Twitter Sentiment Analysis using Linguistic Features, Emoticons and Emoji

Almazhan Kapan
New York University
aa5456@nyu.edu

Betty Kao
New York University
ytk244@nyu.edu

Kirill Dolgin
New York University
kd1881@nyu.edu

Abstract

SemEval has served as a leading forum in the field of sentiment analysis. Our system is based on the Twitter Sentiment Analysis task offered in SemEval 2017 and SemEval 2013 forums. The system uses rich linguistic, emoticon-emoji based features to detect positive, negative or neutral sentiment of a tweet. Our system includes features developed in earlier systems and new features created by our team. We introduce features based on adjective and adverb unigrams, emoticons and emoji and develop new lexicon based features based on popular sentiment lexicons. Based on these elements, we developed an SVM trained system that achieved a macro f-score of 64.9 and 63.8 on the SemEval 2013 and SemEval 2017 datasets respectively. These results would place our system in top 4 at SemEval 2013 and top 8 at SemEval 2017 forums.

1 Introduction

Sentiment analysis is a field that focuses on detecting a polarity of a natural language text at levels of words, phrases, sentences or documents (Hogenboom et al., 2015) and has applications for a variety of tasks ranging from brand management to political event prediction. Twitter, a global social networking tool, has become a popular source of sentiment data among researchers due to the increasingly large number of active users (more than 330 million users as of 2021) and accessibility of the data (Johnson, 2021).

The International Workshop on Semantic Evaluation (SemEval) is one of the leading forums that allows researchers to conduct sentiment analysis using Twitter data at different levels of granularity. This project is modelled based on the task 'Sentiment Analysis in Twitter', one of the most comprehensive tasks offered in SemEval forums from 2013 to 2017. For our model, we reference the latest forum when the task was run, i.e. SemEval 2017 Task 4 to take advantage of the most updated datasets and proposed solutions. Task 4 is a ternary polarity classification task with tweets classified as positive, negative or neutral.

Earlier research in sentiment analysis has treated natural language text as a 'bag-of-words', which allows to represent documents as vectors and enables using both supervised and unsupervised machine learning techniques to solve the task. While supervised solutions can achieve higher accuracy with more training, their classification results might not be intuitive and might not perform well on the non-trained data (Hogenboom et al., 2015). Thus, incorporating lexicons and linguistic features in supervised models, particularly, SVM, still remains very popular (Heerschop et al., 2011). SemEval 2017 Task 4 results also confirm this observation - several submissions ranked in top 10 used SVM models and incorporated various semantic, lexical and other linguistic features (Rosenthal et al., 2017).

This project focuses on sentiment analysis of tweets using SVM classifiers and rich linguistic features, building on the earlier approaches and additionally introduces new adjective and adverb unigrams, emoticon-emoji and lexicon based features. Empirical outcomes suggest that n-gram and lexicon based features are the most important features (Mohammad et al., 2013; Hagen et al., 2015). However, using all n-grams might cause overfitting and increase complexity of the model. In this study, we attempt to reduce the number of n-gram features (all unigrams, bigrams, trigrams) proposed in earlier models by introducing adjective and adverb unigrams. We also introduce new lexicon-based features based on the most popular lexicons adopted in earlier models.

However, one remaining challenge even for the model heavily based on lexicons and linguistic features is that a sentiment of a phrase depends not just on what words are used, but also on how these words are used (Bal et al., 2011). Even in face-to-face conversations, visual cues such as frowning or smiling can outweigh the impact of words in detecting a person's mood or attitude (Ferretti and Papaleo, 2019). Given this context, emoticons or emojis, which have been overlooked in earlier lexicon-based models, might be particularly useful in detecting a text sentiment. Hence,

we introduce new emoticon-emoji based features and our own emoticon-emoji lexicon.

2 Related Work

Generally, most models solving sentiment detection tasks utilize methods from machine learning, computational linguistics, text mining and statistics; often, several methods are used together (Feldman, 2013).

2.1 Lexicon-based features

(Hogenboom et al., 2015) propose that supervised learning sentiment analysis systems should also incorporate lexicon-based methods since the latter's reliance on rules allow to retain a more linguistic view on the data and undertake a deeper analysis. (Heerschop et al., 2011) further found that lexicon-based methods allow for more intuitive ways of accounting for structural or semantic aspects of natural language text. Lexicon-based methods usually use sentiment lexicons to extract polarity of words and later sum all scores to compute the overall text polarity (Hogenboom et al., 2015).

2.2 N-grams

(Kawamae, 2012) proposes using n-gram feature based models for sentiment analysis of the social media posts. The author outlines the following advantages of using n-grams: (i) they allow fuzzy and substring matching, which is a very important functionality for unstructured and open domains such as Twitter and (ii) n-grams are language-neutral and do not make any assumptions about the underlying language of use. (Mulki et al., 2017) present syntax-ignorant n-gram embeddings for the sentiment analysis of several Arabic dialects.

2.3 Part of Speech tags

(Badr and Fatima, 2015) evaluate the importance of part-of-speech tag features, unigrams and sparse phrasal features (bigrams and skipgrams) for the sentiment analysis task with SVM and MNB classifiers. They find that including POS tags to the unigrams consistently improves the performance of both classifiers compared to the unigrams baseline. They also discover that performance of the SVM system decreases if only sparse phrasal features are used (without unigrams) and conclude that combination of unigrams and bigrams produces the best outcomes, especially when the training dataset is limited.

