Erasmus University Rotterdam
Erasmus School of Economics

Master's Thesis Data Science and Marketing Analytics

Extracting Customer Feedback from Social Media using Text Mining and Sentiment Analysis in French

Author: Andrei Pascanean
Student ID number: 451714

Supervisor: Dr. Phyllis Wan
Co-Reader: Dr. Vardan Avagyan

24th of July 2020

Preface

The views stated in this thesis are those of the author and not necessarily those of the supervisor, second assessor, Erasmus School of Economics or Erasmus University Rotterdam.

Table of Contents

Executive Summary

Obtaining reliable customer feedback, for the purpose of strategic decision making, has posed challenges for managers. Large costs and time restrictions, combined with biases in involuntary feedback, have dampened the effectiveness of traditional survey methods. With the growth of online social media platforms, customers have become accustomed to stating their opinion online. In this paper, we propose a method for customer feedback collection and analysis using online social media platforms as a reliable data source. We propose four different methods to extract sentiment insights from tweets, applying supervised machine learning algorithms. Besides customer sentiment, we also show how managers can extract key phrases and locations from user-generated tweets. We focus on the French-speaking telecom market in Canada.

For the collection of data, we propose the use of emojis as search queries in addition to emoticons. This allows for a dataset of 200,000 observations to be collected in a short time frame. Our methods comprise a Naïve Bayes, Maximum Entropy and SVM model, as well as a Lexicon-based approach for determining sentiment. We apply a pre-trained NER and POS tagger to obtain insights on the contents of tweets. An evaluation of these methods shows that the Lexicon-based method performs best at an accuracy of 77.50%, with an SVM model following close with an accuracy of 72.50%. The Naïve Bayes and Maximum Entropy classifiers have an accuracy of 57.50% and 60.00%, respectively. For the detection of location in tweets, we show that it is possible to extract a location from tweets who's authors have provided no embedded co-ordinates. Our method has a recall rate of 70.59% when detecting locations in tweets. Lastly, we also show a method for managers to extract keywords from tweets that can provide insights to what product or service users are focusing on.

We have provided a tangible application of social media sentiment analysis in French, showing that common machine learning methods are applicable in different languages. The methods described in this paper can be implemented as part of a social media monitoring platform, possibly in a SOC or NOC team. Our framework can be applied to live monitoring for specific market segments, allowing managers to integrate social media insights into their decision making process.

1. Introduction

1.1 Importance of Customer Feedback

Consumer-facing firms always strive to satisfy their customers as much as possible. Because the customer eventually is the one who drives sales, keeping them satisfied is a big priority. In order to do this, managers must have a reliable source of customer feedback, whether this feedback is good or bad. One way to find this feedback, is to explicitly ask for it in the form of product and service evaluations. However, these have proven to be unreliable and even downright biased. As shown by Ofir and Simonson (2001), customers who are expected to leave an evaluation, oftentimes are biased towards a more negative review. This effect is regardless of actual product quality, which puts into question how much value can be obtained from explicit requests for feedback. Besides being insincere, customer feedback collection in this way could also prove to be costly. Once this feedback is obtained however, there is also the issue of extracting important firm-related information. When it comes to customer satisfaction, the sentiment of the feedback is important, as it provides managers with an indication of their product's performance. A customer's sentiment gives an indication of how a firm should react, possibly by improving their offerings. Kumar and Pansari (2016) have shown that customer engagement is an important factor in firm performance, especially when it comes to service-oriented companies. Gaining this competitive advantage only requires a consistent and bilateral engagement with a firm's customers. While this engagement is already present at in-person brand experiences, managers should also follow this approach online.

While the sentiment of a customer's feedback is important to managers, the content itself also remains interesting. Features such as the product or service that a customer mentions, or their location, are useful in deciding the next steps to take. By knowing what exactly customers are complaining about, managers can localize the problem to a certain product range or service area. As an example, a customer could be having issues with their new car, thus giving a negative sentiment in their feedback. However, the specific experience of this customer could be a broken windshield wiper or a failing door hinge, which would require a different response from, say, an engine failing to start. The customer experience also helps managers understand the gravity of the situation and the required response. As in the example above, a broken wiper can be fixed with a

simple part replacement, while recurring engine problems could prompt a product recall. This customer feedback context could also include the customer's location. Firms could use this information for geographic market segmentation, location-targeted marketing or even decentralizing marketing, where managers responsible for market subsections can intervene independently. Customer experience can vary between large cities and rural villages, hot and cold climates and between different countries or states. That is why location can be insightful when looking at customer experience.

1.2 Research Question

Overall, this paper's research question reads as follows:

Research Question: How can insights be extracted from Twitter to help with strategic decision making?

While at first this research question may seem broad, it can be subdivided and structured to make it more coherent. First, when looking at a customer's feedback, there are two kind of insights that can be extracted; the feedback's sentiment and the feedback's content. The sentiment gives an important indication of how the customer is feeling about either a specific offering, or the firm itself. Meanwhile, the content provides further information about why exactly the customer is providing feedback. Here, we can find details about what product or service the customer used, what kind of experience he or she had and possibly where the customer is located. Using these key points, we can state two sub-questions that further subdivide our research question:

Sub Question 1: How can the sentiment of tweets be extracted?

Sub Question 2: How can the content of tweets be extracted?

While this paper will focus on applying a collection of machine learning methods to answer the first sub-question, we will also include some interesting applications of methods to answer the second sub-question. There has been previous research in the domain of sentiment analysis, with

most applications being in English. We hope to show that sentiment analysis methods are also very applicable in French. The trained models shown in this paper are flexible and can be modified to suit different languages. The distant supervision method for data collection is also not novel. However, we have updated the methodology by adding emojis to the search queries used in data collection. This makes the data collection method more in-line with modern keyboards on mobile devices. In addition to this, we also apply a pre-trained named entity recognition model to find a customer's location, where oftentimes one is not explicitly provided. Lastly, we also use a part of speech tagger to identify some interesting keywords within the tweet's content, allowing for a product, service and user experience to be extracted.

1.3 Research Scope

The research question begs another important question; why Twitter? For that matter, why a social media platform at all? With this section, we hope to address this issue, showing that Twitter, and social media in general, can offer an interesting source of sincere customer feedback. Furthermore, we will explain the context in which this paper's methods are applied and tested. While we chose to focus on the telecom sector in the Canadian French-speaking market, the methods described in this paper can be applied to a wider range of languages and markets.

1.3.1 Social Media as an Attractive Customer Feedback Alternative

As a solution to the issue of sincere and accurate feedback, we propose the use of social media platforms and user-generated content as input data. Services such as Facebook and Twitter, have given its users a platform for expressing their thoughts and feelings on various topics, with a global reach. This also includes reviews of various products, endorsements of certain services and reactions to a firm's PR decisions. Customers now have an effective and low-effort way of reaching out to businesses, with many other potential customers watching on from the side-lines. Electronic word-of-mouth marketing has been shown to be very effective on social media platforms (Rosario et al., 2016). While the methods described in this paper can be applied to any user-generated data source, the choice was made to focus on only Twitter as a social media platform. This was done because, as Schweidel and Moe (2014) explain, the platform or 'venue' that customers choose for

expressing their opinion can have an effect on the feedback's sentiment. Furthermore, Twitter imposes a limitation on the length of tweets made by its users, with 140 characters being the maximum allowed length. This leads to the content being emotionally charges and forces customers to explain their issues concisely. While this limits the amount of data that can be analysed, it also increases the density of sentiment in customer feedback.

1.3.2 Importance of Social Media Platforms

Social media platforms have become a staple of our digital diet, with the Pew Research Centre (2015) reporting a staggering 65% of adult Americans using at least one social media platform. By age group, 90% of Americans between the ages of 18 and 29 use social networking sites, with a significant 77% of those aged 30 to 49. It is hard to avoid such an important aspect of the customer's daily routine. Meanwhile, research into the workings of Twitter has shown that online user utilize the platform as a possible source of information (Java et al., 2006). Specifically, in the domain of marketing and online customer feedback, Jansen et al. (2009) have shown that Twitter is a useful online tool for word of mouth communication. The especially strong influence on public opinion that social media content can have, was shown recently by Allcott and Gentzkow (2017) in their research towards the effects of fake news on the 2016 U.S. presidential election. User generated content on social media platforms can reach a substantial amount of people, with many users trusting the opinions and experiences of others. Sites such as Twitter and Facebook have become platforms with high levels of exposure for users airing their grievances. This has made social media crucial component of any customer-facing firm's decision making process.

1.3.3 Research Application Context

Concerning the application of the methods described in this paper, the telecom industry was chosen. Specifically, a Canadian francophone telecom provider, with its users in the French parts of Canada. This choice was made due to two important factors that highlight the uniqueness of social media generated feedback. First, the nature of telecom-related complaints is often time-sensitive. Network outages, lack of mobile service and dropped calls are issues that the customer has little patience for. A lack of data connection for users in the morning cannot wait until the evening and

should be detected and addressed as soon as possible. Besides the time-sensitive nature of feedback in the telecom sector, mobile operators also rely on a network. This network connects major cities and towns, and can span across entire countries, which means that location plays an important factor. For a mobile provider, the location of a customer providing feedback can be helpful in determining potential outages that are limited to certain parts of the network.

1.4 Managerial Relevance

In this section, we will look at some of the benefits of social media sentiment analysis. More broadly, we will explain some useful applications of the insights obtained from customers on social media platforms. For the applied context of telecom providers, we will also show a unique network-related application of this paper's methods.

First, using the sentiment obtained from social media platforms, managers can map the overall sentiment of their brand on a specific market. Going more in-depth, firms can find out what the driver is behind a certain sentiment, including the likes and dislikes of a specific product or service. Managers can use the methods outlined in this paper to identify the focus of a customer's criticism, while being assured that this opinion was given voluntarily. This can then be used to understand what has to be improved in a product line-up or a service offering. Furthermore, companies can use this market sentiment to gouge their customer base's response to different advertising campaigns. While in this paper we discuss feedback in the context of a product or service, the customer's voice also concerns itself with the firm's image. Our framework provides a way to listen to the customer, a way for firms to understand what impact they have in a given market. However, the analysis we describe can extend further than only one firm. Applied in parallel, it gives companies the chance to map the sentiment of an entire collection of brands. Managers can use the tools described to get an idea of the customer's sentiment of their competitors, which could provide crucial information in a firm's positioning on the market. For example, managers can use sentiment analysis to find disgruntled sections of the competitor's user base, poaching these customers with better offerings of their own. Especially useful, is the ability to compare customer sentiment between firms, allowing managers to get an idea of market share on social media. Lastly, our sentiment analysis method can also be used to identify influential users on social media. These

are customers with large followings, who influence the opinion of those on social media platforms. Now, managers can track the sentiment of these specific users, being able to react as soon as an influencer tweets out feedback.

1.4.1 Telecom Sector Context

In the telecom sector, changes in customer satisfaction are usually caused by drops in quality of service (QoS). These drops range from small decreases in mobile data bandwidth, to an entire network outage. While telecom providers mainly focus on upholding a minimum level of coverage for all of its customers, QoS needs can differ per customer. On the user's end, the bandwidth required largely depends on the application used. As an example, sending messages over WhatsApp requires less lower data connection speeds than watching videos on YouTube. This means that even if there is no loss of coverage in the network at a given time, users may still experience a worse network performance than was promised. Telecom providers are often unaware of this fluctuation in QoS delivery, mostly because it requires real-time insights into the end-user applications used, which would infringe on user privacy. Instead, our proposed method allows providers to monitor these changes in QoS, while using reliable and voluntary customer feedback. The framework described in this paper can help improve both the reaction time to network outages, as well as provide a method to monitor brand perception and product experience.

Additional insights that can be obtained from customer feedback, is the product or service in question, as well as a possible location. Both these applications can also be used in the telecom sector, being especially useful when cross-referencing with internal data. The keywords identified through the pre-trained part of speech tags model, can be cross-referenced with a custom list of application, product and service names. This way, a customer's tweet can be pinned to a specific topic, such as being about a firm's product or a user-end service. Firms could use this information to better asses what customers are talking about on social media platforms; which products or services are under scrutiny. This same method can be used to find out more about the customer's experience, with words such as 'horrible', 'wonderful' and 'poor' describing a customer's assessment. Lastly, a customer's location can be extracted and later used to map the negative sentiment of a customer base. While this may sound straightforward, modern privacy laws have limited the amount of information customers are willing to share online. As such, most users on

Twitter are reluctant to share embed their geo-coordinates in their tweets. However, the method applied in this paper does not rely on an embedded location and instead extracts the location coordinates from place names mentioned in the tweet's text. While telecom firms can use these locations to cross-reference with their internal network topography, any firm can use customer locations to geographically map out a customer base.

