university of groningen / faculty of arts

Sentiment classification on Twitter: the impact of sarcasm

Youri Schuur

Bachelor thesis
Informatiekunde
Youri Schuur
s2748428
December 12, 2017

ABSTRACT

A large body of work has shown that accuracies in the 80-90ies can be achieved with sentiment classification on Twitter data. However, it is still really hard to achieve scores higher than these percentages. A common problem when facing sentiment classication is sarcasm. For this thesis, the impact of sarcasm on sentiment classification was evaluated. Ngrams were used as features to predict the sentiment of the tweet. A Logistic Regression classifier was tested on a generic dataset of Tweets and evaluated on a dataset that contained sarcastic tweets. The Logistic Regression classifier achieved an accuracy of 66% on the generic dataset and an accuracy of 62% on the dataset that contained sarcastic tweets. The difference between these scores can be explained as the impact of sarcasm on sentiment classification.

CONTENTS

PREFACE

In this chapter I would like to thank a few people that have helped me during the time I wrote this thesis. First of all, I would like to thank my supervisor Mrs. Plank for instructing me during the process of me writing a thesis and for giving me the feedback I needed to improve and finish my thesis. I would also like to thank an old classmate of me named Guido Cnossen. Together, we did a project for Information Retrieval last year and that project have really helped me writing my own thesis. Finally, I would like to thank the SemEval-2013 and SemEval-2014 task forces for collecting and providing their datasets which I could use for my own research. All in all, the process of writing a thesis was really educational and I am pretty satisfied with the end result.

1 INTRODUCTION

Social media platforms like Twitter are common places for people to express their feelings or to give their opinions. Tweets can therefore been seen as an online representation of a user. The quantity of information on Twitter grows every day. These tweets can give us valuable information about the author of the tweet (Kumari et al., 2015). Because of the unlimited information on social media platform nowadays, these social media platforms like Twitter gained a lot of popularity in scientific research over the last decade (Kumari et al., 2015). An example of such a research subject is sentiment analysis on the basis of tweets.

Sentiment analysis is the classification of tweets into either positive or negative (or neutral) sentiment. Nowadays, a lot research has been done on this subject and F-scores around the 80-90% percent have been achieved (Pang et al., 2008; Go et al., 2009). That means that there are classifiers which are very good at predicting the sentiment of the tweet. However, there is still room for improvement. A difficult problem that has not been addressed many times is the sarcasm problem. How is a machine going to predict correctly the polarity of the sentiment of a tweet while the author of the tweet means exactly the opposite of what he/she wrote? In this thesis the impact of sarcasm on sentiment analysis will be tested and evaluated. The goal of this thesis is to research what exactly the impact of sarcasm is on sentiment analysis. If we know what the impact of sarcasm is on sentiment analysis, maybe in the near future more research will be done this solve this complicated problem. The research question for this thesis is:

- What is the impact of sarcasm on sentiment analysis on Twitter?

To answer this research question multiple aspects within sentiment analysis will be handled within this thesis. In the second chapter, the background of this subject will be handled. Previous research on this matter will be discussed and I will refer to previous studies about sentiment analysis and about the problem of sarcasm in sentiment analysis that helped me creating a dataset and a classifier. In the third chapter the data and the material will be discussed. In the first section, I will explain how the data was collected. Subsequently, I will discuss how the data was annotated. In the final section of this chapter the processing of all the data will be explained. In chapter four the methods and features used for this research will be addressed. The results of my research will be evaluated and discussed in chapter five. Ultimately, in chapter 6, the research question will be answered and also future work will be covered.

2 BACKGROUND

In this chapter the background of sentiment analysis on Twitter will be addressed. This background consists of two parts. In the first section, previous research on sentiment analysis on Twitter will be discussed. In the second section I will talk about the SemEval datasets that I will be using for my research. In the final section the definition of sarcasm will be discussed.

2.1 PREVIOUS RESEARCH ON SENTIMENT ANALYSIS ON TWITTER

A lot of research has been done on the classification of sentiment over the last fifteen years, especially on movie reviews, product reviews and blogs (Pang and Lee). For sentiment analysis, multiple machine learning methods were used to classify the sentiment of the tweets (Pang et al., 2008; Go et al., 2009). For these methods, accuracy scores around the 80-90% were achieved (Pang et al., 2008; Go et al., 2009). Sarcasm has always been considered as a difficult problem in sentiment analysis and has not been addressed in too many studies. González-Ibánez et al. (2011) experimented with this problem. For their research, the data that was collected was divided in three categories; sarcastic tweets, positive tweets and negative tweets. They used a Support Vector Machine (SVM) with Sequential Minimal Optimization (SMO) Logistic Regression (LogR) to correctly predict the sentiment of a tweet. They experimented with various combinations of unigrams, dictionary based features, lexical features and pragmatic features. Their model was also trained with bigrams and trigrams, however, the results were not better than when only unigrams were used. Bamman and Smith (2015) also took a closer look at the sarcasm problem in their article. However, they chose a different approach in comparison to the article of González-Ibánez et al. (2011). In the article of Bamman and Smith (2015), they realized that the context is crucial if you are trying to detect sarcasm. "Sarcasm requires some shared knowledge between speaker and audience" (Bamman and Smith, 2015). They tried to solve this contextualized sarcasm problem by adding multiple features to the more general features like unigrams, bigrams and POS tags. For example, they added author features to look at historical tweets from the same author. Also profile information was retrieved like how many followers the author of the tweet had or in which timezone the author lived etc. An accuracy of 75.4% was achieved by only looking at the tweet-features. By adding the author features they achieved an accuracy score of 84.9%. For this thesis, the goal is not to achieve an as high as possible accuracy score on sentiment classification or to detect (contextualized) sarcasm on twitter. The goal for this thesis is to evaluate what the impact of sarcasm is on sentiment analysis on Twitter.

