting the total mentions in the paper. The context has not been accounted for, resulting in some unknown factors not accounted for, such as if another company is being mentioned or other information is picked up that are not related to target company. The analysis has indicated that in some cases the positive polarity score is not valid enough to be trusted as most of the papers convey financial data to the public or other numerical data about the targeted company. Based on the data, such text is being treated as positive, as the results have shown. On the other hand, the negative polarity scores has shown to be valid when identifying the companies as well as the negative sentiment in the text. The research has given potential users a new way of extracting data and applying simple tools to evaluate the data, or another architecture tool to apply other processes and help extract knowledge from the data when identifying companies. The finding suggests that there is a need to continue the research on other newspapers and maybe applying other evaluation methods to the text to identify context and/or sentiment.

RQ1 It has been shown that extracting papers based on their names brings two different outcomes to identifying a company in a text. The mentions based on a company name reveal that the company can in most occasion be identified based on their names. The results also show that a company will be presented given the companies popularity in the given time. Trends are often the reason a company is mentioned and depends on the time of the news if a company is mentioned or not. On the other hand, as results have shown, other bigger companies might not be mentioned at all. If this is the case for the bigger companies, then there might be a possibility that this is also the case for smaller companies. The data reveal that the higher amount of mentions the paper has, the higher is the possibility that the papers is

writing about the targeted company.

RQ2 I order to answer the research question, a sentiment analysis based on the polarity score was conducted. The exploration of the papers presents that the positive polarity was in most cases valid. Based on numerical values, the positive polarity can be identified as non-consistent when trying to analyze the sentiment of the paper. This being the case, the analysis performed poorly on papers with other data than text in it. While this might be true, this is not the case for the papers that were identified with negative polarity. An interesting observation is that although hegnar.no usually writes about financial facts, there are sentiment in the text based on their negative polarity that is being picked up and identifies the text as negative. In addition, it was observed that the positive polarity score closes to 0 were identified as negative text, but for the NLTK sentiment intensity analyzer was described as positive.

Limitation and Future work. The master thesis reveals interest findings in the Norwegian newspapers. Due to the different approaches to extracting knowledge, websites, re-processing, and analysis, some limitation exist. The web-crawling approach in the thesis may only be valid for the given newspaper. Different newspapers have different ways of labeling their HTML, so a new crawling algorithm has to be made for each new newspaper. Furthermore, URLs tend to change, so storing the RSS feeds for later to crawl them might not be the ultimate approach. This is the case if we wish to crawl the websites on a later occasion. In that case, valuable data can be lost, given a change in the URL. An optimal approach might be to store the plain text from the website and crawl the RSS feeds directly instead of storing them for later. Most of the newspapers tend to have pay-to-read features. Even though the extracted papers where free, the approach of extracting knowledge from the media is limited to the idea of paying for knowledge. This comes at hand if we wish to extract knowledge from other instances, such as pay-to-read newspapers.

The features used to predict the polarity of the papers are chosen based on previous research, as well as a combination of "best practices". However, the translation of words can be error-prone. The reason for that is the nature of the language in which the text is written in. The sentiment score can

change in the line of translation. On the other hand, the results present that this is not always the case. A possible solution is to evaluate the translation accordingly and review the text in both languages to see the differences and the validity of the translation. Another limitation arises when considering the plain financial text, such as what hegnar.no usually tend to write. Maybe it is necessary to review other papers that are known to write in a different style and compare them exclusively. A possible future applications is to better understand the media and how they write about a company to support other companies in understanding the popularity of what the media tend to write about. Such a feature could provide new knowledge for other corporations as well as to see the bias a newspaper has towards a corporation, and instead of only written about the big corporations, give light to the smaller one.