1.5 Scientific Relevance

While there has been considerable work in the field of social sentiment analysis, with previous research also focusing on Twitter, this paper hopes to offer a unique angle. Specifically, there are a number of additions on previous research that are made, which together combine to give an insightful addition. First, the method of distant supervision explained by Read (2005) and applied later by famously Go et al. (2009), is expanded on in this paper. Since the publication of those papers, the keyboards used by customers online have changed. Emoticons are becoming more and more out of date, with emojis offering a much broader emotional scale. A selection of emojis has been added to the data collection method in order to offer a fast and effective way of collecting training data. This means that up to 200,000 tweets have been collected in a matter of only two weeks. After all, the data used in this methodology should be easy and quick to collect, in order to offer managers more flexibility in applying the framework to different markets.

Next, this paper expands on the limited application of sentiment analysis in French. While there has been a broad range of applications in using machine learning techniques for extracting sentiment, these have mostly been applied to English data sources. Most of these papers mentions the possibility of applying their methodologies to other languages, yet there are few that actually do. Papers such as that of Rhouati et al. (2018) and Nooralahzadeh et al. (2013), apply sentiment analysis to French tweets. However, these methods focus on applying lexicon-based methods that require a pre-labelled dictionary of positive and negative words. This paper aims to extract sentiment using models that can be trained on data in any language and from any industry.

Lastly, this paper also aims to combine the sentiment of tweets with their location. While research already exists towards the extraction of customer sentiment from tweets, this research gives a

geographic attribute to that data. This allows for an analysis that can segment the output data by location. More importantly, this paper shows that while users do not always disclose their location to social media platforms, they will include location information if they deem it necessary. This is especially true in the context of the telecom industry or any other network-oriented firm. While Song and Xia (2016) have shown the spatial aspect of twitter data, they have done their analysis using only geotagged tweets. This paper, meanwhile, manages to extract a spatial element even from tweets that do not have an embedded geolocation.

2. Relevant Work

In this section, we will be taking a look at research that is relevant to this paper. This includes underlying background information on machine learning, sentiment analysis, social media text mining and natural language processing. This section is not intended as an in-depth theoretical breakdown, and more a general overview with possibilities for further reading.

2.1 Theoretical Background

In order to gain more insight into the methods applied later in this paper, this subsection will give a brief overview of their theoretical foundations. This includes topics such as training models, analysing sentiment, mining social media platforms for data and using language as an input for statistical methods. We hope this section offers our readers a basis to work with when reading through this paper, with possible further reading options to fall back on.

2.1.1 Machine Learning

As a subset of Artificial Intelligence (AI), Machine Learning (ML) is a broad term that is used for computer algorithms that are designed for a certain application. What makes these algorithms unique, is that they combine mathematical and statistical methods to improve their performance through experience. This experience is gained by 'training' the algorithm using a training dataset that consists of a sample of all real-world observations. The goal then is, given a limited training dataset, to train a machine learning algorithm well enough for it to complete a specific task

successfully. In this definition, the terms ‘limited dataset’, ‘well enough’ and ‘successfully’ can change depending on the application. Datasets can range from merely 1,000 observations to millions, if not billions of data points. Deciding when an algorithm has trained sufficiently is also important, in order to make sure it is still applicable to real-world testing data. During its application, ‘success’ can be measured in different ways, depending on the problem at hand.

Overall, there are three forms of machine learning; supervised, unsupervised and reinforcement learning. For supervised learning, a set of inputs and corresponding outputs is given in the training data. The algorithm, or ‘model’, has to then find a way to correctly map a new set of inputs to the correct output. As an example, we could give a supervised learning model a training set of images of cats and dogs, with their corresponding ‘cat’ and ‘dog’ labels. The model would then be given a new image and would try to correctly predict whether the animal shown is a cat or a dog. Contrary to this method, unsupervised machine learning models are not given a target ‘output’ as part of training data. Extending our previous example, in the case of an unsupervised machine learning model, we would supply only the images of cats and dogs without the labels. The model would then have to find a pattern to group the images by itself. Finally, reinforcement learning refers to models that learn from their mistakes. This form of machine learning is most common in teaching models to play games, such as chess, where each successful move or game is rewarded. The model then tries to maximize this reward by making correct decisions, which improves its performance in the game. In this paper we train supervised machine learning models to obtain customer feedback sentiment. For more information about machine learning models and applied statistical methods, James et al.’s (2013) book *An introduction to statistical learning* can be helpful. For a more economics oriented book, with an intuitive explanation of the mathematical basics, Stock and Watson’s (2007) book *Introduction to Econometrics* can also be interesting.

2.1.2 Natural Language Processing

In order to use textual data in conjunction with computers, some important factors need to be taken into consideration. First, is that computers cannot read like humans, text is represented as numbers in all programs. Because of this, features that we use in language like parts of speech, word inflections, verb conjugations and sentences are unknown to machines. In order to solve this

problem, a family of morphosyntax methods have been developed. This collections of methodologies use either hand-made rules or pre-trained machine learning models, to replicate some of the features we are used to when reading. Part of speech tagging assigns labels such as 'noun', 'verb' and 'adjective' to words in sentences, while stemming and lemmatizing removes inflections and conjugations. Going further, Named Entity Recognition (NER) is a machine learning method used to identify different types of proper nouns such as names of people and places. This is oftentimes a complicated model to train, as proper nouns are not always capitalized in every language. Finally, another important process is tokenization, where a set of rules determine how to subdivide a piece of text into separate sentences and words. This can sometimes be a challenging task that is dependent on the text's context. Punctuation, special characters and spaces have to be taken into account when deciding which characters belong together in words. For further reading, see Bird et al.'s (2009) book on their Natural Language Toolkit, which serves as a collection of NLP methodologies applied in the Python programming language.

2.1.3 Sentiment Analysis

A subfield of machine learning that combines text mining methods with natural language processing (NLP), is sentiment analysis. This term refers to the process of extracting and quantifying the subjective element of text, often described as the text's tone or emotion. This supervised machine learning application oftentimes uses human-generated text full of opinions to train classification models. Examples of input data include online product reviews, survey responses, movie critic reviews and social media content. These data sources are 'mined' for large training datasets that are then used to train models in classifying inputs into either 'positive' or 'negative' observations (Pang et al., 2002). While binomial classification using two opposite classes is common, other methods have explored using a range of emotions or even a polarity scale. Such a continuous numerical scale can be obtained by using a Lexicon-based method that depends on a pre-labelled dictionary and set of rules, rather than being trained (Taboada et al., 2011). In the context of sentiment analysis, input observations are referred to as 'documents' which are part of a 'corpus' of text. Each document consists of sentences and words, which can be subdivided into 'tokens'. While most tokens are individual words, some can be punctuation symbols, emoticons and emojis.

2.1.4 Social Media Data Mining

For training supervised machine learning models, large datasets of observations are required. In our case, the training data consists of social media generated content, specifically user-generated posts on Twitter, referred to as tweets. While all platforms are web-based and can be accessed using an internet browser, traditional GUI-based (graphic user interface) scraping is often slow and inefficient. Meanwhile, collecting tweets by hand can also be a long and costly process. Thankfully, most social media platforms offer an Application Programming Interface (API), which can be used by applications to interact without the need for human operators. This means that scripts can be written in programming languages such as Python, C and Java, that communicate with Twitter's servers and request data independently. For Twitter's API, common requests are search queries, user timelines and specific tweets. Each request is returned with a collection of different parameters, each representing a different piece of information. For this paper, requesting a search query for our positive and negative labels, returns a constant livestream of unique tweets. Each tweet comes with its time and date of publication, its author's twitter handle, the tweet text itself and a possible set of coordinates. While there are many other fields included in a single output tweet, these have not been used in our research. In order to prevent an overburdening of its servers, Twitter has imposed some restrictions on its API access. All users must be pre-approved developers with their unique authentication codes, which are only granted after a screening. Furthermore, the API has a limit on the amount of requests made, with a maximum of 900 tweets every 15 minutes. While this seems like a small amount, it serves as a failsafe for search queries that are too broad and can overburden the server. This request rate is more than enough to build up an extensive dataset of 200,000 tweets, which was used in this paper. For more information on Twitter's API, the online documentation is extensive and helpful (Twitter, 2019).

2.2 Previous Research

In this subsection, we will discuss some of the existing research papers in the area of sentiment analysis and classification. The methods used in this paper are not novel, and have been applied before on either English or French text. However, the overlap of methods between the two

languages is limited, with most training-based models only having been applied in an English context. While there have been applications of sentiment analysis on French tweets before, these have mostly focused on using pre-trained models such as a Lexicon classifier. Lastly, the distant supervision method for data collection has also been used in previous research papers. However, we apply this method with added emoji labels to better adapt to modern social media typography.

2.2.1 Distant Supervision for Training Data Collection

There has already been a substantial amount of research in the domain of sentiment analysis of text. This research also extends to microblogging platforms like Twitter, with previous works covering machine learning models. Most importantly is the distant supervised method for data collection introduced by Read (2005) and later applied by Go et.al (2009). This method of using emoticons in collecting twitter training data is also used in this paper, with a further extension to the search query by adding polarizing emojis. However, unlike applications in Go et al. (2009) or Pak and Paroubek (2010), this paper did not use specific twitter accounts, such as newspapers and news outlets, to obtain objective training data. This decision was mostly made due to the disparity in tweet syntax between official PR sources and individual twitter users. Seeing as the final goal is to classify tweets of individuals, tweet components such as slang, profanity and informal writing should not be excluded from the training set.

2.2.2 Classification of English Text Data

In terms of classification methodologies, Go et al. (2009) use three different models to obtain the sentiment of tweets. Using Naive Bayes, Maximum Entropy and Support Vector Machines, they manage to reach an accuracy of around 80% in classifying tweets using unigrams. Adding bigrams as a feature set only slightly improves the accuracy of Naive Bayes and Max. Entropy, with SVM decreasing in accuracy. This decline in SVM effectiveness when adding bigrams is also seen in Pang and Lee's (2002) paper. Furthermore, Part Of Speech features are also found to not be useful, decreasing the accuracy of the SVM classifier. While SVM proves to be an effective text classification method, Pak and Paroubek (2010) found that using bigrams as features in a Naive Bayes model can also have impressive results.

2.2.3 Classification of French Text Data

In order to apply these NLP models to French-speaking data, some changes in methodology are necessary. Denecke (2009) uses a translator to analyse tweets in a lexicon-based pipeline. Here, the language is first detected, and text translated, after which three different pre-trained models are used to find a text's polarity. In this case, reviews from the German Amazon.de website were used in evaluating the method, with a low accuracy for all three methods. Of the three models applied, the lexicon-based logistic classifier performs best with an accuracy of 66%. While this method is applicable to any language that can be translated, it does not prove as effective as SVM classification. There is also a dependency on the accuracy of the translation tool in correctly conveying sentiment in the English output document.

When looking at classifying the sentiment of French texts, Ghorbel and Jacot (2011) apply an SVM model to French movie reviews. Here, they find a high accuracy of 92.5% when using a combination of unigrams and lemmatization as features. This accuracy is slightly improved when adding polarity. The accuracy does not improve much when using the polarity of POS tags, mostly due to the need to translate tags to English. Rhouati et al. (2018) took a different approach and instead used a lexicon-based classifier to achieve a precision of 70%. Here, the authors applied the CoreNLP toolkit to classify French language tweets on sentiment through their calculated polarity. They also applied an SVM classifier for comparison and achieved a precision of 77.5%.

2.2.4 Customer Feedback and Machine Learning

In a marketing context, there has been research focused towards using online, user-generated content through machine learning methods. Online customer feedback can be used to get an idea of brand or product image, or even to segment the market. Gamon (2004) uses an SVM model to fin the sentiment of user-generated reviews online. The training data used here is a global support services survey and as such is more restricted in representing the true population. This survey also is not fully voluntary, as users may have been asked to fill in their opinion. Nonetheless, Gamon achieves an impressive 85.47% accuracy when comparing users who self-reported to be 'not satisfied' to 'very satisfied'. However, the applicability of this method is limited, as all survey results come with a satisfaction score.

When it comes to product characteristics and user experience, Lee and Bradlow (2011) have used customer reviews of photo cameras map the consumer market. This was done by using K-means clustering to group user reviews by attributes they appreciated and disliked. This selection of attributes, however, was based on the user-given rating, which is not easily available for products in other industries.

3. Data

3.1 Twitter as a Data Source

For the detection of network outages through sentiment analysis, this paper looks at live Twitter data, which offers several advantages. Frequent posts allow for a small time frame such as hours or days to be chosen, while the tweets themselves are emotionally charged. Tweets are usually posted with little premeditation and can offer a more direct form of feedback compared to other forms of customer opinion. Lastly, there are also some challenges such as slang, character limits and hyperlink referrals.