2.2 THE SEMEVAL CORPUS

In this section I will discuss the SemEval corpus. This corpus is important for this thesis. The datasets I will use for my research are provided by SemEval, and naturally, these datasets are also used for completing the SemEval-2013 task 2 and SemEval-2014 task 9. Therefore, I decided to take a closer look at these tasks.

SemEval (Semantic Evaluation) is an on going series of evaluations of computational semantic analysis systems. In this section I will discuss two tasks that the SemEval task force has completed. The goal of both of these tasks was to promote research that will lead to a better understanding of how sentiment is conveyed in Tweets and SMS messages (Nakov et al., 2013; Rosenthal et al., 2014).

First I will discuss SemEval-2013 task 2. This task was divided into two subtasks. The subject of subtask A was **Contextual Polarity Disambiguation**: "Given a message containing a marked instance of a word or a phrase, determine whether that instance is positive, negative or neutral in that context" ((Nakov et al., 2013). The subject of subtask B was **Message Polarity Classification**: "Given a message, decide whether it is of positive, negative, or neutral sentiment. For messages conveying both a positive and a negative sentiment, whichever is the stronger one was to be chosen" (Nakov et al., 2013).

For their research, the collected Twitter and SMS messages from 2013 by searching for keywords for porpular trends and created a Twitter-train, Twitter-dev and Twitter-test set. For the SMS messages only a test set was created. The tweets and SMS messages were classified as positive, negative or neutral. The datasets were annotated for sentiment by Mechanical Turk workers . In the SemEval article they explained how the data was annotated: "Each sentence was annotated by five Mechanical Turk workers (Turkers). In order to qualify for the hits, the Turker had to have an approval rate greater than 95% and have completed 50 approved hits. The Turker had to mark all the subjective words/phrases in the sentence by indicating their start and end positions and say whether each subjective word/phrase was positive, negative, or neutral (subtask A). They also had to indicate the overall polarity of the sentence(subtask B)" (Nakov et al., 2013).

For subtask A, 44 different teams participated and many classifiers were used. Most of the teams used a Support Vector Machine (SVM). The Maximun Entropy and Naive Bayes (NB) classifiers were also often used. A lot of different features were used like n-grams, word clusters, word shapes, POS tags etc. The average F1-measure for subtask A on the Twitter testset was 74.1%. For the SMS testset the average F1-score was 70.8%. For subtask B, 38 teams participated and the average F1-score for the Twitter test set was 53.7%. The F1-score was much lower for subtask B than for subtask A. Apparently subtask B was a more complicated task and logical reason for this decline could be that a message can contain parts expressing positive sentiment and parts expressing negative sentiment. The average F1-score for the SMS test set was 50.2%. The best F1-scores on both tasks and both test sets are (Nakov et al., 2013) :

- F1=88.93, NRC-Canada on subtask A, Twitter;
- F1=88.37, GU-MLT-LT on subtask A, SMS;
- F1=69.02, NRC-Canada on subtask B, Twitter;
- F1=68.46, NRC-Canada on subtask B, SMS.

The SemEval-2013 task force was satisfied with the the number of participants. They also were pleased with the results that the teams achieved on these subtasks. Furthermore, they were satisfied with the fact that they created multiple datasets which they have released for the communinity and they thought these datasets would be helpful for further research on sentiment analysis.

Now I will briefly discuss the SemEval-2014 task 9, which is almost the same as the SemEval task 2. For the SemEval task 9, the goals and the subtasks were exactly the same as for the SemEval task 2. However, they made a few additions. An test set containing twitter messages from 2014 was added to their corpus. Also, they were interested in the sarcasm problem and therefore, a sarcastic twitter test set was added to their corpus. Finally, a dataset that consisted of Live Journal messages was added. The data was annotated the same way as the data that was used for

the SemEval task 2. Sarcastic tweets were collected by searching for the hashtag #sarcasm.

Like the task from 2013, the most popular classifiers were Support Vector Machine (SVM), Naive Bayes (NB) and Maximum Entropy. Lots of the features mentioned above were also used for the SemEval task 9. The average F1-scores for both subtasks were not mentioned in the article. In the first table the F1-scores of the top 3 teams are shown on all test sets (Rosenthal et al., 2014).

Table 1: F1-scores from the top 3 teams on all test sets for subtask A

#	Team	Twitter-2013	SMS-2013	Twitter-2014	Sarcasm-2014	LiveJournal-2014
1	NRC-Canada	90.14	88.03	86.63	77.13	85.49
2	SentiKLUE	90.11	85.16	84.83	79.32	85.61
3	CMUQ-Hybrid	88.94	87.89	84.40	76.99	84.21

Overall, in SemEval-2014 task 9 they found similar trends as in the SemEval-2013 task 2. NRC-Canada, the winner of the SemEval task 2 for subtask A improved their score from 88.93% to 90.14% on the Twitter-2013 test set. Unfortunately, in the article it was not mentioned how this improvement was accomplished. In general, they saw consistent improvements on the 2013 testsets for subtask A. In the second table the F1-scores of the top 3 teams on all test sets are shown for subtask B (Rosenthal et al., 2014). Once again it is shown that subtask A was a much harder task than subtask B. Finally, they noted that the systems that performed best on the Twitter-2014 test set also performed best on the Twitter-2013 test set. They also found it interesting that the systems performed better on the Twitter-2013 test sets than on the Twitter-2014 test sets. Most likely, the teams have overfitted on the Twitter-2013 test set.