2.4 Negation

(Wiegand et al., 2010) study the impact of negation on text polarity and outline different ways to take negation into account when developing a sentiment analysis system. They propose using polar expressions to account for negations, but also acknowledge the limitations of the approach. They describe an approach taken by (Pang and Lee, 2004) that incorporates negation in a bag-of-words model by creating artificial words: if a word yyy is preceded by a negative word (e.g. not, shouldn't, etc.), then a new feature NOT yyy is created. Every word is considered starting from negation element (e.g. not, dont) till punctuation (e.g. .,!); for ex., sentence 'I do not like these oranges' becomes 'I do not NOT like NOT these NOT oranges. (Wiegand et al., 2010) observe that although this approach, although not perfect, produces good results due to the usage of n-grams and the fact that larger texts might not contain many instances of negation.

2.5 Adjectives

Most of the research on adjectives and their polarity carrying features was performed by Wiebe (Hatzivassiloglou and McKeown, 1997; Wiebe et al., 2004; Riloff and Wiebe, 2003). These works displayed that adjectives provide the most meaningful cues of subjectivity, which were referred to as 'strongly subjective clues'. It was shown that adjectives' behavior, weight in sentiment texts and frequency are important features for sentiment analysis tasks and development of polarity lexicons (Riloff and Wiebe, 2003).

Similarly, (Turney and Littman, 2002) proposed a strategy to infer semantic orientation (sentiment) from semantic association of words with select positive and negative adjectives. In other words, how closely words are related to certain adjectives (e.g. good, nasty) determines their polarity. The model achieves an accuracy of 80% on the training corpus of one hundred billion words and highlights the importance of adjectives for sentiment detection. A similar idea of comparing words by their shared semantic properties is encountered in the study by (Sharma et al.) who assign intensity levels to adjectives and use it as a feature for the star-rating review sentiment detection task. Adjectives are generally described as the most 'sentiment carrying' parts of speech compared to nouns, verbs and others (Sirajzade et al., 2020).

2.6 Adverbs

(Dragut and Fellbaum, 2014) examine the effect of adverbs on the sentiment values of sentences and find that adverbs, although do not carry inherent sentiment polarity, still alter the degree of the polarity of words they modify. Similarly, (Kiritchenko and Mohammad, 2016) find that adverbs generally increase or decrease sentiment intensity of some words, but, in contrast to modals and negators, only few adverbs can reverse the

sentiment of the word. The study found that only the degree adverb 'less' affects sentiment of words to a larger extent (0.835 points and less on the scale from -1 to 1). Adverbs that do not indicate intensity or degree, along with verbs and adjectives were also identified as important features by (Chesley et al., 2006)

2.7 Emoticons and emoji

(Hogenboom et al., 2015) demonstrated that emoticons and emojis are more superior to textual cues since people specifically use the former to express, stress or disambiguate their sentiments. The authors outline steps for creating an Emoticon-Emoji lexicon, lexicon-based features and test results on a collection of Dutch tweets and English app reviews. They find that accounting for the sentiment conveyed by emoticons on paragraph and sentence levels significantly improves the performance of the sentiment detection system. Moreover, they discover that whenever emoticons or emojis are used, their associated sentiment dominates the sentiment expressed by the textual information and serve as a good approximation for the polarity of the text.

(Singh et al., 2019) suggested a non-lexicon method to incorporate emojis (can be extended to emoticons) in sentiment analysis tasks - replace emojis and emoticons with their textual descriptions. The authors trained their sentiment detection system on neural networks and provided the following reasons to use emoji descriptions: (i) descriptions are openly available; (ii) emojis and emoticons are common, but words are more common: meaning of word embeddings might be captured better than meaning of emoji embeddings; (iii) strategy is fast to implement and test. The authors obtain state of the art results in irony detection and sentiment analysis tasks.

(Felbo et al., 2017) also show that emojis can be often used to predict the emotional content of texts more accurately (e.g. for slang: 'This is shit' is classified as positive by the DeepMoji library proposed by the authors). The authors provide an online demo at deepmoji.mit.edu to allow other researchers to explore the predictions and functionality of their model.

2.8 Other features

Other features such as elongated words (Brody and Diakopoulos, 2011) , word embeddings (Tang et al., 2014), phonetic features (Ermakov and Ermakova, 2013). For sentiment detection tasks involving multiple languages, study of language variations (Volkova et al., 2013) and machine translation methods are used (Balahur and Turchi, 2013).

2.9 SemEval models

Since 2013, SemEval serves as a forum that collects the best-performing approaches for detecting tweet sentiment. Many state-of-the-art models have been proposed and among them, the system by (Mohammad et al., 2013) for the NRC Canada team is regarded as one of the best performing models that incorporate rich linguistic and lexicon features. The system ranked first in the SemEval 2013 Twitter Sentiment Analysis task and offered features such as Part of Speech tags, Negation, word N-grams, character N-grams, all-caps, hashtags, lexicons, clusters and others. This model was further developed for SemEval 2014 and was recreated by (Hagen et al., 2015) as a part of the twitter polarity classifier in SemEval 2015.