Twitter is a platform where posts are generated frequently on a daily basis, with 152 million daily active users (SEC & Twitter, 2019) and more than 500 million tweets per day (Stricker, 2014). This averages at more than three tweets a day per user, which comes in handy when using Twitter as a live sentiment measurement. Not only do users tweet often, they also do it voluntarily, thereby avoiding the annoyance of survey requests or customer satisfaction phone calls. Such a constant and frequent generation of observations makes it easy and quick to collect a sufficiently large dataset for training models, while also offering a constant stream of testing data. The twitter platform is also available and frequently used in most countries, most often in a country's native language. This again guarantees a source of reliable and easy to collect training data. Besides their frequency, tweets usually have little to no prior planning or formatting. This lack of premeditation when writing a tweet is motivated both by the live nature of the platform, as well as the character limit of 140 characters that it imposes on posts. A consequence of this is that tweets are usually emotionally charged and contain various indicators of emotion such as emoticons, emojis and

expletives. These are especially useful in conveying a feeling or sentiment with limited words. Even though tweets contain some challenging aspects, such as internet slang and information embedded in hyperlink referrals, the data extracted is still useful and precise. Its timely and emotion-clad nature makes Twitter a good source of data for live sentiment analysis.

3.2 Data Collection

For the collection of a training dataset, Twitter's own API was used. This interface is free for development and research purposes and allows for users to crawl twitter at a maximum rate of 900 tweets every 15 minutes. If we assume the daily rate of 500 million tweets, resulting in about 5.2 million tweets every 15 minutes, the imposed limit represents under 0.02% of Twitter's entire live throughput. This hard cap on the total stream of tweets, also nicknamed the 'firehose', will not be an issue for this paper's scope in application. Restrictions on tweets for specific characters and language will greatly reduce the number of tweets filtered down to the final training dataset. These restrictions can be imposed within the Twitter Streaming API as *search query* and *tweet language.* The language was chosen as French for an initial analysis of tweets targeted at a francophone telecom provider, with creative use of the search query to obtain training datasets and live tweets. The training dataset must be labelled in accordance with the tweet's polarity and would require hand-labelling. To mitigate this costly endeavour, I have instead chosen to use distant supervision in the training of my machine learning models. This implies the use of emoticons as proxies for either negatively or positively labelled tweets, as was first introduced by Read (2005). This method was also notably used in the work of Go et al. (2009), where a limited amount of 5 positive and 3 negative emoticons were used as query terms for the Twitter API. In my research I have further extended this list of emoticons to further capture a broader range of positive and negative emotions, as done by Bird et al. (2009) in their twitter sample corpus. A sample of emoticons used can be seen in Table 3.1, with a complete list available in Table 7.1.

Table 3.1: Sample of used as search queries for training data collection

Positive Emoticons	Negative Emoticons
:)	:(
;)	:-(
XD	=/
:D	:-[

However, the online landscape is ever-changing, and users have different typing habits. Great strides have been made in text communication, especially in the domain of emotion-clad keyboards. That is why I have also added a list of emojis, representing both positive and negative sentiment, to the list of Twitter API search queries. These were selected on the basis of Novak et al.'s (2015) research into the sentiment behind emojis. From the data collected by them, the emojis with the highest frequency as either positive or negative were chosen, above the threshold of at least 100 appearances. The chosen emojis can be seen in Table 3.2.

Table 3.2 List of emojis used as search queries for training data collection

Positive Emojis		Negative Emojis	
😙	🎉	😩	😣
🎁	🏆	😒	😐
♥	💙	😡	😿
🎊	😚	😐	👎
💞	😍	😑	
🐓	😘	😠	

A combination of emoticons and emojis was used in search queries for the training set generation. Over the span of 2 weeks, a dataset of 200,000 tweets was collected with a balanced 50-50 split between positive and negative labels. While this method is effective in collecting a sizable training set in a relatively short time, it does have limitations. Users can use emojis and emoticons in jest when writing irony and sarcasm, something that cannot be easily picked up. These tweets will be wrongly labelled and automatically misclassified. However, while these noisy labels of emojis and emoticons somewhat deteriorate the training data, its effect is limited when the dataset size is large. Due to the law of large numbers and the more common occurrence of non-ironic emoji use, the effect of wrongly labelled data will be minimal in large datasets. It is important to note that this effect cannot be removed entirely, neither can it be measured accurately without manually labelling data.

3.3 Datasets

This section will provide an overview of the datasets collected, both for testing and training. Using emojis as labels, we have manged to collect a substantial training dataset of 200,000 tweets in a relatively short timeframe. The testing dataset, however, is much smaller at only 71 tweets and represents two weeks' worth of customer-generated feedback. This testing dataset is comprised of tweets specifically targeted at the telecom provider used in our application.

3.3.1 Training Data

The collected dataset of 200,000 tweets contains the features *date*, *user*, *text* and *label*, with the rows representing observations of individual tweets (see Table 3.3). All tweets collected are in French, as that is the target language of the eventual telecom use-case application. The tweets were collected from the 24th of May up until the 2nd of June, with this short time period allowing for a sizable training set to be built. Both classes are balanced with the raw data being cut off at a maximum of 100,000 positive and negative tweets each. Usernames are all publicly available on the Twitter platform and user-chosen, with only standard Unicode characters used in encoding. The tweet text extracted is also in Unicode form and can contain miscellaneous character combinations such as links, emojis, mentions and hashtags. Any audio-visual elements of tweets are not included,

as this paper is limited to the textual analysis of sentiment. Lastly, all tweets are labelled with a corresponding class, with level *"0"* belonging to negative tweets and level *"1"* to positive tweets. A binomial classification was chosen instead of a multiclass method due to the difficulty in choosing a noisy label that would be an accurate enough proxy for a *'neutral'* class. Lastly, the collected data is split using a ratio of 80-20, with 80% of the data used in training the machine learning models and 20% used as a validation set in training hyperparameters.

Table 3.3: Description of features in the training data with a corresponding sample

Feature	Variable	Sample
date	Date and time of tweet being posted.	2020/05/24 12:20:24
user	Username as seen on Twitter.	lisa_etienn
text	Tweet textual content as seen on Twitter.	@lou_pscl Super merci :)
label	Tweet sentiment noisy label.	1

3.3.2 Testing Data

Because the goal of this paper is to find insights from firm-related customer tweets, a more specific testing dataset will be required. This dataset should contain tweets that have been posted by customers of a specific firm in a specific industry and intended as product or service feedback. While it is difficult to narrow down the Twitter feed to this level of precision, there are some restrictions that can be put in place to filter out raw data. First off, in order to prevent long waiting times for collecting a testing dataset, the search function of Twitter's API will be used. This means that tweets in the past seven days can be searched, with specific query terms in mind. For this paper, the query term chosen was that of the Canadian telecom provider's company name. Specifically, the brand name that consumers interact with. It is important to make the distinction from a potential overarching mother company and daughter company's branding when choosing this query term. When providing feedback on Twitter, consumers can choose to '@' mention the

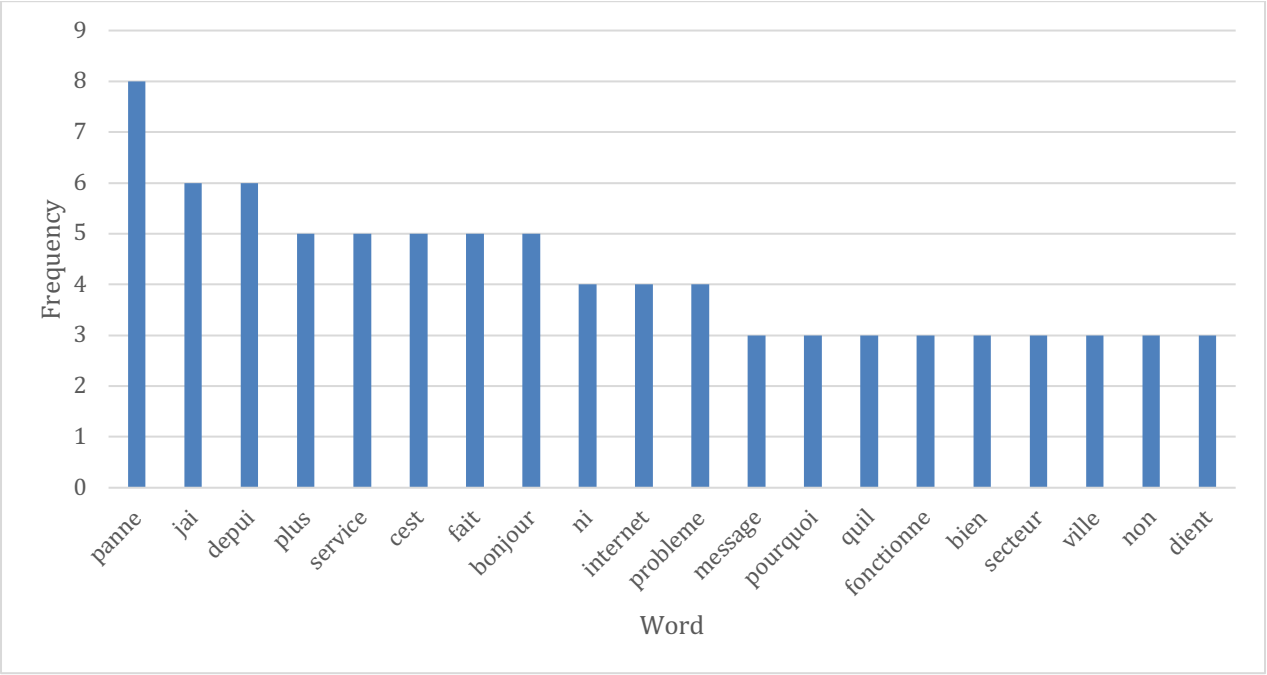
target brand when describing their experience. Besides that, consumers can also mention brand names in passing, as plain text in the tweet's body. Both these kinds of tweets will show up and will be included in the test set. The collected dataset ranges between the dates of 20th of May and 1st of June, with 71 observations found in total. These tweets are representative of about two weeks' worth of tweets aimed at the target telecom provider. All tweets found by the Twitter API's search function have been included in the final test set, with no pre-emptive selection bias displayed. These tweets include the same variables as those found in the training data, with the sentiment label for each tweet being assigned manually. The text in these tweets vary from those in the training dataset as these queried tweets are aimed at the telecom provider. A sample of the testing dataset can be seen in Table 3.4. This testing dataset was used to find the effectiveness of the methodologies used, not to be confused with the validation set established earlier that was used to tune models.

Table 3.4: Description of features in the testing data with a corresponding sample

Feature	Variable	Sample
date	Date and time of tweet being posted.	5/28/2020 0:01:16
user	Username as seen on Twitter.	bouchardmichel
text	Tweet textual content as seen on Twitter.	**@TelecomProvider** pas de télé ni internet à Prévost. Panne?
label	Manually labelled tweet sentiment	0

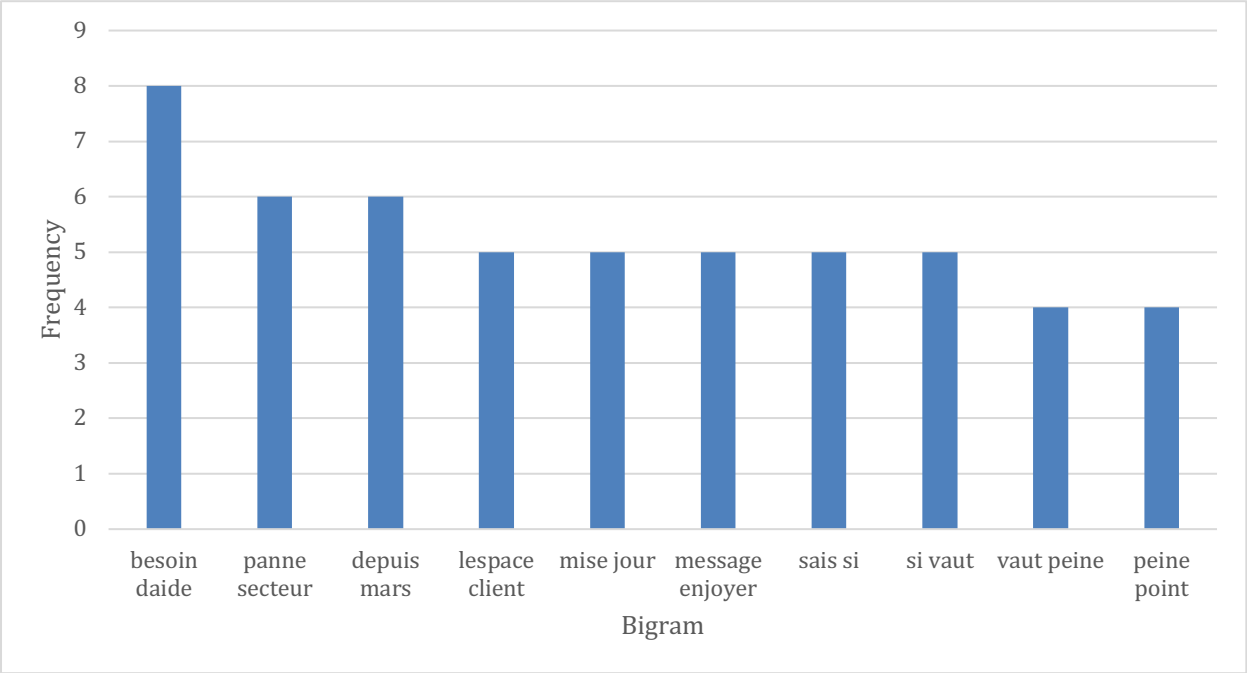
Figure 3.1 shows the 20 most frequent words used in the dataset. Notably, there are some interesting words such as 'panne' (outage), 'service', 'internet' and 'probleme' (problem) that provide some insight into the theme of these tweets.