Table 2: F1-scores from the top 3 teams on all test sets for subtask B

#	Team	Twitter-2013	SMS-2013	Twitter-2014	Sarcasm-2014	LiveJournal-2014
1	TeamX	72.12	57.36	70.96	56.50	69.44
2	coooolll	70.40	67.68	70.14	46.66	72.90
3	RTRGO	69.10	67.51	69.95	47.09	72.20

2.3 THE DEFINITION OF SARCASM

Sarcasm is used often in the English language. The BBC even has its own website that is designed to teach non-native English speakers how to use and understand sarcasm (Maynard and Greenwood, 2014). There are a lot of definitions for the word sarcasm. For example the Oxford English Dictionary defines sarcasm as "a sharp, bitter, or cutting expression or remark; a bitter gibeor taunt." However, these days it is generally used to mean a statement when people "say the opposite of the truth, or the opposite of their true feelings in order to be funny or to make a point", as defined on the BBC sarcasm webpage mentioned above (Maynard and Greenwood, 2014). There is also often confusion between the concepts sarcasm and irony. Traditionally, the distinction between the two was that irony was indirect, whereas sarcasm was direct. However, nowadays it is really hard to make a distinction and therefore I do not make one between these two concepts in this thesis. In this thesis, I therefore define a sarcastic statement as one where the opposite meaning is intended (Maynard and Greenwood, 2014).

3 DATA AND MATERIAL

In this chapter of this project I will discuss the data that are used for this research. This chapter consists of three sections. In the first section I will describe what my data collection looks like and how it was obtained. Next, I will explain how the data was annotated in the second section. Finally, in the third section I will address how I processed the data.

3.1 COLLECTION

The data I used for this project was mostly collected from the SemEval-2013 task 2 and SemEval-2014 task 9 corpus. For more information about the SemEval corpus/task please refer to the second section of the chapter 'Background' about the SemEval corpus. The SemEval task force gathered a corpus of messages on a range of topics, including a mixture of entities (e.g., Gada, Steve Jobs), products(e.g., Kindle, Android) and events(e.g., Japan earthquake, NHL playoffs) (Nakov et al., 2013), which you could download with a script they provided. This was necessary because it is illegal to distribute tweets from others on the internet so you have to download the tweets yourself. Keywords and Twitter hashtags were used to identify messages relevant to the selected topic. The SemEval task force provided multiple datasets extracted from the corpus they created which I used to train, develop and test the classifier. They used all these datasets for their SemEval-2013 task 2 (Nakov et al., 2013; Rosenthal et al., 2014). I will discuss all these datasets in this section.

The train dataset which SemEval provided I obviously used for training purposes. This dataset consisted of 9.684 tweets all collected in 2013. Beside the train dataset, SemEval also provided a dataset which could be used for training or development. This dataset consisted of 1.654 tweets. I decided I would use this dataset to train the classifier since they also provided another development set. Thus, the complete training dataset contained 11.338 tweets. The other development dataset, which could not be used to train the classifier consisted of Twitter and SMS messages. Naturally, I needed the Twitter messages for my research. The development dataset consisted of 3.813 tweets. However, in order to really test the impact of sarcasm, I thought this was too big of a development dataset, therefore, I only used the first 1.500 tweets of the development dataset. SemEval also provided a test dataset. This dataset consisted of four types of data: Twitter messages, SMS messages, Live Journal messages and sarcastic twitter messages. For this thesis, only Twitter messages are needed therefore the SMS and Live Journal messages were deleted. The twitter test dataset consisted of 5.664 tweets. Just like with the development dataset, I only used the first 1.500 tweets of the test dataset for the same reason. Further, the sarcastic tweets were useful. Unfortunately, there were only 85 sarcastic tweets available. I also stumbled on another problem. The SemEval tweets were collected in 2013 (Nakov et al., 2013) and 2014 (Rosenthal et al., 2014). A couple of years later, a lot of tweets were no longer available. The tweets could have been deleted or the account could have been deleted. In section 3.3 of this chapter I will discuss how many tweets were lost and how these tweets were removed from the data.

To check what the impact was of sarcasm on sentiment analysis, more sarcastic tweets were needed. The LWP workspace on the computers at the University of Groningen (RUG) provides a corpus named twitter2_en that contains English tweets from 2015 en 2016. All tweets are stored in JSON files. However, not only the tweets were stored in a JSON objects but also lots of other information about the tweet that

I did not need, like the amount of followers of the author of the tweet. Naturally, I could have extracted only the tweet from a particular JSON object and ignore the other information. However, this corpus also provided a built-in Tool that would only display the author of the tweet and the tweet itself. This came very handy since only the tweet was needed for my research. Instead of extracting the tweets from the JSON objects, I decided to use this particular tool that was provided by the corpus twitter2_en. Thus, I collected a file containing the username of the author of the tweet and the tweet itself divided by a tab. Via a small algorithm I wrote I removed the username of the author of the tweet from the file and I was left with a file consisting of only the tweets. This was the file I needed for my research.