2.10 Our Approach

Based on the review of the literature above, the model outlined in (Mohammad et al., 2013) is closest to our model as it incorporates most of the features that we would like to include in our system. However, the NRC Canada model heavily relies on n-grams (trigrams, bigrams, unigrams) and the number of generated n-gram features becomes too large, which might cause overfitting and increase model complexity. Thus, in our project, we study how to reduce the number of generated n-gram features without significant loss in performance.

In addition, lexicon-based features proposed by the NRC Canada model can be further modified to improve the performance. Lastly, the NRC Canada model does not incorporate emojis and uses Christopher Pott's tokenizing script to detect and analyze emoticons, which currently might not be the state-of-the-art method to handle emoticons . In our project, we include emoji and emoticon handling features discussed in the literature above: (i) replace emojis and emoticons with their descriptions and (ii) develop sentiment lexicons for emojis and emoticons and use features.

Dataset		Pos	Neg	Neu	Total
SemEval'13	Train	3640	1458	4586	9684
	Dev	1475	559	1513	3547
	Test	575	340	739	1654
SemEval'17	Train	5214	2920	7438	15572
	Dev	1935	1322	2936	6193
	Test	705	1166	1700	3571

Table 1: Distribution of tweets across polarity groups for Training, Development and Testing groups for the SemEval 2013 and SemEval 2017 datasets

3 Data

The collected tweets are based on popular topics and events trending on Twitter using a Twitter related named entity recognition system (Ritter et al., 2011). Tweets for the test dataset had topics different from training dataset and were collected from later periods. The topics include geopolitical entities (e.g. country names), named entities (e.g. Barack Obama) and other entities ('Western media').

3.1 SemEval 2013 dataset

We use the datasets that were offered for the SemEval 2013 Task B since the model that we propose to enrich (Mohammad et al., 2013) was submitted for that task and we would like to compare our performance to that model. The collected tweets were retrieved using Twitter API and annotated on Mechanical Turk.

Table 1 illustrates the distribution of the SemEval 2013 tweets across training, development and testing sections and negative, positive and neutral polarity groups.

3.2 SemEval 2017 dataset

Additionally, we also test and train our model on the dataset from SemEval 2017 (latest SemEval forum to host 'Twitter Sentiment Analysis' task), so that we can evaluate performance of the model on the latest datasets. For the SemEval 2017, the organizers provided datasets from previous SemEval forums (from 2013 to 2016) for training and development and offered a new test dataset, which was annotated using CrowdFlower. Twitter API was used to download tweets and additional filtering applied e.g. only topics with more than 100 tweets remained and near-duplicates were removed (Rosenthal et al., 2017).

Table 2 illustrates how many tweets from earlier SemEval forums were included in the training, development and testing dataset for SemEval 2017.

Table 1 illustrates the distribution of the SemEval 2017 tweets across training, development and testing sections and negative, positive and neutral polarity groups.

Data	2013	2014	2015	2016	2017	total
Train	3308	458	958	1174	9761	15572
Dev	496	185	239	393	4880	6193
Test	-	-	-	-	-	3571

Table 2: Distribution of tweets (across prior year datasets) for Training, Development and Testing set at SemEval 2017

Emoticon	Emoji	Sent Score	Description
o/	👋	0.4543	WAVING HAND SIGN
</3	💔	-0.1331	BROKEN HEART
<3	💗	0.7227	GROWING HEART
8-D	😁	0.4939	GRINNING FACE WITH SMILING EYES
x-D	😁	0.4939	GRINNING FACE WITH SMILING EYES
:\-)	😂	0.2431	FACE WITH TEARS OF JOY
O:-)	😇	0.6457	SMILING FACE WITH HALO
3:)	😈	0.2915	SMILING FACE WITH HORNS
3:-)	😈	0.2915	SMILING FACE WITH HORNS

Figure 1: Sample entries for the Emoticon Emoji Mapping Sentiment Lexicon

3.3 Sarcastic Tweets dataset

To evaluate our hypothesis that emoticons and emoji can more effectively detect sentiment in tweets containing figurative language, we also test our model on the dataset of Sarcastic tweets retrieved from the Movie Review task on (Kaggle, 2015). In total, there are 968 tweets with 431 positive, 251 negative and 40 neutral tweets divided for training (500), development (240) and testing (228).

4 Methods

4.1 Emoji Sentiment Lexicon

In this project, we created an Emoji Sentiment lexicon that maps emojis to their sentiment scores. We used Twitter API to collect 5000 tweets containing emojis and then, with the Stanford Core NLP library, we computed sentiment scores of all tweets and converted scores to be in the range from -1 to 1 (to be consistent with other lexicon based features). Then, we mapped emoji to the sentiment of the tweet containing the emoji. Also, for each emoticon and emoji pair, we retrieved descriptions from the Unicode's full emoji list (Charts, 2021). See Figure 2 to view sample entries of the lexicon.