Figure 3.1: Frequently occurring words in the testing dataset.



The 10 most frequently occurring bigrams are shown in figure 3.2, with some interesting combinations such as ‘besoin daide’ (need help), ‘panne secteur’ (area outage) and ‘depuis mars’ (since March).

Figure 3.2: Frequently occurring bigrams in the testing dataset.



3.5 Data Cleaning

After collecting raw Twitter data, a thorough data pre-processing step is needed in order to prepare the text for sentiment analysis. This is done in order to remove elements that are irrelevant to the classification process, as some words and characters hold little emotive value. The set of steps followed in cleaning the dataset were applied to each observation independently, with the final cleaned data still pertaining to its correct label. First, patterns for URLs, hashtags, mentions, reserved words, emojis, smileys and numbers were identified and removed. Each pattern has its own reason for being removed. URLs represent hyperlinks to other online websites, with all being converted to Twitter's proprietary link service, http://t.co. Together with *'@'* mentions of other users and *'#'* hashtag topics, they represent character combinations that are frequently found in all tweets, irrespective of sentiment. Next, reserved words such as *'RT'* and *'FAV'*, retweet and favourite, were removed as they pertain more to Twitter's platform-specific lingo rather than emotional human interaction. These character combinations, together with numbers, are also found frequently in tweets, irrespective of sentiment. Emojis and smileys were also removed, in order to

counteract the selection bias in the collection of tweets for model training. If these were to be left in the dataset, they would have a high correlation with the dependent variable. This would represent a simultaneity bias, where a linguistic independent feature would have a causal effect on the tweet's sentiment, while it in turn would affect the type of emoji or emoticon in the tweet's text. By removing emojis and emoticons, an attempt is made in minimizing any possible bias caused by the method of data collection. At the same time, this handicaps our model's ability to use emojis and emoticon features in analysing new tweets.

It is important to note at this point that further cleaning beyond twitter-specific patterns was only done for methods that use tokenized data for training. Lexicon-based methods did not use further cleaned text data, due to the non-binary polarity being influenced by items such as punctuation and stop words. Once these basic twitter-specific patterns were removed, the next step was to remove any punctuation from the pre-cleaned text. This included characters such as full stops, exclamation marks, colons and currency-specific signs. These characters would substantially increase the number of features used in subsequent models, without adding any sentiment-specific value. To further reduce the dimensionality of our final feature set, common stop words were removed. These are language-specific words that have little sentimental value and are only useful in a grammatical context. These stop words were retrieved from Bird et al.'s (2009) Natural Language Toolkit French corpus collection. However, in order to improve further analysis of the specifically collected training data, I also added words with high frequency and low discriminating value to the list of stop words. These additional words are related to the topics discussed in French tweets in the collection timeframe. Some are examples of online francophone slang, often used on social media platforms as abbreviations for non-emotive words. After the initial text cleaning, each tweet was tokenized. This process involves separating a document of text, usually a single observation, into 'tokens' of separate words. Now the tweets that comprise the training data can be manipulated on a per-word basis instead of only per-sentence or document. Some of the models applied later in this paper generate their features from individual words. However, many words can take on the same meaning but have a different inflection or conjugation. This would result in many variations of the same feature, without adding the benefit of discriminatory power. In order to curtail this issue, a French language stemmer was applied to shorten words to their core meaning, an example of this is shown in Table 3.5.

Table 3.5: An example of French word stemming using inflections of 'malade' (sick)

Original Word	Word Stem
malade	malad
malades	malad
maladie	malad
maladies	malad

4. Methodology

In this section, the methodologies that have been applied to extract insights from Twitter data will be explained. The models included in this section can be split up into two separate sections; models used to find a tweet's sentiment and models used in finding a tweet's location and keywords. Each section carefully explains the intricate details behind the models applied and specifically how they relate to the application in this paper. The sentiment classification models are Naïve Bayes, Lexicon-based, Maximum Entropy and Support Vector Machines. Meanwhile, the models used in finding keywords and locations in tweets are Part of Speech tagging and Named Entity Recognition.

4.1 Sentiment Classification Models

4.1.1 Naive Bayes

The Naive Bayes model is a good and simple starting point for classifying text. In this paper, a binomial Naive Bayes model is used, with labels for negative and positive tweets (0 and 1 respectively). This methodology uses the Bayes' rule to calculate the probability of a document belonging to a certain class. For a given document d, the class c^* is assigned as follows:

$$c^* = \text{argmax}_{c_0,c_1} \frac{P(d|c_i)P(c_i)}{P(d)}$$

Here, the posterior probabilities of both classes c_0 and c_1 are calculated, with the largest of the two chosen as predicted class c^*. In order to do this, both the class-conditional probability $P(d|c_i)$ as well as the prior probability $P(c_i)$ of class c_i are required. The evidence $P(d)$ however, is not required. This is because it represents a constant when comparing the posterior probabilities of two classes, as is done in this paper:

$$c^* = argmax_{c_0,c_1} P(d|c_i)P(c_i)$$

In order to predict the document's class, the Naïve Bayes model makes two important assumptions. First, it assumes that observations are independent and identically distributed, which means that the order that observations appear in is irrelevant. In practice, this means that our tweets are all independent from one another, and that a certain tweet does not influence the next one. An example of this could be a coin toss not being affected by the result of the preceding toss, and in turn not affecting the following coin toss either. Besides this, a second assumption that makes this model 'naive', is that of conditional independence. This means that all features used to calculate the probability of a certain class are independent from one another. In this paper's application, this can be interpreted as the words that belong to a certain class in the training data being independent of one another. When looking at a document, the probability of the word 'cream' appearing in the tweet should be independent of the word 'ice' appearing. In practice, both these assumptions are of course often violated. As an example, the tweet character limit of 140 often is not enough to express yourself, which is why users often tweet out concurrent tweets that belong in a certain order. Even though these issues persist in most text analysis problems, Naïve Bayes classifiers still manage to perform at an acceptable rate.

For calculating the conditional probability $P(d|c_i)$ of a document, the Naïve Bayes model uses a multi-variate Bernoulli model. First, all words used in documents of a given class c_i are placed in a vocabulary vector $\mathbf{v}$ of length l. This means that the t^{th} word in a document is represented by word w_t in the vocabulary. A feature vector $\mathbf{b}$ is generated for document $\boldsymbol{d}$, with the t^{th} element b_t being either a 0 or 1, depending on if the word w_t is present. So then $P(w_t|c_i)$ represents the probability of a word w_t being used in a tweet of class i. With the previous assumptions, we can then write the conditional probability as follows:

$$P(d|c_i) = P(\mathbf{b}|c_i) = \prod_{t=1}^{l}\left[b_t P(w_t|c_i) + (1 - b_t)\left(1 - P(w_t|c_i)\right)\right]$$

What this formula really shows is that the probability of each word in a class's vocabulary, occurring in a given document, is represented by the product of all individual word occurrence probabilities. Armed with this mathematical representation, we can then calculate the probability of a certain word appearing in a given document, for a given class i. Let $n_i(w_t)$ be the number of tweets in class i where the word w_t is found and N_i the total number of tweets in class i, so that:

$$P(w_t|c_i) = \frac{n_i(w_t) + 1}{N_i + 2}$$

This can be seen as the relative frequency of tweets in a given class that use a certain word w_t. The additional values to the numerator and denominator, are Laplace smoothing factors and are included to prevent probabilities of 0 occurring. These are possible if a word is used in a tweet that is not included in the training set. Similarly, the prior probability $P(c_i)$ can be calculated as follows:

$$P(c_i) = \frac{N_i}{N}$$

This gives us the of number of tweets in class i relative to the total number of tweets N in the training set. Using these formulas, each new tweet that is observed in the training set can be classified based on the maximum probability calculated. This probability in turn, depends on if the words contained in the tweet are also found in either of the two class's training vocabularies.

4.1.2 Lexicon-based

The lexicon-based model is a method that does not use any machine learning technique, thereby not requiring any training data. This method relies on pre-labelled exhaustive dictionaries of words and their given polarity. Besides needing this substantial information, the model is also comprised of a set of rules that are applied to a document in order to determine its polarity. These rules also consist of some text pre-processing steps added in order to improve classification accuracy. Once a document is assigned a calculated polarity, an appropriate cut-off point is used to classify documents into either having a positive or negative sentiment.

First, certain patterns in the raw text are identified and replaced. One of these patterns is abbreviations. These are common abbreviations in the French language and are used outside of social media platforms as well. In order to assign a polarity to these words, they are first converted to their full form (Table 4.1).

Table 4.1: A sample of abbreviations, their original form and a translation

Abbreviation	Original Phrase	Translation
Ltée./Ltee.	limitée	limited (Ltd.)
av. J.-C.	avant Jésus Christ	before Christ (B.C.)
boul.	boulevard	boulevard (Blvd.)
MM.	Madame	Madame (Mrs.)

Next, contractions are also converted to their original form, creating a separate word that can later be processed. These contractions are common in French and include some important information like negations (Table 4.2).

Table 4.2: A sample of abbreviations and their original form

Contraction	Original Word
m’	me
n’	ne
s’	se
qu’	que

In order to improve the lemmatization process, all diacritics are removed from letters, in order to prevent any misidentification. This means that some words may lose their inflection but will still hold their core meaning once lemmatized. The lemmatization process itself involves shortening words to their core form. Verbs have their conjugation removed and are reverted to the infinitive form while nouns are singularized, and adjectives take on their predicative version. Common articles, determiners and pronouns are also converted to a single form. These changes allow for a smaller required lexicon, with a single verb entry now covering a number of potential conjugations (Table 4.3).

Table 4.3: A sample of lemmatizations, and their translation

Original Word	Lemmatized	Translation
les	le	the
parisiennes	parisien	Parisian
danseuses	danseur	dancer
dors	dormir	to sleep

Each word processed is also assigned a Part of Speech tag such as 'VB' for verbs or 'JJ' for adjectives. These can later be used in assigning polarity to a sentence, as certain adverbs or adjectives, for example 'horrible', can contain useful sentiment information. Besides POS tags, negations within sentences are also identified and labelled. Words such as 'non' (no), 'pas' (not) and 'jamais' (never) can completely change the meaning of a sentence. Finally, once all the pre-processing steps have been complete, the rules for polarity can be applied, which will result in a final polarity per document. When a word in a given document is recognized, its polarity is extracted from a pre-defined lexicon. This polarity is summed up for all words found and averaged to find the base polarity that is then further modified. When a negation is detected, the base polarity gets multiplied by -0.5, which both changes the inflection and decreases the 'intensity' of the document. Negations oftentimes diminish the positivity of a noun, but do not reduce it to complete negativity. Instead, negative adverbs and adjectives are used when referring to a completely negative situation. As an example, 'The pancakes were not amazing' and 'The pancakes were disgusting', while both not describing pancakes positively, differ substantially in their negativity. That is why a negation holds a multiplier of -0.5 instead of -1. Besides this, modifier words such as 'very' also bolster whatever the current polarity is. This means that the inverse intensity is used to multiply the base polarity calculated. For all negations n, modifiers m and ordinary words w, the following formula is used to calculate polarity P_d of document d:

$$P_d = \prod_{i=1}^{n}[-0.5] \prod_{j=1}^{m}\left[\frac{1}{intensity_j}\right] \frac{1}{w} \sum_{k=1}^{w}[polarity_w]$$

As an example, the sentence 'Je n'aime vraiment pas jouer au foot.' (I really do not like playing football) would have its polarity calculated as follows:

$$P_d = \prod_{i=1}^{1}[-0.5] \prod_{j=1}^{1}\left[\frac{1}{2}\right] \frac{1}{1} \sum_{k=1}^{1}[0.60]$$

$$P_d = -0.5 * 0.5 * 0.6 = -0.15$$

Here, there is one negation ‘ne … pas’ (do not), one modifier ‘vraiment’ (really) and one regular verb ‘aime’ (like). These words have been recognized and are part of the lexicon, with words like ‘jouer’ (play) and ‘foot’ (football) not being included. Finally, the threshold of 0 is applied when classifying documents based on their polarity, with values smaller than 0 being assigned a negative class and those larger than 0 a positive class. While this method does not employ any machine learning algorithms, it is still a reliable way of detecting sentiment, with the user being confident in the custom feature set used. However, the limitations of this method are clear; the lexicon is exhaustive and cannot possible include all words, the pre-processing rules do not catch exceptions and the averaging of polarity is not always accurate.