Previous research on sarcasm detection on Twitter has shown that sarcasm detection is not only difficult for machines but also for humans. This research found low agreement rates between human annotators at the task of judging the sarcasm of others' tweets; consequently, recent research exploits users' self-declarations of sarcasm in the form of sarcasm or sarcastic tags of their own tweets (González-Ibánez et al., 2011). Therefore, I decided to search for the hashtag #sarcasm. This way, we are sure that the tweet was meant to be sarcastic because the author of the tweet reported that is was meant sarcastic (González-Ibánez et al., 2011; Bamman and Smith, 2015; Maynard and Greenwood, 2014). I searched for all tweets in 2016 containing the hashtag #sarcasm and finally I collected 8.095 sarcastic tweets. I decided to annotate 1.415 sarcastic tweets from that collection by myself. Remember that 85 annotated sarcastic tweets were already provided by SemEval so in total the sarcastic corpus consisted of 1.500 tweets.

From this sarcastic corpus, 1.000 tweets were used for training, 250 tweets were used for development and 250 were used for testing. A sarcastic training dataset was created from which 1.000 non-sarcastic tweets were replaced by 1.000 sarcastic tweets. Also, a sarcastic development dataset was created from which 250 non-sarcastic tweets were replaced by 250 sarcastic tweets. Likewise, a sarcastic test set was created from which 250 non-sarcastic tweets were replaced by 250 sarcastic tweets. Finally, all six datasets that I needed were collected:

1. Non-sarcastic training dataset - consisting of 8.385 tweets
2. Non-sarcastic development dataset - consisting of 1.500 tweets
3. Non-sarcastic test dataset - consisting of 1.500 tweets
4. Sarcastic training dataset - consisting of 8.375 tweets (1,000 sarcastic)
5. Sarcastic development dataset - consisting of 1.500 tweets (250 sarcastic)
6. Sarcastic test dataset - consisting of 1.500 tweets (250 sarcastic)

3.2 ANNOTATION

The by SemEval provided datasets, which I discussed in the section 'Collection' above and in section 'SemEval corpus' in chapter 2, were all annotated in the same way. There are 3 labels that were used to classify the tweets. The tweet could be classified as positive, this meant that the polarity of the sentiment of the tweet was positive. The tweet could also be classified as negative, this meant that the polarity of the sentiment of the tweet was negative. If the tweet was not really expressing positive or negative sentiment, the tweet was classified as neutral. The tweets in the training dataset were classified as follows: 3097 positive, 1256 negative and 4032 neutral. The development dataset consisted of 664 positive tweets, 229 negative tweets and 607 neutral tweets. Finally the test dataset contained 646 positive tweets, 177 negative tweets and 677 neutral tweets. Below in table 3 are a few examples of tweets classified as positive, negative and neutral.

Table 3: Positive, Negative and Neutral tweet

Positive	Love that James Bond montage at the end of Super Sunday by Sky.
Negative	You know what, i don't give a shit about tomorrow YOLO.
Neutral	Just finished watching "The Vow" for the 5th time.

The sarcastic dataset, which I had to annotate myself, was a bit more complicated. For information about how I defined sarcasm, please refer section named 'The definition of sarcasm' in the chapter 'Background'. For the sarcastic corpus, 3 labels were used to classify the tweets. Every time a tweet expressed positive or negative polarity of sentiment, the polarity was reversed. Like the non-sarcastic datasets, if a tweet did not express any positive or negative sentiment, the sarcastic tweet was classified as neutral. The tweets in the sarcastic dataset were classified as follows: 232 positive, 695 negative and 573 neutral. Below in table 4 are a few examples of sarcastic tweets classified as positive, negative and neutral.

Table 4: Positive, Negative and Neutral sarcastic tweet

Positive	Winters in Alberta truly are terrible. #sarcasm
Negative	I just love sitting here weekend after weekend doing nothing nofriends #sarcasm
Neutral	Yes. I do wake up every morning and spend 2 hours curling my hair. #sarcasm

3.3 PROCESSING

Once the data was collected, I had to make sure the data was ready to be used for the classifier. Every line in the downloaded files from the SemEval webpage consisted of four parts. First there was an ID number (probably the ID of the user who wrote the tweet), this ID number was followed by another ID number (probably the ID of the tweet), next was the label of the tweet, thus positive, negative or neutral and in the final part the tweet was shown. These four parts were divided by tabs. Below in table 5 I have an example of one line of the downloaded training dataset.

Table 5: Format of one line from downloaded training dataset

264183816548130816	15140428	positive	Imaaa be the 1st Tongan hooper in the NBA #Fact

For this thesis I only needed the labels and the tweets. The two ID numbers were deleted from the file and that left us with a file containing only the label followed by a tweet divided by a tab. As I said in section 3.1, a lot of tweets were no longer available because the tweets or the account was hacked. If this was the case, the fourth part of the line would not show a tweet but it would show "Not Available". Unfortunately all these tweets could not be used for my research. Below in table 6 the loss of data (highlighted) is shown.

As you can see in table 6, a lot of data was lost. In total, approximately 25% of all data was lost. This was unfortunate for the the training data. However, for the development data and the test data it was not that big of a deal. Remember that for both the development data and the test data only the first 1.500 tweets were used. Thus, the first 1.500 tweets from the 2.872 available tweets from the development dataset were used to develop the system. Likewise, the first 1.500 tweets from the 4.312 available tweets from the test dataset were used to test the system.