A Appendix

A.1 Crontab Command

```
ubuntu@news-data-machine */10 * * * * /usr/local/bin/python3
/home/ubuntu/Programs/rss_feed_crawler >> crontab.log 2>&1
```

A.2 RSS Crawler

```
import requests, os, traceback
import xml.etree.ElementTree as ET
from datetime import datetime as dt
import logging as log

papers = [
    {
        'name': 'hagnar',
        'rss_url': 'https://www.hegnar.no/rss/feed/all',
        'params': {}
    },


]

cwd = os.path.dirname(os.path.realpath(__file__))
log.basicConfig(filename=os.path.join(cwd, 'error.log'),
    level=log.WARNING, datefmt='%Y-%m-%dT%H:%M:%S',
                    format='[%(asctime)s] %(levelname)-s: %(
    message)s')

for paper in papers:
    try:
        now = dt.now()
        paper_dir = os.path.join(cwd, "data/" + paper['name'
    ])
        last_feed = os.path.join(paper_dir, "last_feed.xml")

        new_feed = requests.get(paper['rss_url'], params=
    paper['params'])

        previous_guids = []

        if os.path.isfile(last_feed):
```

```
            last_tree = ET.parse(last_feed)
            items = last_tree.getroot()[0].findall('item')
            for child in items:
                previous_guids.append(child[4].text)
            last_tree.write(last_feed)

            root = ET.fromstring(new_feed.content)
            tree = ET.ElementTree(root)
            for item in root[0].findall("item"):
                guid = item[4].text
                if guid in previous_guids:
                    root[0].remove(item)
            os.makedirs(paper_dir, exist_ok=True)
            tree.write(os.path.join(paper_dir, now.strftime("%Y-%m-%d_%H:%M:%S") + ".xml"))
            root = ET.fromstring(new_feed.text)
            ET.ElementTree(root).write(last_feed)
        else:
            root = ET.fromstring(new_feed.content)
            os.makedirs(os.path.dirname(last_feed), exist_ok=True)
            ET.ElementTree(root).write(last_feed)
    except Exception as e:
        trace = traceback.format_exc()
        log.error(trace)
        print(paper)
```

A.3 CSV Generator of Companies

```

import os
import xml.etree.ElementTree as et
import pandas as pd
from datetime import datetime

"""
This Script creates a CSV file from XML files in a directory.
"""

path = "/Users/felipesepulveda/PycharmProjects/norwegian_news/data/hegnar"


def getvalueofnode(node):
```

```
    """ return node text or None """
    return node.text if node is not None else None


def parse_XML(xml_file):
    dfcols = ['pubdate', 'title', 'description', 'link', 'paper', 'subscription']
    df_xml = pd.DataFrame(columns=dfcols)
    namespace = {
        'subscription': "http://www.api.no/rss/"
    }

    for filename in os.listdir(xml_file):
        if not filename.endswith('.xml'):
            continue
        fullname = os.path.join(xml_file, filename)
        parse_xml = et.parse(fullname)
        xroot = parse_xml.getroot()

        for node in xroot:
            paper = node.find('title')
            for channel in node.findall('item'):
                pubdate = channel.find('pubDate')
                title = channel.find('title')
                description = channel.find('description')
                link = channel.find('link')
                subscription = channel.find('subscription:subscription', namespace)

                df_xml = df_xml.append(
                    pd.Series([getvalueofnode(pubdate), getvalueofnode(title), getvalueofnode(description),
                               getvalueofnode(link), getvalueofnode(paper), getvalueofnode(subscription)], index=dfcols)
                    , ignore_index=True)
                df_xml['pubdate'] = df_xml['pubdate'].apply(lambda x: datetime.strptime(x, '%a, %d %b %Y %H:%M:%S %z'))

                df_xml.to_csv("hegnar.csv")


if __name__ == '__main__':
    parse_XML(path)
```

A.4 Time Converter

```
from datetime import datetime
import pandas as pd
import os
import numpy as np

cwd = os.getcwd()

path = os.path.join(cwd, "norwegian_csv_data", "hegnar.csv")

df = pd.read_csv(path, encoding="utf-8")

try:
    df['date2'] = df['pubdate']

    df['date2'] = df['date2'].apply(lambda x: datetime.
   strptime(x, '%a, %d %b %Y %H:%M:%S %z'))

    df['date'] = df['date2'].dt.strftime('%Y-%m-%d')

except ValueError as e:
    print(e)
except TypeError as e:
    print(e)

df.to_csv('pubdate.csv')
```