4.1.3 Maximum Entropy

The maximum entropy classifier (MaxEnt) is, like the previously discussed Naïve Bayes model, a probabilistic model. However, unlike its related counterpart, MaxEnt does not assume all features to be conditionally independent from one another. The model is represented by the following formula, for all features l contained in the training documents:

$$P(c|d,\lambda) = \frac{1}{Z(d)} exp\left(\sum_{i=1}^{l} \lambda_i f_i(d,c)\right)$$

With $P(c|d,\lambda)$ the probability of class c being assigned to document d, given contraints λ. This is further dependent on the normalizing factor $Z(d)$, which is shown below:

$$Z(d) = \sum_{c=0}^{1} exp\left(\sum_{i=1}^{l} \lambda_i f_i(d,c)\right)$$

These two functions can then be combined into the final representation of the MaxEnt model:

$$P(c|d,\lambda) = \frac{exp\left(\sum_{i=1}^{l} \lambda_i f_i(d,c)\right)}{\sum_{c=0}^{1} exp\left(\sum_{i=1}^{l} \lambda_i f_i(d,c)\right)}$$

In all of these formulas, $f_i(d,c)$ represents the indicator function for a given class c , empirically, as follows:

$$f_i(d|c) \coloneqq \begin{cases} 1, if\, y = c\ and\ v(d) > 0 \\ 0\ otherwise \end{cases}$$

With y representing the true class of document d in the training dataset and $v(d)$ being the number of features from document d that are contained in the class vocabulary v. Intuitively, this means that a 1 is assigned if the class corresponds with the target class and if the document contains some word that is present in the bag-of-words vocabulary. Meanwhile, the λ shown in the previous formula represents a weight vector that is used to obtain a model as close as possible to uniform. This lambda parameter can be optimized through a variety of algorithms such as Generalized Iterative Scaling (GIS) or Improved Iterative Scaling (IIS).

4.1.4 Support Vector Machines

A support vector machine (SVM) is a model that focuses on finding a border to divide the data most efficiently. This border can be seen as a multi-dimensional hyperplane with the number of features influencing the number of dimensions used. In the context of this paper, the hyperplane

divides the data points into two classifications; the documents with positive sentiment and those with negative sentiment. Around this hyperplane, the data points that are closest and most influential to the plane's position, are considered support vectors. These crucial observations determine the final model and classification performance of the SVM. The more removed a hyperplane is from the training data points, the higher the chance of these observations being classified correctly. The aim then, is to increase the distance between the hyperplane and its support vectors as much as possible, while still classifying correctly as many data points as possible. This distance between the support vectors and the hyperplane is called a margin, and it can be either hard or soft depending on the cleanliness of the data.

In order to classify text documents, the SVM model takes as input unigram features, or words in the vocabulary of each class. The presence of these features is then used to construct a vector for each observation, in this case for each tweet in the dataset. These vectors can then be used in the dot-product representation of the hyperplane, with the nearest vectors selected by measuring the Euclidian distance of the margin. For the analysis of the twitter dataset, the assumption is made that the data is linearly separable, which allows for a linear kernel to be used. This assumption is done on the basis of previous successful work in the field of NLP using SVM models (Go et. al., 2009). This kernel represents the main method used in building the hyperplane, with a linear kernel corresponding to a linear formula representation of the plane.

First, the training dataset with l number of documents can be formulated in the following mathematical representation:

$$(x_i, y_i), \dots, (x_l, y_l)$$

with x_i being document i's sparsely populated feature vector and y_i being the document's sentiment label, either a 1 or a -1. The core formula of the SVM model can be formulated as follows:

$$f(x) = sign(w \cdot x + b)$$

with w being the weight vector perpendicular to the hyperplane, x being the feature vector and b representing the linear intercept vector. The margin between the i^th feature vector x_i and the hyperplane $\langle w, b \rangle$ is given by $y_i(w \cdot x_i + b)$. However, this representation also requires some constraint, as the weights vector w and intercept vector b can be increased to keep on increasing the margin. The functional margin that we have defined now has to be connected to the geometric margin that can be imagined in a two-dimensional example.

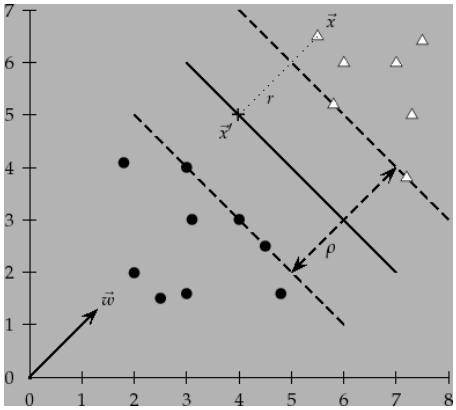


Figure 4.1: Geometric margin r of a point x in relation to decision boundary p.

Adapted source: Manning et al., 2008

In figure 1, this geometric margin r is represented for a two-dimensional point x. The point on the hyperplane closest to x, is represented by x'. This margin r, generated as a result of orthogonal projection, can then be defined as:

$$x' = x - yr\frac{w}{|w|}$$

In this formula, y decides the side of the hyperplane that the margin is measured on, and x' is on this hyperplane. This means that $w \cdot x' + b = 0$ must hold. By substituting the above formula for x' and rewriting w.r.t r, the following formula can be achieved:

$$r = y\frac{w \cdot x + b}{|w|}$$

Now, increasing the values for vectors w or b will not change the geometric margin r due to the normalization in the denominator. If the value for $|w|$ is set to 1, the geometric margin and functional margin will be the same. Because this margin can be scaled, the choice is made to require that it be equal to 1 for at least a single data point:

$$y_i(w \cdot x_i + b) \geq 1$$

for each item i in the dataset. The goal then, is to maximize this margin r on both sides of the hyperplane, while considering all training examples in the dataset. In order to solve this problem, a Lagrange multiplier α_i is included with the constraint for each observation i:

$$\text{maximize } \sum_i \alpha_i - \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j \, y_i y_j x_i \cdot x_j \text{ for each } i, j \text{ training examples } l$$

$$\text{subject to } \sum_i \alpha_i y_i = 0 \text{ and } \alpha_i \geq 0 \text{ for all } 1 \leq i, j \leq l$$

It is important to note that in the above formulations, i and j denote a pair of two different training examples from the same dataset. This implies that all pair combinations and their dot products are considered, in order to arrive at the optimal margin and corresponding hyperplane. The solution has been found to be:

$$w = \sum_i \alpha_i \, y_i x_i$$

and $b = y_k - w \cdot x_k$ for any x_k such that $\alpha_k \neq 0$

Which means that the following formula can be used to classify a given document feature vector x:

$$f(x) = sign\left(\sum_i \alpha_i \, y_i x_i \cdot x + b\right)$$

A negative sign (-1) is then mapped to a 0 or negative sentiment and a positive sign $(+1)$ to a 1 or positive sentiment.

4.2 Keywords Collection and Location Detection Models

Besides extracting the sentiment of a tweet, this paper also employs methods for examining the context of tweets. This includes the product or service used, the user's experience and the user's location. While the sentiment of a tweet is more subliminal and spread across different words, the context of a tweet is more concrete. In the next subsections, the methods used in finding this context will be discussed. While they use pre-trained models and extract literal words used in tweets, these methods are still useful for managers to understand the firm's customers.

4.2.1 Part of Speech Tagging

In order to extract insightful keywords from tweets, it is first necessary to identify the various syntax components of each sentence. This includes elements such as nouns, verbs and adjectives. Nouns are usually the objects talked about within a sentence and, in the context of the telecom app layer, can represent keywords such as 'YouTube', 'télé' (TV) and 'internet'. In terms of adjectives, descriptions of nouns can offer insight into the user experience with words like 'horrible'. These elements can be extracted from sentences using a pre-trained part of speech (POS) model. This model is trained on pre-labelled data in the target language and can offer accurate labelling on new data. The labels used in the pre-trained model are those of the universal dependencies format, which are widely used in POS training datasets (Nivre et al., 2016). The model then utilises the universal dependencies French Sequoia treebank, which is a dataset of annotated sentences in French (Candito & Seddah, 2012). This is further expanded by a deep syntactic annotation of the treebank to improve accuracy even further (Candito et al., 2014). The final pre-trained model's accuracy stands at 95.72% and was used in this paper to find keywords in tweets.

The model itself was trained using a simple but effective averaged perceptron algorithm. Once this algorithm is trained, its weights vectors can be used to predict a word's tag, given its context. This assignment of a tag for a given word i can be represented as follows:

$$Tag_i = argmax_{t \in T}(x_i \cdot w_{i,t})$$

With the tag t being chosen that maximizes the result of the dot product between the features vector x_i and weights vector $w_{i,t}$. Vector x_i is a 1 by X features row vector, while vector $w_{i,t}$ represents a column vector of X feature weights belonging to tag t. These two vectors both require certain steps to build, with both being dependent on the training dataset used. For every word analysed, the surrounding words are taken into account. This collection of words and their properties are referred to as a word's context and will be used in predicting the word's tag. Context features are all binomial variables such as the previous word being 'car', the previous word being a noun or even

the next word being a verb. To add to this, suffixes and prefixes of words in the context can also be used in order to let the model learn conjugations and patterns for sentence syntax. For each sentence in the training data, the method iterates over the words and their respective tags, each time generating a new observation. This observation consists of the context features for a given word and the word's tag as dependent variable, with each observation representing a training example. Now there is a very sparse matrix of training data, with context features and their tag. Next, a weights matrix is generated with a random initial allocation of weights $\alpha_{x,t}$ for a given feature and tag. This is an X features by T tags matrix and contains a collection of weights for each feature and tag pair. For predictions, each tag will be tested by taking a 'slice' of this weights matrix as a weights column vector for each corresponding tag.

The algorithm then iterates over each training example, every time using the context features to predict a word's tag. If this prediction is correct, it updates the weights of the features used in the prediction by one. Conversely, if a prediction is wrong, the weights associated with the feature and tag pairs used are decreased by one. This iterative process is performed for all training examples, with at the end each weight being divided by the times it was updated. This averaging also takes into account how often or rare a weight gets updated and adjusts the accumulator accordingly. This averaging also helps in generalising the weights and increasing their performance, without this feature the model's predictions would be highly dependent on the training data. Once all weights are finalized, the pre-trained model can be applied to new testing data.

4.2.2 Named Entity Recognition

While the POS tagger finds the syntax used in sentences, with the corresponding tags of individual words, the named entity recognition (NER) model finds real-world objects in sentences. These objects can be persons, organisations and locations. In this paper, the NER tagger will be used to detect place names within sentences, allowing for a location to attributed to a given tweet. The NER model used in this paper, is a pre-trained convolutional neural network that used the French wikiNER corpus, which is a labelled dataset of Wikipedia (Nothman et al., 2013). These labelled named entities are passed through a convolutional neural network, which transforms the raw text data into "Bloom" embeddings. These embeddings are then fed into a fully connected neural

network for multi class classification. In training the model, the gradient of the loss function is used to calculate the error between predictions and training labels. Using this error, the model iteratively improves after each training example.

This pre-trained model then can be used to classify any entities found in tweets. While the F-score of the NER tagger used is 85.63%, it performs slightly worse on datasets that deviate from Wikipedia's language syntax. Twitter poses plenty of challenges for the pre-trained NER model, as the text processed can contain slang words and a modified syntax. This paper tries to mitigate this by pre-processing the input data, meaning the NER tagger receives a cleaner text and can perform better.

5. Results

This section is subdivided into four different parts, with each explaining the results and performance of a different method. Each section also corresponds with a different insight that can be extracted from the collected tweets. In practice, managers can combine these insights into either a live dashboard overview, or a periodic report. Some of the insights extracted can be applied to a per-tweet basis, while others require an aggregated set of tweets.

5.1 Sentiment Classification

For the classification of sentiment, two different sets of data were used in calculating model performance. The first dataset is part of the training data collected using the Twitter API. This set represents 20% of the total amount of tweets collected for training and was left out when training the models. These tweets contain emojis, Twitter-specific and francophone slang and other Twitter-related characters. Most importantly, they are not specific to the scope of telecom providers, meaning that the subject matter of these tweets varies greatly. Meanwhile, a second testing dataset that consist of tweets aimed at a specific telecom provider is also included. This dataset was collected using the provider's brand name as search query in the Twitter API and is therefore more representative of tweets within this paper's scope. Because these tweets concern a more specific

subject matter, the performance of sentiment classification models on this dataset will be more interesting. Because this telecom-specific dataset has unbalanced classes with a 20:51 ratio of positive to negative tweets, downsampling was performed. This procedure involves sampling observations randomly from the overrepresented class in order to obtain balanced classes. While this means that the final dataset will have 40 observations compared to the original 71, its classes will instead be balanced at a 1:1 ratio. This means that performance measures are not biased towards class size.

In order to measure the performance of the models on the testing datasets, this paper will use classification accuracy, as is common with binomial text classification. This accuracy is the result of a confusion matrix, generated using both the labels on the testing dataset and the predictions generated by a given model. An overview of a confusion matrix is given in Table 5.1.

Table 5.1: An example of a confusion matrix reporting format.