Once the tweets that were not available were deleted, other noise like Retweets (RT), URL's and @users were filtered out (González-Ibánez et al., 2011). I wrote a small algorithm using among other things, regular expressions that would replace the noise by a single space. Later I would delete those spaces from the tweet and I wrote the labels and the processed tweets to a new file. I decided to keep

Table 6: Loss of data

	Amount of tweets	Available tweets	**Non-available tweets**
Train dataset	9.684	7.184	**2.500**
Dev dataset (used for training)	1.654	1.201	**443**
Dev dataset (used for development)	3.813	2.872	**941**
Test dataset	5.664	4.312	**1.352**
Total	20.815	15.574	**5.241**

the hashtags because often twitter users express their feelings by using hashtags (González-Ibánez et al., 2011). Below in table 7 you can see an example of how a tweet was processed.

Table 7: Example of a tweet being processed

Before	@JohnKid Sun flare by the Washington Monument #beautiful http://t.co/WbGjIcF1
After	Sun flare by the Washington Monument #beautiful

For the sarcastic corpus a slightly different approach was used to process the data. Because the tweets were collected from a corpus, they were already available. Every line is this corpus consisted of two parts; the username and the tweet. The username was unnecessary information, therefore, the username was deleted from every line. Just like with the SemEval datasets, the RT's, URL's and @users were deleted from every line. However, I made one addition, the #sarcasm was also deleted from every line.

4 METHOD

In this chapter I will discuss the methods that have helped me answering the research question. This chapter will consist of two parts. First, the features I used for this thesis will be addressed. In the second part I will explain which classifiers I chose to classify the tweets.

In this thesis I used two models that are often used for sentiment classification: a **Logistic Regression** (LogR) model and a **Support Vector Machine** (SVM) model (González-Ibánez et al., 2011). The tweets were classified as positive, negative or neutral. The structure of the model that we are using is implemented in the Sklearn/ Scikit-learn packages (Pedregosa et al., 2011) of python and works with a few preset parameters. For example the fit function from sklearn will fit the model according to the given training data.

For this research not many features were used. Like I pointed out in the introduction, the goal of this thesis was not to achieve an extremely high accuracy and F-score. The goal for this thesis is to evaluate what the impact is of sarcasm on Twitter sentiment analysis. The features that we are using in our model are word **ngrams** (unigrams, bigrams, trigrams) to look for a contigious sequences of one, two or three words that may influence a prediction (Go et al., 2009; Bamman and Smith, 2015). Stopwords were filtered out by a function NLTK provided. In many scientific sentiment analysis studies POS tags are also used as features. I did not use POS tags as features because I did not think these features would improve the models we were using (Go et al., 2009).

The crucial part of this thesis is the evaluation. The classifier will be evaluated by using the accuracy-, precision-, recall- and F-scores (harmonic mean of the precision and recall) against a baseline of 0.33 (33 percent). Without training, the tweets have 33% chance to be classified correctly, because I use three labels to classify the tweets (positive, negative and neutral). These scores are easily obtainableby a few implemented functions, accuracy-score and classification-report of the sklearn/ scikit- learn python packages. In addition to these scores I will show the 20 'most-informative-features'. This shows the way in which certain occurrences of words have influenced the prediction of sentiment. The 20 most-informative-features will be shown for every label and hopefully it shows us something about the impact of sarcasm on sentiment analysis. Finally, I will show some examples of the predictions of the tweets when the classifier is trained on the non-sarcastic dataset and tested on the sarcastic dataset. It will be interesting to see which tweets were labeled incorrectly by the classifier.

All the algorithms I created and all the SemEval data I used for this thesis are available at my github page at `https://github.com/YouriSchuur/BA_thesis_IK`.

5 RESULTS AND DISCUSSION

In this chapter the results of my research will be shown. To clarify and point out these results, many tables will be used. The classifier have been evaluated by using the accuracy-, precision-, recall- and F-scores (harmonic mean of the precision and recall) against a baseline of 0.33 (33 percent). Also the most informative features per label will be addressed in this chapter. Finally, the predictions of the sarcastic tweets will be evaluated and the correlation between the predictions and the most informative features will be discussed. The evaluation consisted of seven parts:

1. Test a Logistic Regression (LogR) classifier on non-sarcastic test set while training the model with the non-sarcastic train set
2. Test a Support Vector Machine (SVM) classifier on non-sarcastic test set while training the model with the non-sarcastic train set, evaluate if there are any differences between the LogR- and SVM classifier.
3. Test classifier (Logistic Regression) on sarcastic test set while training the model with the non-sarcastic train set and compare with test 1.
4. Test classifier (Logistic Regression) on sarcastic test set while training the model with the sarcastic train set and compare with tests 1 & 3.
5. Evaluate the most informative features per label.
6. Evaluate if there are any differences between the most informative features per label for the sarcastic- and non-sarcastic test sets.
7. Evaluate the predictions of the tweets made by the classifier when trained on the non-sarcastic dataset and tested on the sarcastic dataset

First, the non-sarcastic was evaluated by testing the classifier (LogR) on the non-sarcastic test set by training the model with the non-sarcastic train set. For this test, an accuracy of 0.66 was achieved. Below in table 8 you are able to see more statistics from this test.