A.5 HTML Crawler

```
import requests
from bs4 import BeautifulSoup
import pandas as pd
import os
import copy
import time

start = time.time()

cwd = os.getcwd()

path = os.path.join(cwd, "norwegian_csv_data", "hegnar.csv")

df = pd.read_csv(path, encoding="utf-8")
```

```
15
16 new_data = []
17 for index, row in df.iterrows():
18     # REINITIALIZE THE API
19     urls = row['link']
20     newrow = copy.deepcopy(row)
21     try:
22         # translate the 'text' column
23         request = requests.get(urls).text
24         soup = BeautifulSoup(request, 'html.parser')
25
26         # Removes all <div> with id "sponsor-slug"
27         for child_div in soup.find_all('figcaption', attrs={'
    class': "article__image-text"}):
28             child_div.decompose()
29
30         # Removes all <a> tags an keeps the content if any
31         a_remove = soup.find_all("stong")
32         for unwanted_tag in a_remove:
33             unwanted_tag.replaceWithChildren()
34
35         # Removes all <a> tags an keeps the content if any
36         a_remove = soup.find_all("a")
37         for unwanted_tag in a_remove:
38             unwanted_tag.replaceWithChildren()
39
40         # Remove all <p> with class "copyright"
41         for child_p in soup.find_all('footer', attrs={'class'
    : "site-footer"}):
42             child_p.decompose()
43
44         # Remove all <p> with class "copyright"
45         for child_p in soup.find_all('div', attrs={'class': "
    column site-footer__column"}):
46             child_p.decompose()
47
48         # Remove all <p> with class "copyright"
49         for child_p in soup.find_all('div', attrs={'class': "
    row site-footer__row row__threes additional-info"}):
50             child_p.decompose()
51
52         p = soup.find_all('p')
53         p = [p.text for p in soup('p')]
54         p = [' '.join(p)]
55         p = [el.replace('\xa0',' ') for el in p]
```

```
            p = str(p).strip('[]').strip('\'')

            newrow['new_text'] = p

        except Exception as e:
            print(str(e))
            continue
        new_data.append(newrow)

df = pd.DataFrame(new_data)

df.to_csv("all_text.csv", index=True, encoding="utf-8-sig")
```

A.6 Company Extraction

```
import os
import pandas as pd
import numpy as np
import re
import re

cwd = os.getcwd()

news = os.path.dirname(cwd)

news = os.path.join(news, "hegnar.csv")

df = pd.read_csv(news)

filtered_sentence = []

df['text'] = df['text'].str.strip("\"")

df['lowercase'] = df['text'].str.lower()

df = df[df['text'].str.contains(Company Name, na=False, case=
    True)]

df['mentioned'] = df.text.str.count(Company Name)

df = df[df.mentioned != 0]

df['total_words'] = df.text.str.count(' ') + 1

```

```
df['count'] = df['text'].str.len()

df.to_csv(file_name, index_label=True, index=True)
```

A.7 Translation of Text

```
from googletrans import Translator
import pandas as pd
import os
import copy

cwd = os.getcwd()

path = os.path.join(cwd, "a.csv")

df = pd.read_csv(path, encoding="utf-8")

translatedList = []
for index, row in df.iterrows():
    # REINITIALIZE THE API
    translator = Translator()
    newrow = copy.deepcopy(row)
    try:
        # translate the 'text' column
        translated = translator.translate(row['text'], src='
   no', dest='en')
        newrow['english'] = translated.text
        translatedList.append(newrow)
    except Exception as e:
        print(str(e))
        continue

df = pd.DataFrame(translatedList)
```