		Label	
		Positive	**Negative**
Prediction	**Positive**	True Positives	False Positives
	Negative	False Negatives	True Negatives

In a confusion matrix, each value represents a summation of the amount of observations that have the corresponding label and prediction. As an example, the True Positives would be a count of all positive tweets in the corresponding dataset that have been classified as positive. The accuracy then, is the number of predictions made by the model that were correct as a percentage of all predictions. This is calculated as follows:

$$Accuracy = \frac{True\ Positives + True\ Negatives}{Number\ of\ observations}$$

The overview of accuracies for both the fraction of the training dataset (Emoji Dataset) and the telecom provider specific dataset (Telecom Dataset) can be seen in Table 5.2.

Table 5.2: Accuracies of sentiment classification models on two different datasets.

Classification Model	**Emoji Dataset**	**Telecom Dataset**
Naïve Bayes	79.56%	57.50%
*Lexicon Based**	53.08%	77.50%
Maximum Entropy	81.51%	60.00%
Support Vector Machine	80.05%	72.50%

*The lexicon based model does not require training on Twitter data.

There appears to be a disparity in performance between the two datasets, depending on the model used in classification. For trained models, the accuracy decreases when telecom-specific tweets are classified. The clearest example of this is the Naïve Bayes classifier decreasing in accuracy from 79.56% to 57.50%. Besides this, both the Maximum Entropy and Support Vector Machine models suffer a considerable drop in accuracy, with drops from 81.51% to 60.00% and from 80.05% to 72.50% respectively. This effect, however, is reversed when looking at the performance of the Lexicon Based model on both datasets. Here, the accuracy is higher on the Telecom Dataset compared to the Emoji Dataset, with an accuracy of 77.50% and 53.08% respectively. This could be due to the difference in syntax between the two datasets. While the Telecom Dataset contains tweets where users are focused on giving impactful and detailed feedback, the Emoji Dataset simply contains positive and negative tweets. These tweets can range greatly in length and quality, with no minimum character length imposed. Furthermore, this low accuracy of 53.08% on the Emoji Dataset could be an indication of the quality of the dirty labels assigned to tweets. In order to further understand the performance of these models, this section takes a more in-depth look at the classification models.

5.1.1 Naïve Bayes Model

The confusion matrix for the Naïve Bayes classifier on the Telecom Dataset can be seen in Table 5.3.

Table 5.3: Confusion Matrix of the Naïve Bayes Model on the Telecom Dataset

		Label	
		Positive	**Negative**
Prediction	**Positive**	16	13
	Negative	4	7

The Naïve Bayes model seems to be having trouble classifying negative tweets correctly. A reason for this could be the lack of context used when classifying, as only unigrams are included as features in the model. Furthermore, this model is sensitive to dirty data and requires extensive cleaning, which could at the same time remove some important features from the dataset. Lastly, there could simply be a large discrepancy between the dataset used to train and the Telecom Dataset, especially when it comes to sentence topic. When digging deeper into the reasons for this poor performance on negative tweets, it is evident that the negative features list used is quite limited. When the model encounters new words that are not present in the training data, it is simple ignored. Because of this, tweets with vocabulary outside of the feature set get misclassified. When looking at the confusion matrix in Table 5.3, a bias towards predicting a positive sentiment is noticeable. A possible reason for this could be that negative tweets in the training dataset do not have sufficient distinct negative words. This would result in the model not having enough negative features that could be attributed to a negative sentiment correctly.

5.1.2 Lexicon Based Model

The confusion matric in Table 5.4 corresponds with the Lexicon Based classifier's performance on the Telecom Dataset.

Table 5.4: Confusion Matrix of the Lexicon Based Model on the Telecom Dataset

		Label	
		Positive	**Negative**
Prediction	**Positive**	16	5
	Negative	4	15

Using this model, a much higher accuracy is achieved. Only a mere 9 observations are misclassified by the Lexicon Based model, for an accuracy of 77.50%. While this model performs well on the Telecom Dataset, it falters when classifying tweets in the Emoji Dataset. The confusion matrix for the Emoji Dataset can be found in Table 5.5.

Table 5.5: Confusion Matrix of the Lexicon Based Model on the Emoji Dataset

		Label	
		Positive	**Negative**
Prediction	**Positive**	9226	7998
	Negative	10768	12006

In this case the model performs poorly and seems to have a tendency of classifying tweets as negative, irrespective of label. This can be attributed to distorted syntax, where many tweets use elements such as hyperlinks and emojis. These kind of internet-specific characters hamper the model's ability to classify, as the algorithm used is meant for French sentences. Notably, the model does perform well on the Telecom Dataset, which could indicate a clearer and more sentence-like syntax in customer feedback.

5.1.3 Maximum Entropy Model

The Maximum Entropy model was trained using both GIS and IIS methods, with little difference in accuracy between the two. IIS performs slightly better, with an increase of 0.01 percentage points in accuracy. The training is run until accuracy increases by less than 0.001, which equates to a change smaller than 0.1%. For GIS, this leads to 20 iterations, while IIS takes only 16 iterations to converge. The confusion matrix for the IIS trained model is show in Table 5.6.

Table 5.6: Confusion Matrix of the Maximum Entropy Model on the Telecom Dataset

		Label	
		Positive	**Negative**
Prediction	**Positive**	16	12
	Negative	4	8

A similar issue to the Naïve Bayes method can be seen here, where the model has trouble classifying negative tweets correctly. This could be again due to a lack of features used for classification. Overall, the two methods share some similarities and both generate lists of features from the training dataset, to be used when classifying new observations.

5.1.4 Support Vector Machine Model

In order to find a model that could perform best on the Telecom Dataset, a linear kernel was chosen for the Support Vector Machine. Furthermore, the final model was chosen after optimizing the penalty parameter C using a 5-fold cross validation. This cost hyperparameter influences the margin chosen for the hyperplane. More precisely, for higher levels of penalty C, the model will aim for a hyperplane with a smaller margin, at the risk of overfitting to the training data. Essentially, the trade-off between having a large margin and classifying as many observations as possible is taken into consideration when choosing C. The range of values 0.01, 0.1, 1, 10, 100 was used in combination with a 5-fold cross validation. This means that for each value of C, 5 different accuracies were generated on the validation set. Using this method prevents any knowledge from

the final Emoji Dataset from leaking into the model. The penalty C resulting in the highest accuracy was 0.1. After selecting the optimal SVM model configuration, the confusion matrix in Table 5.7 was generated after predicting on the Telecom Dataset.

Table 5.7: Confusion Matrix of the SVM Model on the Telecom Dataset

		Label	
		Positive	**Negative**
Prediction	**Positive**	14	5
	Negative	6	15

The performance of the SVM model is noticeably better than that of other trained models. With an accuracy of 72.50%, it outperforms both the Naïve Bayes and Maximum Entropy models. While it does not reach the same level as the Lexicon-Based model, its performance is impressive considering that it uses no pre-labelled list of features.

5.2 Extracting Insights from Tweet Content

This subsection will cover the application of two pre-trained models, used for obtaining information about the content of a tweet. By using a part of speech tagger and a named entity recognition model, we were able to extract insightful keywords and locations. Importantly, while the original Telecom Dataset of 71 tweets contained no embedded coordinates, as shared by users, we were able to find 12 locations that could could be parsed to coordinates. While this may seem as a small amount, only 17 tweets contained some place name, meaning our method had recall rate of 70.59%.

5.2.1 Keyword Extraction

For extracting keywords, there are two important approaches that have been applied. The first approach is applied on a per-tweet basis, meaning that tweets can be analysed individually for

keywords. This method involves using part of speech (POS) tagging to extract nouns and adjectives as descriptors of product or service, and user experience. To demonstrate, Table 5.8 and 5.9 show how two tweets sampled from the Telecom Dataset are used to extract relevant POS tag keywords.

Table 5.8: A sample of two tweets from the Telecom Dataset and the extracted keywords using POS tagging

Tweet	Extracted Keywords
@TelecomProvider Merci! Vous êtes une excellente compagnie. J'adore vos services.	TelecomProvider, compagnie, excellente, J', services
@TelecomProvider pas de télé ni internet à Prévost. Panne?	TelecomProvider, télé, internet, Prévost, Panne

Table 5.9: A sample of two tweets from the Telecom Dataset and the extracted keywords using POS tagging, translated to English

Tweet	Extracted Keywords
@TelecomProvider thank you! You are an excellent company. I love your services.	TelecomProvider, thank you, company, excellent, I, services
@TelecomProvider no TV or internet in Prévost. Outage?	TelecomProvider, TV, internet, Prévost, Outage

Both the tweets shown in Table 5.8 have been collected using the Twitter API and represent genuine user-generated feedback. The first tweet has a positive sentiment, while the second tweet has a negative sentiment. By using the POS tag extraction method, some important keywords were found. In the case of the positive tweet, the adjective 'excellente' was identified, together with 'compagnie' and 'services'. Meanwhile, the keywords extracted from the negative tweet include some important descriptions of service like 'télé' and 'internet', while at the same time offering an explanation of experience in 'Panne'. There is also a proper noun extracted, namely 'Prévost', as

a location. While this keyword could be useful in case the Named Entity Recognition tagger fails, it cannot be used to parse a mappable location. In the case of simple POS tagging, the proper noun for a location could not be distinguished from any other proper noun.

5.2.2 Location Extraction

With a recall rate of 70.59%, we have shown that using named entity recognition (NER) to identify locations in tweets can be successful. For each tweet in the Telecom Dataset, of which none contained an embedded set of coordinates, the NER model was applied. This resulted in a possible location tag for place names contained in the tweet's text. Identified locations where then passed through a reverse geotagging service, that parsed a given place name to a set of coordinates. It is important to note that for broad place denominations such as 'Montréal' or 'Canada', a coordinate set is chosen as close to the geographic centre of the location as possible. As such, 'Montréal' is denoted by the coordinate pair 45.4972159, -73.6103624 for latitude and longitude. While place names representing large areas are assigned a more limited location, precise place names come with more precise coordinates. However, in this context precision can mean different things. Importantly, the smaller the geographic area that place names represent, the more precise the coordinates will be. As an example, a postal code or street name will be more precise than the name of a city. At the same time, the name of a village can be more precise than the name of city's suburb, as the former represents a smaller geographic area than the latter. The more precise the location extracted, the less geographical area it covers, and them more useful it can be in obtaining insights. Tables 5.10 and 5.11 give a sample of locations extracted from tweets in the Telecom Dataset with their corresponding coordinates.

Table 5.10: A sample of three tweets from the Telecom Dataset and the extracted location using an NER tagger

Tweet	Location	Coordinates (latitude, longitude)
@TelecomProvider bonjour. Il semblerait que sur Montréal il y ai un problème non? Pertes de pacquet récurrent.	Montréal	45.4972, -73.6104

@TelecomPrtovider la vitesse d'internet est pourrie ce soir. Vous avez un problème ? À Beloeil ?	Beloeil	45.5643, -73.204
@TelecomProvider J'espère qu'il y aura un service pour le village de Fulford ma petite fille serait tellement heureuse.	Fulford	45.2967, -72.5579
@TelecomProvider Le réseau est très lent et je paye pour la haute-vitesse. On se croirait au Mexique. Vitesse descendant.	Mexique	*N.A.*

Table 5.11: A sample of three tweets from the Telecom Dataset and the extracted location using an NER tagger, translated to English

Tweet	**Location**	**Coordinates (latitude, longitude)**
@TelecomProvider hello. It seems that there is a problem in Montreal right? Recurring packet loss.	Montreal	45.4972, -73.6104
@TelecomPrtovider the internet speed is rotten tonight. Is there a problem? In Beloeil?	Beloeil	45.5643, -73.204
@TelecomProvider I hope there will be service in the village of Fulford my little girl would be so happy.	Fulford	45.2967, -72.5579

@TelecomProvider The network is very slow and I pay for high speeds. It feels like Mexico. Falling speed.	Mexico	*N.A.*

In these example tweets, different types of locations were extracted. While Montreal is a city with a large geographic area, Beloeil is more narrowed down as a suburb of Montreal. Meanwhile, Fulford is a small village and Mexico represents an entire country. For each place name, the respective coordinates have been parsed, which can then be plotted on a map. In the case of Mexico however, a set of coordinates was not generated, as the bounds were set only for within Canada. This offers a useful restriction for filtering out tweets that either do not originate from the target market, or simply refer to a location they are not currently in.

However, while the pre-trained NER tagger offers a useful functionality, it does not always succeed in finding place names in tweets. From the 17 tweets containing a location in the Telecom Dataset, the NER tagger fails to find a place name in 5 tweets. Two examples of these failures are given in Table 5.12.