Table 8: Results when classifier was trained on non-sarcastic train set & tested on non-sarcastic test set

	Precision	Recall	F1-score	Support
Negative	0.67	0.27	0.38	177
Neutral	0.60	0.89	0.72	677
Positive	0.80	0.53	0.64	646
Average/total	0.69	0.66	0.64	1500

I managed to achieve an F1-score of 0.64, which I thought was pretty decent since the baseline was 0.33 and only words **ngrams** (unigrams, bigrams, trigrams) were used as features. The recall score of the as negative labeled tweets was very low. This has probably to do with the fact that there were only 1256 negative instances to train the classifier in comparison to the 3097 positive and the 4037 neutral instances. The classifier performed best on the neutral labels with an F1-score of 0.72. The recall for the neutral labels was decent with a score of 0.89. However, the table shows us that the precision scores for especially the positive labels but also the negative labels are superior to the precision score of the neutral labels. This is probably caused by the fact that positive and negative tweets often contain a few

particular words that express strong sentiment. The neutral tweets do not contain these particular words that express strong sentiment. This is probably the reason that the recall score of the negative labels is superior to the positive- and negative labels whereas the positive and negative labels have higher precision scores. Below in the confusion matrix (table 9), it is shown that many negative an positive tweets were labeled as neutral. Many positive and negative tweets did not contain words that express strong sentiment and that is the reason so many tweets were labeled as neutral.

Table 9: Confusion matrix: True values (vertical) & Predicted values (horizontal) when model is trained on non-sarcastic train set and tested on non-sarcastic test set

	Negative	Neutral	Positive
Negative	47	114	16
Neutral	4	602	71
Positive	19	285	342

For the second test, a Support Vector Machine (SVM) was implemented to see which classifier would perform better on the non-sarcastic test set. The results showed me the same trends and therefore I did not include all these tables in this thesis though I will briefly discuss them. The highest accuracy and average F1-score were achieved by the SVM by training the model with the non-sarcastic train set and testing the model on the non-sarcastic test set. The SVM even achieved a higher score than the Logistic Regression model. The SVM achieved an accuracy of 0.68 and the average F1-score was 0.66. Overall, the SVM classifier performed slightly better than the LogR classifier but the differences were negligible, therefore I decided to only use Logistic Regression for the rest of the tests.

For the third test, the classifier (LogR) will be tested on the sarcastic test test while it is being trained by the non-sarcastic train set. I expected that the scores for this this would be lower in comparison to test 1 because it is unlikely that the classifier would correctly predict the sentiment of the sarcastic tweets. Those expectations were correct, this test was completed with an accuracy score of 0.62. Below in the table more statistics from this test are shown.

Table 10: Results when classifier was trained on non-sarcastic train set & tested on sarcastic test set

	Precision	Recall	F1-score	Support
Negative	0.63	0.19	0.29	259
Neutral	0.58	0.88	0.70	653
Positive	0.68	0.51	0.59	588
Average/total	0.63	0.62	0.58	1500

We can see in the table that the statistics have decreased in comparison to to the first test. The accuracy has decreased by 4% and the average F1-score has decreased by 6%. This makes sense, since the classifier was trained by the non-sarcastic train set and it was tested on the sarcastic test set. Furthermore, the table shows the same trends that I encountered when I tested the classifier on the non-sarcastic test set. Notice that especially the scores for the negative labels decrease while the classifier is being trained by more negative instances. This decrease is caused by the sarcastic tweets in the test set. Most sarcastic tweets are negative and therefore, sarcastic tweets have especially an impact on the classification of the negative tweets. Below in the confusion matrix (table 11) you can see that only 48 negative tweets were correctly predicted as negative while 70 tweets were incorrectly predicted as positive. This shows the impact of sarcasm on sentiment analysis.

Table 11: Confusion matrix: True values (vertical) & Predicted values (horizontal) when model is trained on non-sarcastic train set and tested on sarcastic test set

	Negative	Neutral	Positive
Negative	48	141	**70**
Neutral	9	573	71
Positive	19	267	302

Lastly, the classifier was again tested on the sarcastic test set. However, this time the classifier was also trained on the sarcastic train set. I did not exactly know what to expect from this test. On the one hand, I thought the sarcastic tweets that were used for training would improve the scores because the system would probably correctly predict more sarcastic tweets from the test set. On the other hand, I thought the sarcastic tweets used for training would decrease the scores because it could mess up the classification of the non-sarcastic tweets from the test set. It turned out that both possible expectations were not correct. Test 3 was completed with an accuracy score of 0.62. Below in the table the other statistics from this test are shown.

Table 12: Results when classifier was trained on sarcastic train set & tested on sarcastic test set

	Precision	Recall	F1-score	Support
Negative	0.60	0.26	0.37	259
Neutral	0.58	0.88	0.70	653
Positive	0.71	0.48	0.58	588
Average/total	0.64	0.62	0.59	1500

This table, once again, shows us the same trends we have seen at the previous tests. However, there a few details I would like to discuss. The accuracy score for this test was approximately the same as for the previous test and the average F1-score improved by 1%. We could say there was not much of a change in scores. Nevertheless, the F1-score for the negative tweets was significantly higher. A logical reasoning could be that there were simply more negative tweets available for training. Most of the sarcastic tweets that were added to the sarcastic train set were labeled as negative because in general sarcasm if most often used in negative messages. For the rest, there were not any interesting statistics compared to the other tests.