A.8 Adding Sentiment Polarity to Text

```
import pandas as pd
import re
from nltk.sentiment import SentimentIntensityAnalyzer
from nltk.corpus import stopwords


path = "path to csv"

df = pd.read_csv(path, encoding="utf-8")

stop_words = set(stopwords.words("english"))

sia = SentimentIntensityAnalyzer()

# Remove punctuation
df['paper_text_processed'] = df['english'].apply(lambda x: re
    .sub('[,\.!?]', '', x))

df['paper_text_processed'] = df['paper_text_processed'].apply
    (lambda x: ' '.join([word for word in x.split() if word
                         not in (stop_words)]))  # Removes
    StopWords


def sentiment(x):
    sentiment = sia.polarity_scores(x)['compound']
    return sentiment


df['sentiment'] = df['paper_text_processed'].apply(sentiment)
```

References

Alistair, K. and Diana, I. (2005). Sentiment Classification of Movie and Product Reviews Using Contextual Valence Shifters. *Proceedings of FINEXIN*, 22:2006.

Alomari, K. M., Elsherif, H. M., and Shaalan, K. (2017). Arabic tweets sentimental analysis using machine learning. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*.

Amolik, A., Jivane, N., Bhandari, M., and Venkatesan, D. M. (2016). Twitter sentiment analysis of movie reviews using machine learning technique. *International Journal of Engineering and Technology*, 7(6):2038–2044.

Andranik Tumasjan, Timm O. Sprenger, P. G. S. I. M. W. (2005). Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment. *Journal of Biological Chemistry*, 280(39):33411–33418.

Azevedo, A. I. R. L. and Santos, M. F. (2008). Kdd, semma and crisp-dm: a parallel overview. *IADS-DM*.

Bharti, S. K., Babu, K. S., and Jena, S. K. (2015). Parsing-based Sarcasm Sentiment Recognition in Twitter Data. *2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 1373–1380.

Bollen, J., Mao, H., and Pepe, A. (2011a). Modeling Public Mood and Emotion : Twitter Sentiment and Socio-Economic Phenomena. *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, pages 450–453.

Bollen, J., Mao, H., and Zeng, X. (2011b). Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8.

Cios, K. J., Pedrycz, W., Swiniarski, R. W., and Kurgan, L. (2007). The Knowledge Discovery Process. *Data Mining - A Knowledge Discovery Approach*, pages 9–22.

Das, T. K., Acharjya, D. P., and Patra, M. R. (2014). Opinion mining about a product by analyzing public tweets in Twitter. *2014 International*

Conference on Computer Communication and Informatics: Ushering in Technologies of Tomorrow, Today, ICCCI 2014, pages 3–6.

Davidov, D., Tsur, O., and Rappoport, A. (2010). Enhanced sentiment learning using twitter hashtags and smileys. In *Proceedings of the 23rd international conference on computational linguistics: posters*, pages 241–249. Association for Computational Linguistics.

Fang, F., Datta, A., and Dutta, K. (2012). A Hybrid Method for Cross-domain Sentiment Classification Using Multiple Sources. *Icis*, pages 1–14.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, 17(3):37.

Felbo, B., Mislove, A., Søgaard, A., Rahwan, I., and Lehmann, S. (2017). Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. *arXiv preprint arXiv:1708.00524*.

Fong, S., Zhuang, Y., Li, J., and Khoury, R. (2013). Sentiment analysis of online news using MALLET. *Proceedings - 2013 International Symposium on Computational and Business Intelligence, ISCBI 2013*, pages 301–304.

Hammer, H., Bai, A., Yazidi, A., and Engelstad, P. (2014). Building sentiment Lexicons applying graph theory on information from three Norwegian thesauruses. *Norsk Informatikkonferanse (....*

Jena, S., Babu, K., Vachha, B., Bharti, S., and Pradhan, R. (2016). Sarcastic sentiment detection in tweets streamed in real time: a big data approach. *Digital Communications and Networks*, 2(3):108–121.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer.

Kim, S.-M. and Hovy, E. (2007). Crystal: Analyzing predictive opinions on the web. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 1056–1064.