4.2 Emoticon to Emoji Sentiment Mapping Lexicon

We retrieved the most frequently used emoticons from Wikipedia page (2020), which contains the latest data for Unicode version 13.0. Using the Emoji Sentiment lexicon that we generated earlier, we map emoticon X to their corresponding emoji Y using Smile2Emoji library in the npm package (Ballarini, 2018). Then, we assign the sentiment score and description of the emoji Y from the Emoji Sentiment Lexicon to the emoticon X. Compared to the Emoji Sentiment Lexicon, this lexicon has fewer entries since several emoticons can be mapped to the same emoji and while every emoticon can be converted to an emoji, not every emoji has an equivalent emoticon (at least in the list of frequently

Emoji	Unicode	Sent score	Description
😠	0x1f620	-0.3289	ANGRY FACE
😧	0x1f627	-0.0693	ANGUISHED FACE
😲	0x1f632	-0.0748	ASTONISHED FACE
😺	0x1f431	0.5643	CAT FACE
😹	0x1f639	0.1551	CAT FACE WITH TEARS OF JOY

Figure 2: Sample entries for the Emoji Sentiment Lexicon

appearing emoticons). See Figure 1 to view sample entries of the lexicon.

4.3 Preprocessing

We tokenized tweets with TweetTokenizer from NLTK library as it allows detecting emoticons, hashtags, usernames. We removed stop words by using NLTK stopwords library for English and normalized all URLs. Each tweet was represented as a feature vector made up of the features defined in the next subsection. For every tweet in the training dataset, we extract feature vectors and then train in the SVM. Then, for every test tweet, we extract feature vectors and then predict the output using SVM.

4.4 Features

The submission code for NRC Canada is not publicly available, so we implemented all the features mentioned in (Mohammad et al., 2013) and our new proposed features in python3 using Jupyter notebook and sklearn library for SVM.

Word ngrams. Similarly to (Mohammad et al., 2013), we count presence or absence of contiguous sequences of 1,2 tokens as a feature. However, differently from the former, we further reduce n-gram features. We select and compare two sets of n-gram features: (i) unigrams with at least 5 and bigrams with at least 7 occurrences and (ii) unigrams with at least 3 and bigrams with at least occurrences. Even after reduction, generated n-gram features were very sparse. For the SemEval 2013 dataset, model (i) had 8194 features and model (ii) had 6436 features. For the SemEval 2017 dataset, model (i) had 11558 features and model (ii) had 8850 features. For every n-gram feature, we observe if the tweet contains the feature: if yes, we count number of occurrences; if not, we write 0.

Part of Speech We count the number of occurrences of each part of speech tag as a feature. We also use Carnegie Mellon University POS tagger since it allows us to detect Twitter specific tags such as hashtags, at mention, URL, emoticons. We generated 25 POS features for both SemEval 2013 and SemEval 2017 datasets.

Lexicon features Similarly to NRC Canada, we use the following lexicons to generate the lexicon based features:NRC-Emotion, NRC-Sentiment140, NRCHashtag (Mohammad et al., 2013), BingLiu (Hu and Liu, 2004) and MPQA (Wilson et al., 2005). A more detailed description of the lexicons is available in (Mohammad et al., 2013). Separate features were produced for unigrams, bigrams and results were compared. The following four lexicon based features were reproduced from the NRC Canada model: (i) total count of tokens in the tweet with positive score, (ii) total score of the tweet, (iii) max score in tweet and (iv) score of the last positive token in tweet (Mohammad et al., 2013).

In addition to these features, we introduce four new features that mirror the original features: (v) total count of tokens with negative score, (vi) total sum of negative scores and (vii) minimum score in tweet, (viii) total count of neutral (none) tokens. See Figure-fig:code We test the system using the original 4-feature set and new 8-feature set for both SemEval 2013 and SemEval 2017 datasets. Figure 3 illustrates how the features for the 8 feature set were computed for unigrams. Following earlier discussion on the importance of degrees in words, for the MPQA Lexicon, we also introduce two additional features (10 features in total): total count of tokens labelled as 'strong subjective' and 'weak subjective'.

Negation. The number of negated contexts is used as a feature. We adopt the method outlined by (Pang and Lee, 2004) discussed above and define negated context as a segment of tweet that starts with one of the negation words (e.g. not, don't etc.) and ends with a punctuation mark. A negated context has effect on the ngram features: NEG suffix is appended to each word between negation word and punctuation.

Adjectives. We create a dictionary of adjectives and count presence or absence of adjectives in a tweet as a feature. In contrast to n-grams features, we do not apply feature reduction for adjective and adverb unigrams since the number of generated features is rather modest (8-10 times smaller than the number of n-gram features). There are 1663 adjective features generated for the SemEval 2013 dataset and 2561 features for the SemEval 2017 dataset.

Adverbs. Similarly to adverbs, we create a dictionary of adverbs and count presence or absence of adverbs in a tweet as a feature. There are 450 adverb features generated for the SemEval 2013 dataset and 688 features for the SemEval 2017 dataset.

Emoticons and Emoji. We adopt two approaches towards handling emojis and emoticons:

1. We replace emoticons and emojis with their Unicode descriptions using the Emoticon-Emoji Mapping Sentiment lexicon that we created.

```python
def polarity(n_gram):
    score = word_dict[n_gram]
    if score > 0:
        return 'positive'
    if score < 0:
        return 'negative'
    else:
        return 'none'
```

```python
def count_polarity_tokens(string, tokenizer):
    scorelist = []
    for token in tokenizer.tokenize(string):
        token = token.lower()
        score = polarity(token)
        score_list.append(score)

    return dict(Counter(scorelist))
```

```python
def last_token(string, tokenizer):

    negList = []
    posList = []
    for token in reversed(tokenizer.tokenize(string)):
        token = token.lower()
        if polarity(token) == ('positive'|'negative') :
            return {'last_polarity' : word_dict[token]}
        else:
            continue

    return {'last_polarity' : 0}
```

```python
def max_token(string, tokenizer):

    negList = []
    posList = []

    for token in tokenizer.tokenize(string):
        token = token.lower()
        if polarity(token) == 'positive':
            pos_list.append(word_dict[token])
        elif polarity(token) == 'negative':
            neg_list.append(word_dict[token])


    try:
        pos_max = max(pos_list)
    except ValueError:
        pos_max = 0
    try:
        neg_max = min(neg_list)
    except ValueError:
        neg_max = 0

    return {'pos_max' : pos_max, 'neg_max' : neg_max}
```

```python
def all_feats_dict(string, tokenizer):
    ct = count_tokens_with_polarity(string, tokenizer)
    pol = polarity_sum(string, tokenizer)
    max_tkn = max_token(string, tokenizer)
    last = last_token(string, tokenizer)

    complete = dict()
    for dictionary in [ct, pol, max_tkn, last]:
        complete.update(dictionary)
    return complete
```

```python
t = all_feats_dict(only_tweet_train_data[0], tweet_tokenizer)
t
```

```
{'none': 4,
 'negative': 9,
 'positive': 9,
 'pos_sum': 3.9839999999999995,
 'neg_sum': 1.682,
 'pos_max': 1.607,
 'neg_max': -0.878,
 'last_polarity': 0.449}
```

Figure 3: Excerpt from code that calculates Lexicon-based features (8 feature set)

2. We use our Emoticon Emoji Mapping Sentiment Lexicon and Emoji Sentiment Lexicon to retrieve polarity scores for each emoticon and emoji. Based on these polarity scores, we create 8 lexicon features following the same approach as for other lexicon features.

5 Evaluation

We follow evaluation guidelines defined in SemEval 2017 Task 4 with slight modifications. The system proposed (Mohammad et al., 2013) competed at SemEval 2013 where the 'macro f-score' was the main evaluation metric. Since our goal is to enrich the mentioned system and compare results, we use the 'macro f-score' as the primary measure for our project too. The 'macro f-score' is not the primary metric for evaluation at SemEval 2017, however, the value is still computed for all top team submissions, which allows us to evaluate our model in comparison to top scoring SemEval 2017 systems as well.

Macro f-score is denoted as F_1^{PN} and will be computed as below:

$$F_1^{PN} = (F_1^P + F_1^N)$$

F_1^P refers to the f-score for the Positive class of tweets and F_1^N refers to the f-score for the Negative class of tweets. F_1^P will be computed as below:

$$F_1^{PN} = (2prec^P * recall^P)/(prec^P + recall^P)$$

In this formula $prec^P$ and $recall^P$ denote precision and recall for the Positive class. More details are available in the task description (Nakov et al., 2016)

6 Experiments and Results

6.1 Classifiers

We train a Support Vector Machine (SVM) with SVC classifier as specified in (Mohammad et al., 2013) to compare the effectiveness of our new features. However, for a sample of tweets, we also trained our model on other classifiers, particularly, KNeighborsClassifier (n neighbors=3,5), DecisionTreeClassifier (random state=3) and GradientBoostingClassifier (random state=3). For training only lexicon-based features, GradientBoosting classifier showed a better performance than the SVC classifier.

As in (Mohammad et al., 2013), we selected a linear kernel and the value C=0.005 for the parameter. However, we also used the StandardScaler library to determine the best value for C and experiment with C=0.005, 1, 2 for individual sets of features. C=1 performs better than C=0.005 for adjective, adverb and n-gram features.

6.2 Resampling

Since in both SemEval 2013 and SemEval 2017 datasets, there are significantly fewer negative and positive tweets than neutral tweets, we have applied resampling with the sample weight parameter from sklearn library. We chose a resampling ratio based on the approximate ratio of negative, positive and neutral tweets.

6.3 Results and Discussion

We trained our model in two stages: on SemEval 2013 datasets (9684 annotated tweets) and on SemEval 2017 datasets (15572 annotated tweets). We apply 5-fold cross validation and later run the model on the test sets for SemEval 2013 and 2017. We use a majority classifier as a baseline model - always select the most frequent class (between positive and negative) as output (f-score 29.5). The positive group was the most frequent class. We also select unigrams with at least 3 occurrences as a baseline (f-score 40.3). Compared to the baseline results in (Mohammad et al., 2013), our baseline model with reduced unigrams performed better. The system proposed by our model (SVM and all features) outperforms both baseline models and achieves a macro f-score of 64.9 and 63.8 for SemEval 2013 and SemEval 2017 datasets respectively. With these results for the Twitter Sentiment Analysis task, we would be ranked in top 4 at SemEval 2013 (Nakov et al., 2013) and top 8 list for SemEval 2017 (Rosenthal et al., 2017).

We perform our experiments in the following way (not exact order):

1.We determine the best method for handling lexicon-based features: using only original 4 features proposed in (Mohammad et al., 2013) or using 8 features that include both original and new features proposed in our system.

2. We also determine the best method for handling emoticons and emojis: using emoji and emoticon descriptions or using lexicon features based on the Emoji Sentiment and Emoticon-Emoji Mapping Sentiment Lexicons proposed in this project.