Now that all three tests have been evaluated for both classifiers, I will discuss the most informative features of the non-sarcastic dataset. Why are these features interesting? Well, these features tell us a lot on how the tweets were being labeled. In table 10, I presented the 20 most informative features for all three labels. If a word was given a high score for a label, it meant that this particular word was often used in tweets that were classified as this label.

This table shows us that words like 'Great', 'Happy' and 'love' influenced the prediction of positive sentiment. This is logical because these words obviously express positive sentiment. Same goes for words like 'hate', 'bad' or 'sad'. These words influenced the prediction of negative sentiment. I was also satisfied with the fact that two different types of emoticons were present in the 20 most informative positive features and one emoticon was present in 20 most informative negative features. There was a time during my research that I wanted to filter out all punctuation. Luckily, I did not filter it out because these features show me that the emoticons can influence the prediction of sentiment strongly. I found the 20 most informative neutral features also quite interesting. Naturally, words that express positive or negative sentiment were not in this list. Instead, words like '2013', 'Saturday' and 'April' were present in this list. These types of words, are indeed often

Table 13: The features that influenced the prediction of the sentiment of a tweet the most for the non-sarcastic dataset

Most Informative Features (non-sarcastic dataset)	
Labels	Features
Positive	(2.7372 W:great), (2.6907 W:Happy), (2.6202 W:good), (2.5337 W:excited), (2.4991 W:love), (2.2758 W:best), (2.2137 W:Great), (2.1917 W:Good), (2.1909 W: **:)**), (2.1222 W:fun), (1.7923 W:amazing), (1.7121 W:happy), (1.6918 W:nice), (1.6444 W:wait), (1.6258 W:enjoy), (1.5491 W: **;)**), (1.5350 W:Love), (1.5275 W:awesome) (1.5103 W: **:D**), (1.5076 W:Thanks)
Negative	(2.6207 W: **:(**), (1.8522 W:bad), (1.8064 W:don't), (1.7257 W:shit), (1.5843 W:fuck), (1.5276 W:can't), (1.4965 W:hate), (1.4740 W:lost), (1.3223 W:sad), (1.3146 W:worst), (1.2923 W:won't), (1.2824 W:Why), (1.2714 W:worse), (1.2684 W:didn't), (1.2308 W:Sorry), (1.1415 W:cancelled), (1.1266 W:hell), (1.0441 W:kill), (1.0152 W:Niggas), (1.0046 W:ass)
Neutral	(1.0584 W:Who's), (0.9771 W:practice), (0.9511 W:Nov), (0.9437 W:For), (0.9407 W:Joe), (0.8948 W:Did), (0.8101 W:check), (0.7964 W:8th), (0.7917 W:score), (0.7728 W:vs.), (0.7603 W:2013), (0.7423 W:training), (0.7418 W:original), (0.7350 W:attend), (0.7327 W:April), (0.7183 W:second), (0.7180 W:Who's going), (0.7148 W:set), (0.7126 W:live), (0.7080 W:Saturday)

used in neutral messages. This shows us, once again, that these most informative features really influence the prediction of the sentiment

Now that we have seen the most informative features for the non-sarcastic dataset, I will be evaluating the differences between the top 20 most informative features for the non-sarcastic dataset (table 13) and the top 20 most informative features for the sarcastic dataset (table 14). In the top 20 most informative negative features I found something interesting. The most informative positive- and neutral features were almost the same as for the non-sarcastic dataset, therefore, only the table for the negative features will be shown.

Table 14: The features that influenced the prediction of the sentiment of a tweet the most for the sarcastic dataset

Most Informative Features (sarcastic-dataset)	
Labels	Features
Negative	(2.6508 W: :( ), (1.4882 W:don't), (1.4739 W:shit), (1.2749 W:bad), (1.2698 W:can't), (1.2590 W:fuck), (1.2544 W:won't), (**1.2220 W:fantastic**), (1.1347 W:didn't), (1.1174 W:hell), (1.1090 W:lost), (1.0559 W:hate), (1.0428 W:Really), (1.0378 W:worse), (1.0191 W:Sorry), (0.9703 W:cancelled), (**0.9657 W:Thanks**), (0.9515 W:: -( ), (0.9496 W:fucking), (0.9402 W:news,)

The interesting words are highlighted in this table. Those words are 'fantastic' and 'thanks'. These words normally express positive sentiment. However, they are also often used in a sarcastic way. This shows us why that the classifier is confused by the sarcastic tweets and that explains the decline in accuracy and F1-scores for the sarcastic test set.

Finally, the predictions of the sarcastic tweets will be discussed. To evaluate the predictions of the sarcastic tweets I will take a look at the predictions of the last 250 tweets of the sarcastic test set, since these 250 tweets are all sarcastic. Below in the table 15, six interesting examples of the predictions are shown.