Kiritchenko, S., Zhu, X., and Mohammad, S. M. (2014). Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research*, 50:723–762.

Korkontzelos, I., Nikfarjam, A., Shardlow, M., Sarker, A., Ananiadou, S., and Gonzalez, G. H. (2016). Analysis of the effect of sentiment analysis on extracting adverse drug reactions from tweets and forum posts. *Journal of Biomedical Informatics*, 62:148–158.

Lau, R. Y., Lai, C. C., Ma, J., and Li, Y. (2009). Automatic Domain Ontology Extraction for Context-Sensitive Opinion Mining. *Icis*, pages 35–53.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Naaman, M., Boase, J., and Lai, C.-H. (2010). Is it really about me?: message content in social awareness streams. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, pages 189–192. ACM.

Nasukawa T, Yi, J. (2003). Sentiment Analysis: Capturing Favorability Using Natural Language Processing. *Biological Reviews*, 31(1):1–29.

Nielsen, F. A. (2011). A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. *CEUR Workshop Proceedings*, 718:93–98.

Nirmal, V. J. and Amalarethinam, D. I. G. (2015). Parallel Implementation of Big Data Pre-Processing Algorithms for Sentiment Analysis of Social Networking Data. *Intern. J. Fuzzy Mathematical Archive*, 6(2):149–159.

O'Shea, M. and Levene, M. (2011). Mining and visualising information from RSS feeds: A case study. *International Journal of Web Information Systems*, 7(2):105–129.

Öztürk, N. and Ayvaz, S. (2018). Sentiment analysis on Twitter: A text mining approach to the Syrian refugee crisis. *Telematics and Informatics*, 35(1):136–147.

Pang, B. and Lee, L. (2006). Opinion Mining and Sentiment Analysis. *Foundations and Trends® in InformatioPang, B., & Lee, L. (2006). Opinion Mining and Sentiment Analysis. Foundations and Trends® in Information Retrieval, 1(2), 91–231. doi:10.1561/1500000001n Retrieval*, 1(2):91–231.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? Sentiment Classification using Machine Learning Techniques. *Antike und Abendland*, 57(July):79–86.

Preece, J., Rogers, Y., and Sharp, H. (2015). *Interaction design : beyond human-computer interaction*. John Wiley & Sons, Chichester W. Sussex, 4th editio edition.

Raina, P. (2013). Sentiment analysis in news articles using sentic computing. *Proceedings - IEEE 13th International Conference on Data Mining Workshops, ICDMW 2013*, pages 959–962.

Rogalewicz, M. and Sika, R. (2016). Methodologies of knowledge discovery from data and data mining methods in mechanical engineering. *Management and Production Engineering Review*, 7(4):97–108.

Rohanizadeh, S. S. and Bameni, M. M. (2009). A proposed data mining methodology and its application to industrial procedures.

Soelistio, Y. E. and Surendra, M. R. S. (2015). Simple text mining for sentiment analysis of political figure using naive bayes classifier method. *arXiv preprint arXiv:1508.05163*.

Swati, U., Pranali, C., and Pragati, S. (2015). Sentiment Analysis of News Articles Using Machine Learning Approach. In *International Journal of Advances in Electronics and Computer Science*, number 2, pages 2393–2835.

Thelwall, M. and Prabowo, R. (2009). Sentiment analysis: A combined approach. *Journal of Informetrics*, 3(2):143–157.

Turing, A. M. (2009). Computing machinery and intelligence. In *Parsing the Turing Test*, pages 23–65. Springer.

Walker, M. A., Anand, P., Tree, J. E. F., Abbott, R., and King, J. (2012). A Corpus for Research on Deliberation and Debate. *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 812–817.

Wanner, F., Rohrdantz, C., Mansmann, F., Oelke, D., and Keim, D. A. (2009). Visual sentiment analysis of RSS news feeds featuring the US presidential election in 2008. *CEUR Workshop Proceedings*, 443(February).