Table 5.12: A sample of two tweets from the Telecom Dataset where the NER tagger failed to find a place name, and their English translations

Tweet	**Translation**
@TelecomProvider y a t il une panne a Jonquiere?	@TelecomProvider is there an outage in Jonquiere?
@TelecomProvider Y a t'il des problèmes de réseau à Joliette ? J'ai perdu internet depuis une trentaine de minutes	@TelecomProvider Are there network problems in Joliette? I lost internet connection for about thirty minutes

In the case of the first tweet, the NER tagger does not recognise ‘Jonquiere’ as a location because of the missing preposition. In this case, spelling ‘à’ (in) without the accent leads to the NER tagger not recognising the following word, ‘Jonquiere’, as a location. Such spelling mistakes, while small and insignificant to users on Twitter, can pose challenges when parsing text. The second tweet does use the correct preposition, but in this case the NER tagger sees ‘Joliette’ as a name instead of location. This could be due to the occurrence of ‘Joliette’ as a name more frequently than as a location in the NER model’s training dataset. This can occur with smaller locations such as towns or villages. In this case, Joliette seems to be a smaller city North of Montreal with a population of 19,621 (Statistics Canada, 2015). In Figure 5.1, all the extracted location coordinates are plotted on a heatmap, with the frequency of tweets represented by colour.

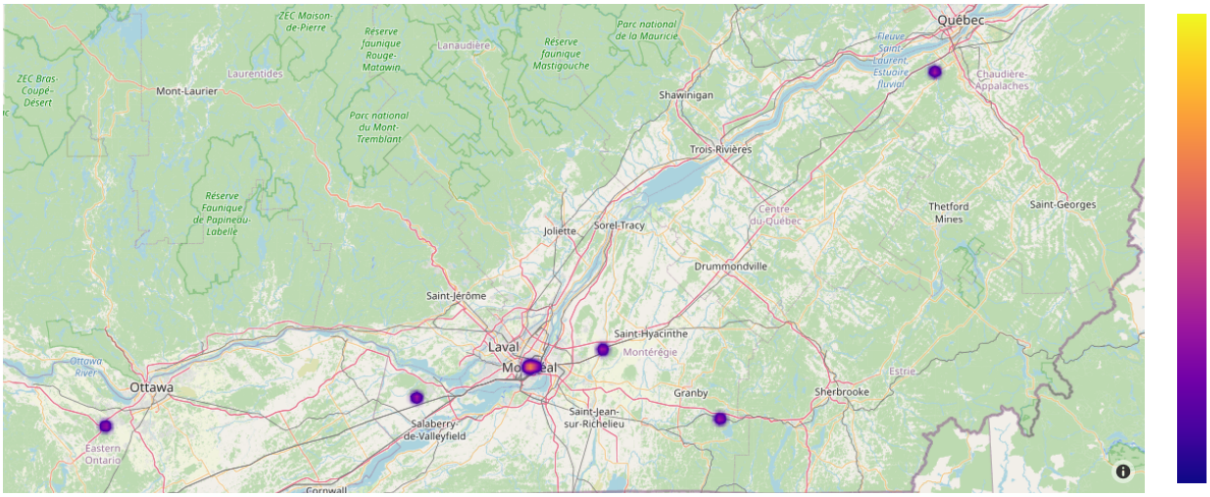


Figure 5.1: Heatmap of tweets plotted by NER extracted location

Note: Yellow represents a high frequency, with blue being a low frequency

This heatmap was generated using the tweets with an attributed location from the Telecom Dataset, consisting of around two weeks’ worth of data. However, such an overview can be generated on any time scale, with live plotting of data being a useful tool for customer feedback monitoring. The tweets used in plotting the heatmap can be filtered by sentiment, displaying, for example, only negative sentiment hotspots.

6. Conclusion

6.1 Discussion of Findings

Using the methods described in this paper, we have successfully applied sentiment analysis to social media customer feedback. In doing so, we have shown that methods previously applied in an English context, are also successful in French. Furthermore, by using emojis as additional search queries, we have adapted the distant learning method of data collection to be more suited to current social media syntax. We have also shown a method for extracting location from social media feedback, in the case that one is mentioned by the customer. This method is applicable to cases where customers do not share an embedded set of coordinates.

In this section, we will first elaborate on the conclusion per sub-question, followed by a final conclusion. This final conclusion will answer the research question and elaborate on possible applications of this framework for managers. . In the following section, Limitations and Recommendations, possible improvements and changes will be discussed.

6.1.1 How can the sentiment of tweets be extracted?

The first of the sub-questions looks at finding the sentiment behind tweets, which will offer a layer of context to the customer feedback. We have shown that it is possible to apply machine learning methods in extracting sentiment from tweets in French. This, in turn, shows the applicability of this framework to multiple languages and markets. For extracting sentiment, four different methods have been applied, each with varying degrees of success. Table 5.2 from the results section is again repeated in Table 6.1 for convenience.

Table 6.1: Accuracies of sentiment classification models on two different datasets.

Classification Model	Emoji Dataset	Telecom Dataset
Naïve Bayes	79.56%	57.50%
*Lexicon Based**	53.08%	77.50%
Maximum Entropy	81.51%	60.00%
Support Vector Machine	80.05%	72.50%

*The lexicon based model does not require training on Twitter data.

First, while looking at the performance of the models on the Emoji Dataset, it seems that all three trained models (Naïve Bayes, Max. Ent. and SVM) have a comparable accuracy. It is once the models are applied to the Telecom Dataset, that the differences truly start showing. Both the Naïve Bayes and Maximum Entropy models fail to live up to their accuracies on the Emoji Dataset. Meanwhile, the Support Vector Machine model comes closest to its validation set performance, with an accuracy of 72.50%. Lastly, the Lexicon model outperforms the SVM method slightly.

When looking at the application of these methodologies, managers have to not only take into account the reported accuracies but also the methods themselves. While the Lexicon model does perform better than the SVM and far better than the Maximum Entropy and Naïve Bayes models, it does require a large pre-labelled lexicon. While for this paper, a rather extensive pre-labelled lexicon was found, that is no guarantee for each future application. A lexicon's quality depends on factors such as the linguistic skill of the person doing the labelling as well as the variety of words and topics included. While in this case the Lexicon model does perform well, in other languages the resources may simply not be available. Meanwhile, an SVM model depends only on the factors it gathers from the training dataset. While the cleaning process can be modified to be more language-specific, a more superficial one that does not include steps such as lemmatization, can also be used. This flexibility offer managers the options to apply their analysis in different languages and to different markets, which is especially relevant for multinational firms. Furthermore, the SVM model can be improved by choosing to collect only industry-specific training data. While this may take considerably longer, especially for narrow specifications, it can offer a more improved performance by bridging the gap between the training and testing data. Thus, while the Lexicon method does perform better, this paper recommends that managers look at an SVM model for assessing customer feedback sentiment. This method has a higher performance ceiling and is much more flexible than its pre-labelled counterpart.

6.1.2 How can the content of tweets be extracted?

With respect to location, this paper has shown a method for extracting place names from a tweet's text that has an effective recall rate of 70.59%. This means that firms do not need to rely on users sharing their location through the social media platform's privacy settings. Customers need only include a place name in their tweet, which can then be extracted using a Named Entity Recognition (NER) tagger and further parsed to coordinates. While the NER tagger is not perfect, it does perform well in the context of customer feedback, where a location is usually specified by a city name. However, in the case that a smaller location, such as a village, town or even suburb is mentioned, the NER tagger should be able to identify this as well. While not all tweets contain some information about the user's location, this paper recommends managers to use this feature in conjunction with some form of online engagement. In the event that a location cannot be identified, a company could choose to engage with the customer, asking them for a location where they had their product or service experience. This could be in the form of a chatbot, that would request additional information from users based on the missing pieces of insight.

For the extraction of the product or service used, this paper has used part of speech tagging to find insightful keywords. By extracting part of speech tags, managers can learn more about the content of user feedback at a glance. More importantly, these keywords can be cross correlated with internal data to make the generated data more precise and useful for managers. While this method of keyword extraction is not perfect, it is intended as a starting point for managers who wish to gain more insight into the firm's social media presence.

6.1.4 How can insights be extracted from Twitter to help with strategic decision making?

Overall, extracting insights from a social media platform such as Twitter poses a variety of challenges. First, in order for any analysis to be effective, the input data must be cleaned and pre-processed properly. This cleaning is dependent both on the medium that was used to place the feedback online, as well as the language it was written in. In this paper, the data section explained how various Twitter-specific pre-processing steps were undertaken to remove unwanted characters and phrases. Furthermore, a French-specific lemmatization reduced the amount of features used in machine learning models for sentiment analysis. With respect to the methods for extracting

sentiment, each is different and offers a unique edge compared to the others. The SVM model offers the best combination of performance and flexibility, meaning managers can apply the method to different languages. While an embedded location would be nice, it is rarely included in tweets, which is why the NER tagger is especially useful in extracting a customer's location. By finding place names within text, the NER tags can then be parsed to coordinates, which can be used by managers to filter feedback to specific markets. Lastly, extracting part of speech tags and filtering these by nouns, pronouns and adjectives, offers managers an overview of what keywords are being used by their customers. These keywords can offer insight into the products being complained about or the services being recommended to others. At the same time, managers can use these POS tags to better understand the experience of the user.

This process is best depicted in Figure 6.1. Here, all parts are divided into three sections; Data, Models and Insights. The insights obtained can then be used either in a live dashboard, or as an aggregated periodical report for managers to use in their strategic decision making. Especially useful is the timestamp that comes with each tweet, which offers firms a way to correlate social media activity to internal processes and events. In order to obtain even more efficiency when applying the analysis described in this paper, managers could set up a chatbot. This would allow for an automated request for more information, in the situation that not enough was provided. While this can easily apply to no location being registered buy the NER tagger, it can also be extended to include requests for product or service and even experience. Lastly, managers can also use the Twitter API to extract the follower counts of the users behind the tweets. This would allow for a 'weighting' of customer feedback depending on the user's following size and influence. Besides this, the overview can be expanded to include amount of views and interactions that a tweet has, which allows firms to understand which tweets have had the most impact with its customer base.

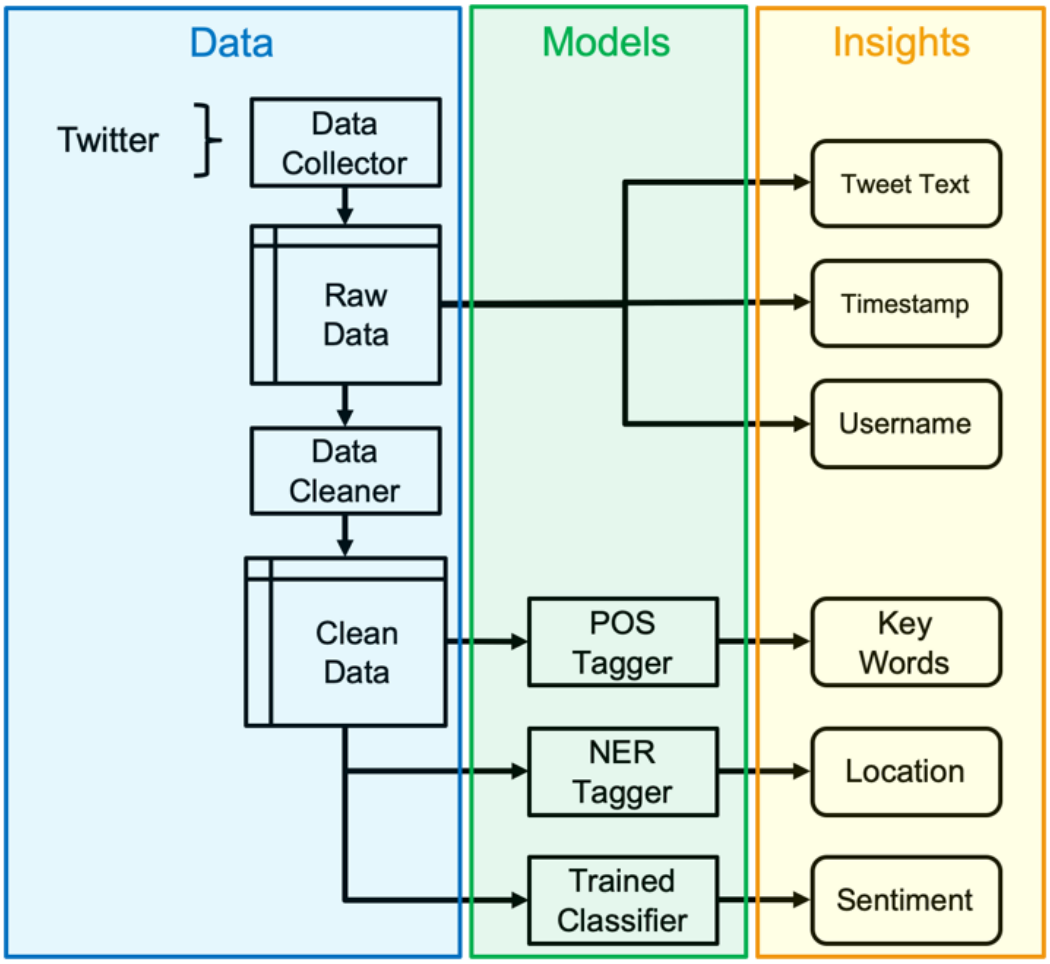


Figure 6.1: An overview of the processes applied in this paper for extracting insights from Tweets.