Table 15: Examples of predictions of sarcastic tweets

#	**Tweet**	**Gold**	**Predicted**
1	it's this reason alone, I really **enjoy** watching the notebook.	Negative	Positive
2	Just **fantastic** how everything is arranged today	Negative	Positive
3	wall to wall people on the platform at South Norwalk waiting for the 8:08. **Thanks** for the Sat. Sched. Great sense	Negative	Positive
4	Work tomorrow is gonna be so **fun** #tired	Negative	Positive
5	Well, the Cards got their first loss out of the way. No more pressure at least. #STLCards	Neutral	Neutral
6	I'm offended! Why can't it just be Cheese? Kraft give me some money for my pain and suffering	Positive	Negative

The first four examples are great examples of how sarcasm is most often used. All these four examples contain words that express strong positive sentiment. However, because these sentences are all meant sarcastic, the sentiment of these tweets should be reversed. It is also interesting to see what the correlation is between these predictions of sarcastic tweets and the most informative features that have already been discussed in this chapter. It appears that there is a strong correlation between the predictions of the sarcastic tweets and the most informative features. All four examples contain a word that is present in the most informative positive features (enjoy, fantastic, Thanks & fun). These words are all highlighted in table 15. Examples 2 & 3 even contain words that we have seen in the most informative negative features for the sarcastic dataset (Thanks, fantastic).

Overall, the classifier found it much easier to classify the neutral instances correctly, the fifth tweet in table 15 is a good example. This tweets does not contain any words that express strong positive or negative sentiment. Therefore, the tweet is predicted as neutral. However, as we have seen in the confusion matrix in this chapter, many times positive and negative tweets were predicted as neutral. If this was the case, the tweet probably did not contain a particular word that expresses strong positive or negative sentiment.

The sixth tweet is an example of a tweet that expresses negative sentiment but is meant to be positive. I included this example because it is pretty rare that sarcastic tweets are meant to be positive. As we can see in the table, the classifier indeed classified this tweet as negative while the tweet was meant to be positive.

6 CONCLUSION

In this chapter I will answer the following research question:

- What is the impact of sarcasm on sentiment analysis on Twitter?

Limitations of this research will also be addressed and finally I will indicate directions for future work.

It is complicated to exactly measure the impact of sarcasm on sentiment analysis. For this thesis, I have come to the conclusion that the impact of sarcasm on sentiment analysis is reasonably small. How did I come to this conclusion? Well, in my research, the accuracy and average F1-scores declined by approximately 5%. Normally, for other sentiment analysis studies I think the impact would be smaller because I used many sarcastic tweets in the sarcastic datasets. I used so many sarcastic because I was interested in what it would do to the scores. In generic datasets for sentiment analysis, I doubt there are as many sarcastic tweets as I have used in my sarcastic datasets. In think the problem of sarcastic messages in sentiment analysis would only decline the scores by a few percentages. However, nowadays other sentiment analysis studies have shown that F1-scores around the 90% could be achieved. In that case, a decline of a few percentages is really unfortunate.

I also came to the conlusion thay the impact of sarcasm differs per label. For example, for the neutral and positive instances , it does not seem to have that big of an impact. In contrary, for the negative instances, the impact of sarcasm is quite big. This is logical, since most sarcastic tweets are meant to be negative. Because of the sarcastic tweets, the F1-score of the negative tweets even fell below the baseline score of 0.33. Thus, I can conclude that sarcasm in sentiment analysis has the biggest impact on the negative tweets.

The research I did had a few limitations. Like I already mentioned above. You could say that I used too many sarcastic tweets in the datasets. Nonetheless, I am still glad I used as many sarcastic tweets as I did because otherwise the decline of the scores could have been neglected since accuracy and F-scores tend to differ a little when the classifier is tested on different test sets. Another limitation is the lack of features. The ngrams as features worked decent but I think I could have achieved higher scores by using more features.

This bring us to the the question what further work could be done to solve the sarcasm problem in sentiment analysis. In my opinion, it will be very hard to completely solve this problem. Even humans have trouble detecting sarcasm in utterances and messages (González-Ibánez et al., 2011). The use of intensifiers used as features could be helpful for detecting sarcasm. Also author features should be addressed more in future sentiment analysis studies because collecting a lot of tweets from a person can give you an insight in what kind of person the author of the tweet is. Some people use lots of sarcasm in their messages and some people never use sarcasm in their messages. Author-Addressee features could also be helpful. If you are able to understand the relationship and find similarities of interest between the author and the addressee you are much more likely to detect sarcasm in their messages. Some of these features have already been studied and tested in scientific articles (Bamman and Smith, 2015). However, I think a lot more studies are needed to completely fix the sarcasm problem in sentiment analysis.

BIBLIOGRAPHY

Bamman, D. and N. A. Smith (2015). Contextualized sarcasm detection on twitter. In *ICWSM*, pp. 574–577.

Go, A., R. Bhayani, and L. Huang (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford 1*(2009), 12.

González-Ibánez, R., S. Muresan, and N. Wacholder (2011). Identifying sarcasm in twitter: a closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers-Volume 2*, pp. 581–586. Association for Computational Linguistics.

Kumari, P., S. Singh, D. More, D. Talpade, and M. Pathak (2015). Sentiment analysis of tweets. *International Journal of Science Technology & Engineering 1*(10), 130–134.

Maynard, D. and M. A. Greenwood (2014). Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *LREC*, pp. 4238–4243.

Nakov, P., S. Rosenthal, Z. Kozareva, V. Stoyanov, A. Ritter, and T. Wilson (2013). Semeval-2013 task 2: Sentiment analysis in twitter. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, Volume 2, pp. 312–320.

Pang, B., L. Lee, et al. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval 2*(1–2), 1–135.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12*, 2825–2830.

Rosenthal, S., P. Nakov, A. Ritter, and V. Stoyanov (2014). Semeval-2014 task 9: Sentiment analysis in twitter. In *In Proceedings of the Eighth International Workshop on Semantic Evaluation, SemEval'14*. Citeseer.