To execute steps 1 and 2, we hold POS, Negation features constant and test four combinations of emoticon-emoji and lexicon feature handling methods (e.g. 4 features and descriptions vs 8 features and lexicon features etc.). We run these combinations with all n-grams (unigrams, bigrams), adjective, adverb, combined (adjective and adverb) unigrams (separately i.e. run only with n-grams or only with adverbs). Table 3 illustrates the results of steps 1 and 2. For n-gram features, we select unigrams with at least 3 occurrences and bigrams with at least 5 occurrences.

	Experiment	All n-grams (unigram + bigram)	Adjective unigrams	Adverb unigrams	Adjective + Adverb unigrams
All features	POS+Negation+Emo (**descriptions**) + **4** feature set for lexicon based features	62.9	59.6	54.8	61.1
All features	POS+Negation+Emo (**descriptions**) + **8** feature set for lexicon based features	**64.7**	61.5	56.3	63.0
All features	POS+Negation+Emo (**lexicons**) + **4** feature set for lexicon based features	63.1	59.9	55.0	62.8
All features	POS+Negation+Emo (**lexicons**) + **8** feature set for lexicon based features	**64.9**	63.3	56.9	**63.3**

Table 3: Performance of various methods that handle emoticon-emojis and lexicon based features (step 1 and 2)

We found that adjective and adverb unigram features perform better when they are combined together. A method that replaces emoticons and emojis with their descriptions performs worse than the method that uses lexicon features based on our Emoji Sentiment and Emoticon-Emoji Mapping Sentiment lexicons. Moreover, the lexicon-based feature sets with 8 features perform better than sets with 4 features for all lexicons, including our newly created Emoticon-Emoji lexicons.

Particularly, experiments show that combination of (i) lexicon-based features with 8 features, (ii) emoticon-emoji features based on sentiment lexicons and (iii) all n-grams achieves the best performance: f-score of 64.9.

The second best result is achieved by combining the same (i) and (iii) features, but using descriptions instead of lexicon based emoticon-emoji features. This shows that n-grams and non-Emoticon-Emoji lexicon features (BingLiu, Sentiment 140 etc.). have more impact on performance than emoticons and emojis (regardless of whether description or lexicon method is used to handle the latter).

The adjective and adverb unigrams combined with (i) and (ii) achieve the third best result. This shows that the combination of all n-grams (bigrams and unigrams) outperforms adjective, adverb unigrams (and their combination). However, the difference in performance is not too large (1.8 points), while the number of features decreases almost by 5 times when we use adjective and adverb unigrams instead of all n-grams. In other words, the small decrease in performance from using adjective and adverb unigrams might be outweighed by advantages from having a less complicated model and faster training and testing time with fewer features.

Similar results were achieved for the SemEval 2017 dataset: the combination of all n-grams, lexicon-based features with 8 features and emoticon-emoji features based on lexicons achieved the highest performance with an f-score of 63.8.

From the experiments in step 1 and step 2, we determined that using Emoji-Emoticon features (based on lexicons) and using 8 feature set lexicon features (for all Sentiment lexicons) achieve the best performance.

3. Next, we evaluate performance of other features by removing sets of features one at a time and holding emoticon-emoji features and number of lexicon-based features (8) constant (since we already tested their performance in step 1 and step 2 above). Table 4 illustrates the experiment results. We can view that removing sentiment lexicons causes the largest drop in performance: -9.3 points for all n-gram based model and -10.4 points for adjective and adverb unigrams model. Among the lexicons (not in the table), removing NRC Hashtag and Sentiment 140 lexicons compared causes a larger drop in performance compared to other lexicons. These results are similar to the outcomes in (Mohammad et al., 2013). The second largest drop in performance is caused by removing n-grams (both bigrams and adj/adv unigrams). Removal of Emoticon and Emoji features causes the third largest drop in performance.

Similar results are achieved for the SemEval 2017 dataset: the largest drop in performance is caused by removing lexicons and n-grams (both all n-grams

and adj/adv unigrams). The removal of emoticons and emoji based features for SemEval 2017 causes a larger performance drop of 3.2 points, probably due to the fact that the 2017 dataset contains more emoji data than the 2013 dataset, therefore Emoji-Emoticon features are more important.

Also, we find that for sarcastic tweets, impact of all n-grams (unigrams, bigrams) tends to dominate over other features. Emoticon and emoji features based on lexicons cause the third largest performance drop for sarcastic tweets.

6.4 Future Work

For a more complex version of our system, the Negation and Adjective features can be further developed. Moreover, new unsupervised training methods can be used to test the impact of Adjectives, Adverbs, Emoticons and Emoji based features. Also, other classifiers and kernel values can be applied to further improve the performance.

Negation based features. (Zhu et al., 2014) propose to incorporate negation modeling into lexicons used in the system proposed by (Mohammad et al., 2013). In particular, a lexicon-based approach is proposed to determine the sentiment of words in negated contexts by creating separate lexicons for the negative and non-negative contexts. The training dataset is divided into two parts: Affirmative Context Corpus and Negated Context Corpus. This approach, although might result in a better performance, requires significantly more time due to the required re-engineering of the existing features (affected by negation), datasets and all the lexicons used in the system. For now, due to time limitation, we decided not to test this approach.

Adjectives (Sharma et al.) present a semi-supervised approach to assign intensity levels to adjectives (high, medium, low) and use this information as a feature. This feature potentially can lead to a better performance as it is not fully corpus dependent - adjectives can be assigned intensity levels even if they are absent in the corpus. The system has shown improvement from baseline models in the sentiment analysis task for a movie review dataset.