While this paper aims to set the basis for an impactful social media monitoring system, managers can further extract insights such as user followers and retweets to get the full picture of a firm's online image. By performing this analysis for other competitors in the same industry, a firm can visualize its online market share. Specifically, this method can be applied to, for example, visualize on a heatmap where the competitor's services or products are failing and tailor the marketing campaign to target these areas. As an example, the manager of a telecom provider could apply this methodology to find areas of the country that are underserviced and where customers are unsatisfied with the competition's offering. This would allow the firm to target these customers in an advertising campaign. Going even further, by tracking influential usernames, a firm could also

target these specific online customers for extra feedback or product promotions. Social media analysis is a powerful tool and one that all managers should consider using when deciding their next move in the market. The methods described in this paper are an especially effective way to monitor a brand or company's online presence.

6.2 Limitations

Although the processes in this paper are very applicable and provide useful insights for managers and marketing teams, there are some limitations and areas with failures. In this section, these shortcomings will be highlighted and explained, with indications for how to improve. Lastly, possible future research will be discussed, which will outline further steps for improving the collection of insights from online customer feedback.

6.2.1 Limitations in Data

The data section itself is subdivided into two parts; data collection and data pre-processing. First, the method used for collecting data brings with it some limitations. In order to train a machine learning model, accurately pre-labelled data is required, which in this context means user-generated tweets that have been labelled by sentiment. Because of the substantial monetary and time costs involved in manually labelling a large enough dataset, the decision was made to use distant supervision in order to train the sentiment analysis models. While this method did provide a substantial pre-labelled dataset of 200,000 tweets, the quality was of course lower. When using emoticons and emojis as search queries for positive and negative sentiments, attributes such as irony and sarcasm are not taken into account. Furthermore, the tweet's length, while limited to a maximum of 140, can oftentimes be too short to contain valuable information for training a model. In the context of customer feedback, users offer enough information for the target firm to understand the sentiment and experience. However, the subset of tweet collected for training the machine learning models is not representative of the 'population' of tweets pertaining to customer feedback. Concretely in this paper's context, the training data collected is not customer feedback specific, let alone being about the telecom industry. This means that the model misses out on useful elements when choosing its features, such as industry-specific slang and acronyms.

It also seems that models such as Naïve Bayes and Maximum Entropy struggle to classify negative tweets, which could either be due to incorrect emojis used or a lack of distinctively negative words. In the former, the emojis used in search queries for negative tweets could be too ineffective in identifying a negative sentiment, compared to positive emojis. This could lead to a large amount of tweets being collected as having a negative sentiment, while in fact being neutral or even positive. Meanwhile, having a lack of distinctly negative words, would mean that finding features with great discriminating power could be hard. This could be the result of negative tweets in the training dataset using a broader range of vocabulary compared to positive tweets. In order to improve the way that data is collected, managers should carefully consider some of the queries used. Potentially better choices could be made for the emojis and emoticons included. Another improvement could be adding some manually labelled and industry-specific tweets to the training dataset. This would not necessarily have to be firm-specific and could include tweets aimed at companies outside of the target market. Furthermore, these manually labelled tweets could also include general forms of customer feedback from other industries, in the same language.

In the pre-processing steps used in the data section, there are also some improvements to be made. Specifically, most text cleaning and processing tools are either oriented around a specific text data source, or a specific language, rarely around both. In order to sidestep this issue, this paper used explicit processing rules, which both removed unwanted Twitter-specific elements, while at the same time preserving the format of French text. However, this list of replacements and adjustments is not exhaustive and could further be extended to include impurities in the text that were missed. Besides that, methods such as lemmatization or stemming are language-specific and are not designed to work optimally only online user-generated text. Issues such as spelling mistakes, unknown acronyms and internet slang are some of the limitations to cleaning data completely.

6.2.2 Limitations in Methods

When looking at the sentiment analysis models two main issues come up; the processing power required to train models on large datasets and the limited size of the training dataset. First, the

amount of training data collected, while good for improving model performance and generalizing the model, can prove challenging. In this paper, a training dataset of 200,00 tweets was collected, with 160,000 tweets representing the 80% used in training the models. This can lead to long processing times, especially for the support vector machine model. However, once trained and tuned for its hyperparameters, this model can be saved and later used in analysing new data. Next, the size of the testing dataset also poses an issue. In order to get a good measurement of a model's performance, a testing dataset must be used, representative of the tweets found in the model's application. What this means, is that the tweets collected where specific to a given telecom provider. However, this also limits the amount of tweets that can be captured. For this paper, two weeks' worth of tweets were collected, all aimed at the target telecom provider, to build a testing dataset. This only resulted in 71 tweets, which was further reduced to a balanced dataset of 40 tweets after downsampling. While this removed any bias for a particular class in performance metrics, it also amplified the effect of each independent classification. This means that the accuracy measured of each model on the testing dataset could change if more tweets were added. Given more time this testing dataset can be expanded to include a more substantial set of tweets, allowing managers to better understand the performance of these models.

Besides the sentiment analysis, the methods used to extract keywords and locations also have their limitations. The named entity recognition tagger, while being trained on a substantial dataset, does not always succeed in finding place names. This failure can be attributed to the different syntax used on social media platforms, as well as the terminology used in a specific industry. As an example, in one instance a customer mentioned the area code of one of Montreal's suburbs as a location for service outage. While local users in Canada can understand this reference, the NER tagger trained on a general French-language dataset cannot pick this up. This use of suburbs to describe location is seen in numerous tweets aimed at telecom providers, with most using the suburb's full name. While an NER tagger may be trained to find country, province, city and even village names, it may fail with items such as postal or area codes. Some more pre-labelled training data can be added to improve the NER tagger, with industry-specific terminology for location. The part of speech tagger can also be improved or modified, extracting not only nouns, proper nouns and adjectives but also verbs an adverbs in order to paint a broader picture.

6.3 Recommendations for Future Research

Future research should focus on improving the sentiment analysis process, by including the emojis and emoticons used in a tweet. While these are useful in collecting tweets for a training dataset, they are removed in the cleaning step in order to decrease the models' dependence on them as features. A possible application could be to use a pre-trained word vector model, such as the CamemBERT model, to analyse tweets (Martin et al., 2019). This approach would require using pre-trained word embeddings in combination with a neural network model, to obtain a more powerful sentiment classifier. Besides trying new models, the current methods could be improved for better results. For example the dictionary of words used in the Lexicon based model could be expanded to include industry or platform-specific terminology. Furthermore, other features could be used in determining a tweet's sentiment, such as the amount of comments and retweets. Tweets with a negative sentiment could 'go viral' more easily than those with a positive sentiment.

7. References

Stricker, G. (2014, December 10). The 2014 #YearOnTwitter. Retrieved June 16, 2020, from https://blog.twitter.com/official/en_us/a/2014/the-2014-yearontwitter.html

SEC, & Twitter. (2019, March). *Twitter Q4 and Fiscal Year 2019 Shareholder Letter*. Retrieved from https://www.sec.gov/Archives/edgar/data/1418091/000141809120000019/twtrq419ex991.htm

Read, J. (2005, June). Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL student research workshop* (pp. 43-48).

Bird, Steven, Edward Loper and Ewan Klein (2009).
Natural Language Processing with Python. O'Reilly Media Inc.

Novak, P. K., Smailović, J., Sluban, B., & Mozetič, I. (2015). Sentiment of emojis. *PloS one*, *10*(12), e0144296.

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N project report, Stanford*, *1*(12), 2009.

Pak, A., & Paroubek, P. (2010, May). Twitter as a corpus for sentiment analysis and opinion mining. In *LREc* (Vol. 10, No. 2010, pp. 1320-1326).

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. *arXiv preprint cs/0205070*.

Denecke, K. (2008, April). Using sentiwordnet for multilingual sentiment analysis. In *2008 IEEE 24th international conference on data engineering workshop* (pp. 507-512). IEEE.

Ghorbel, H., & Jacot, D. (2011). Sentiment analysis of French movie reviews. In *Advances in Distributed Agent-Based Retrieval Tools* (pp. 97-108). Springer, Berlin, Heidelberg.

Rhouati, A., Berrich, J., Belkasmi, M. G., & Bouchentouf, T. (2018). Sentiment Analysis of French Tweets based on Subjective Lexicon Approach: Evaluation of the use of OpenNLP and CoreNLP Tools. *J. Comput. Sci.*, *14*(6), 829-836.

Martin, L., Muller, B., Suárez, P. J. O., Dupont, Y., Romary, L., de la Clergerie, É. V., ... & Sagot, B. (2019). Camembert: a tasty french language model. *arXiv preprint arXiv:1911.03894*.

Manning, C., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval* (pp. 319–329). Cambridge University Press. https://nlp.stanford.edu/IR-book/

Nivre, J., De Marneffe, M. C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., ... & Tsarfaty, R. (2016, May). Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)* (pp. 1659-1666).

Candito, M., & Seddah, D. (2012, June). Le corpus Sequoia: annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical.

Candito, M., Perrier, G., Guillaume, B., Ribeyre, C., Fort, K., Seddah, D., & de La Clergerie, É. V. (2014, May). Deep syntax annotation of the sequoia french treebank.

Nothman, J., Ringland, N., Radford, W., Murphy, T., & Curran, J. R. (2013). Learning multilingual named entity recognition from Wikipedia. *Artificial Intelligence*, *194*, 151-175.

Ofir, C., & Simonson, I. (2001). In Search of Negative Customer Feedback: The Effect of Expecting to Evaluate on Satisfaction Evaluations. *Journal of Marketing Research, 38*(2), 170-182. Retrieved July 6, 2020, from www.jstor.org/stable/1558622

Kumar, V., & Pansari, A. (2016). Competitive Advantage Through Engagement. *Journal of Marketing Research, 53*(4), 497-514. Retrieved July 6, 2020, from www.jstor.org/stable/44134928

Schweidel, D., & Moe, W. (2014). Listening In on Social Media: A Joint Model of Sentiment and Venue Format Choice. *Journal of Marketing Research, 51*(4), 387-402. Retrieved July 6, 2020, from www.jstor.org/stable/26661842

Rosario, A., Sotgiu, F., De Valck, K., & Bijmolt, T. (2016). The Effect of Electronic Word of Mouth on Sales: A Meta-Analytic Review of Platform, Product, and Metric Factors. *Journal of Marketing Research, 53*(3), 297-318. Retrieved July 6, 2020, from www.jstor.org/stable/44134844

Nooralahzadeh, F., Arunachalam, V., & Chiru, C. G. (2013, May). 2012 Presidential Elections on Twitter--An Analysis of How the US and French Election were Reflected in Tweets. In *2013 19th International Conference on Control Systems and Computer Science* (pp. 240-246). IEEE.

Gamon, M. (2004). Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*(pp. 841-847).

Song, Z., & Xia, J. (2016). Spatial and Temporal Sentiment Analysis of Twitter data. In Capineri C., Haklay M., Huang H., Antoniou V., Kettunen J., Ostermann F., et al. (Eds.), *European Handbook of Crowdsourced Geographic Information* (pp. 205-222). London: Ubiquity Press. Retrieved July 6, 2020, from www.jstor.org/stable/j.ctv3t5r09.20

Lee, T., & Bradlow, E. (2011). Automated Marketing Research Using Online Customer Reviews. *Journal of Marketing Research,48*(5), 881-894. Retrieved July 6, 2020, from www.jstor.org/stable/23033526

Statistics Canada. (2015, May 6). *Census Profile Joliette*. Statcan.Gc.Ca; Statistics Canada. https://www12.statcan.gc.ca/census-recensement/2011/dp-pd/prof/details/page.cfm?Lang=E&Geo1=CSD&Code1=2461025&Geo2=PR&Code2=24&Data=Count&SearchText=Joliette&SearchType=Begins&SearchPR=01&B1=All&Custom=

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational linguistics*, *37*(2), 267-307.

Twitter. (2019). *API Docs*. Twitter.Com. https://developer.twitter.com/en/docs

Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. (2013). *An introduction to statistical learning : with applications in R*. New York :Springer

Stock, J. H., & Watson, M. W. (2007). *Introduction to econometrics*. Boston: Pearson/Addison Wesley.

8. Appendix

Table 7.1: List of emoticons used as search queries for training data collection

Positive Emoticons			**Negative Emoticons**		
:-)	:^)	:)	:L	:-/	>:/
;)	:-D	:o)	:S	>:[	:@
:]	:D	:3	:-(	:[	:-||
:c)	8-D	:>	=L	:<	:-[
=]	8D	8)	:-<	=\\	=/
=)	x-D	:}	>:(	:(	>.<
X-D	xD	XD	:'-(	:'(	:\\
=-D	=D	=-3	:-c	:c	:{
=3	:-))	:'-)	>:\\	;(	
:')	:*	:^*			
>:P	:-P	:P			
X-P	x-p	xp			
XP	:-p	:p			
=p	:-b	:b			
>:)	>;)	>:-)			
<3					