Semi-supervised and unsupervised models. The top-scoring teams in SemEval 2017 Task 4 used semi-supervised and unsupervised machine learning techniques (Rosenthal et al., 2017). Therefore, a suggestion for future work would be to trthe ain the given data with a model based on neural networks, in particular, RNN with attention networks. Compared to CNN, RNN is generally more preferable for NLP tasks since the former does not retain information about the word order.

Emoticon and Emoji lexicon. Another suggestion for future work would be to create a separate Emoticon Sentiment Lexicon where sentiment scores of emoticons will not be dependent on the scores of their corresponding emojis (no mapping). For example, similarly to Emoji Sentiment lexicon, retrieve sentiment scores of emoticons by (i) using a sentiment detecting library on tweets containing emoticons and (ii) later mapping sentiment of the tweets to emoticons. One difficulty of this approach is that on many social media platforms, emoticons are automatically converted to emoji once the text is posted/sent. Therefore, in order to retrieve the tweets that contain solely emoticons, we need to gather tweets for earlier dates and potentially manually filter them.

Classifiers. Since the GradientBoosting classifier performed better when lexicon based features were tested (the most important set of features), we suspect that using this classifier instead of SVC would achieve better performance for both SemEval 2013 and SemEval 2017. For now we primarily used SVC classifier, so that we can more accurately compare our performance to the performance of the model in (Mohammad et al., 2013). Training and testing the model with a new classifier might take more time.

7 Conclusion

In our experiments, we find that n-grams are still more important to the performance of the model (larger drop in f-score when they are removed) than adjective and adverb (and their combination). This outcome is consistent with the earlier research (Badr and Fatima, 2015). However, the performance difference is not as large when adjective and adverb unigrams are used together, while the number of generated features decreases almost by 5-6 times for both SemEval 2013 and SemEval 2017 datasets, leading to a less complicated model.

We find that non-Emoticon-Emoji lexicon features (e.g. Bing Liu, Sentiment140 etc.) are the most important features for the overall performance of our model. In particular, 8 feature set of lexicon features (4 features from (Mohammad et al., 2013) and 4 new features created in our system) seem to affect the performance of the model the most.

The emoticon and emoji features also improve the performance of the model, however, they are not the most important features. This can possibly be explained by the observation that n-grams, adjectives, adverbs appear more often in tweets than emojis or emoticons. Our model performs worse than the model in (Mohammad et al., 2013) for the SemEval 2013. Nevertheless, we still successfully confirmed our main assumptions:

	Experiment	All n-grams (unigram + bigram)	Adjective unigrams	Adverb unigrams	Adj + Adv unigrams	Other
All features	POS+Negation+Emo (lexicons) + 8 feature set for lexicon based features	**64.9**	64.1	57.6	64.6	
All features-no lexicon features	POS+Negation+Emo (descriptions)	55.6 (-9.3)	53.7 (-10.4)	46.9 (-10.7)	54.2 (-10.4)	-
All features-no emoticon and emoji features	POS+Negation+8 feature set for lexicon based features	62.8(-2.1)	61.9(-2.2)	55.4(-2.2)	62.5(-2.1)	-
All features-no n-grams (including adj, adv unigrams)	POS+Negation+Emo (lexicons)+8 feature set for lexicon based features	-	-	-	-	56.3 (-8.6)
All features-no POS	Negation+Emo (lexicons)+8 feature set for lexicon based features	64.3(-0.6)	63.4(-0.7)	56.9(-0.7)	64.0(-0.6)	-
All features-no Negation	POS+Emo (lexicons)+8 feature set for lexicon based features	63.4(-1.5)	62.4(-1.7)	56.0(-1.6)	63.0(-1.6)	-

Table 4: Performance of the system as individual sets of features are removed (step 3)

(i) adjective and adverb unigrams achieve almost as good performance as all n-grams model and bring a less complicated system (fewer features).

(ii) our new lexicon based features (8 features) do improve the performance compared to original 4 feature set.

(iii) emoticon-emoji features do contribute to the better performance of the model.

References

Badr Mohammed Badr and S. Sameen Fatima. 2015. Using skipgrams, bigrams, and part of speech features for sentiment classification of twitter messages. In *Proceedings of the 12th International Conference on Natural Language Processing*, pages 268–275, Trivandrum, India. NLP Association of India.

D. Bal, Malissa Bal, A. V. Bunningen, A. Hogenboom, F. Hogenboom, and F. Frasincar. 2011. Sentiment analysis with a multilingual pipeline. In *WISE*.

Alexandra Balahur and Marco Turchi. 2013. Improving sentiment analysis in twitter using multilingual machine translated data. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 49–55, Hissar, Bulgaria. INCOMA Ltd. Shoumen, BULGARIA.

A Ballarini. 2018. smile2emoji.

Samuel Brody and Nicholas Diakopoulos. 2011. Cooooooooooooooollllllllllll!!!!!!!!!!!!!! using word lengthening to detect sentiment in microblogs. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 562–570, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Unicode Charts. 2021. Unicode® emoji charts v13.1.

Paula Chesley, Bruce Vincent, Li Xu, and R. Srihari. 2006. Using verbs and adjectives to automatically classify blog sentiment. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*.

Eduard Dragut and Christiane Fellbaum. 2014. The role of adverbs in sentiment analysis. In *Proceedings of Frame Semantics in NLP: A Workshop in Honor of Chuck Fillmore (1929-2014)*, pages 38–41, Baltimore, MD, USA. Association for Computational Linguistics.

Sergei Ermakov and Liana Ermakova. 2013. Sentiment classification based on phonetic characteristics. In *Advances in Information Retrieval*, pages 706–709, Berlin, Heidelberg. Springer Berlin Heidelberg.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Proceedings*

of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 1615–1625, Copenhagen, Denmark. Association for Computational Linguistics.

Ronen Feldman. 2013. Techniques and applications for sentiment analysis. *Commun. ACM*, 56(4):82–89.

Valentina Ferretti and Francesco Papaleo. 2019. Understanding others: Emotion recognition in humans and other animals.

Matthias Hagen, Martin Potthast, Michel Büchner, and Benno Stein. 2015. Webis: An ensemble for twitter sentiment detection. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 582–589, Denver, Colorado. Association for Computational Linguistics.

Vasileios Hatzivassiloglou and Kathleen R. McKeown. 1997. Predicting the semantic orientation of adjectives. In *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 174–181, Madrid, Spain. Association for Computational Linguistics.

Bas Heerschop, Frank Goossen, Alexander Hogenboom, Flavius Frasincar, Uzay Kaymak, and Franciska de Jong. 2011. Polarity analysis of texts using discourse structure. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, page 1061–1070, New York, NY, USA. Association for Computing Machinery.

Alexander Hogenboom, Danella Bal, Flavius Frasincar, Malissa Bal, Franciska De Jong, and Uzay Kaymak. 2015. Exploiting emoticons in polarity classification of text. *J. Web Eng.*, 14(1–2):22–40.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, page 168–177, New York, NY, USA. Association for Computing Machinery.

Joseph Johnson. 2021. Internet users in the world 2021.

Kaggle. 2015. Sentiment analysis on movie reviews.

Noriaki Kawamae. 2012. Identifying sentiments over n-gram. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12 Companion, page 541–542, New York, NY, USA. Association for Computing Machinery.

Svetlana Kiritchenko and Saif Mohammad. 2016. The effect of negators, modals, and degree adverbs on sentiment composition. In *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 43–52, San Diego, California. Association for Computational Linguistics.

Saif Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. NRC-canada: Building the state-of-the-art in sentiment analysis of tweets. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 321–327, Atlanta, Georgia, USA. Association for Computational Linguistics.

Hala Mulki, Hatem Haddad, Mourad Gridach, and Ismail Babaoglu. 2017. Tw-StAR at SemEval-2017 task 4: Sentiment classification of Arabic tweets. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 664–669, Vancouver, Canada. Association for Computational Linguistics.

Preslav Nakov, Alan Ritter, Sara Rosenthal, Fabrizio Sebastiani, and Veselin Stoyanov. 2016. SemEval-2016 task 4: Sentiment analysis in twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1–18, San Diego, California. Association for Computational Linguistics.

Preslav Nakov, Sara Rosenthal, Zornitsa Kozareva, Veselin Stoyanov, Alan Ritter, and Theresa 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)*, pages 312–320, Atlanta, Georgia, USA. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, page 271–es, USA. Association for Computational Linguistics.

Ellen Riloff and Janyce Wiebe. 2003. Learning extraction patterns for subjective expressions. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 105–112.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. SemEval-2017 task 4: Sentiment analysis in twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518, Vancouver, Canada. Association for Computational Linguistics.

Raksha Sharma, Mohit Gupta, Astha Agarwal, and Pushpak Bhattacharyya. Adjective intensity and sentiment analysis.

Abhishek Singh, Eduardo Blanco, and Wei Jin. 2019. Incorporating emoji descriptions improves tweet classification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2096–2101, Minneapolis, Minnesota. Association for Computational Linguistics.

Joshgun Sirajzade, Daniela Gierschek, and Christoph Schommer. 2020. Component analysis of adjectives in Luxembourgish for detecting sentiments. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 159–166, Marseille, France. European Language Resources association.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1555–1565, Baltimore, Maryland. Association for Computational Linguistics.

Peter D. Turney and Michael L. Littman. 2002. Unsupervised learning of semantic orientation from a hundred-billion-word corpus. *CoRR*, cs.LG/0212012.

Svitlana Volkova, Theresa Wilson, and David Yarowsky. 2013. Exploring sentiment in social media: Bootstrapping subjectivity clues from multilingual twitter streams. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 505–510, Sofia, Bulgaria. Association for Computational Linguistics.

J. Wiebe, Theresa Wilson, Rebecca F. Bruce, Matthew Bell, and M. Martin. 2004. Learning subjective language. *Computational Linguistics*, 30:277–308.

Michael Wiegand, Alexandra Balahur, Benjamin Roth, Dietrich Klakow, and Andrés Montoyo. 2010. A survey on the role of negation in sentiment analysis. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, pages 60–68, Uppsala, Sweden. University of Antwerp.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 347–354, Vancouver, British Columbia, Canada. Association for Computational Linguistics.

Xiaodan Zhu, Svetlana Kiritchenko, and Saif Mohammad. 2014. NRC-canada-2014: Recent improvements in the sentiment analysis of tweets. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 443–447, Dublin, Ireland. Association for Computational Linguistics.