# Stance Classification of Tweets Using Skip Char Ngrams

Yaakov HaCohen-kerner(✉), Ziv Ido, and Ronen Ya'akobov

Department of Computer Science, Jerusalem College of Technology, 9116001 Jerusalem, Israel
kerner@jct.ac.il, ziv0798@gmail.com, ronenya4321@gmail.com

**Abstract.** In this research, we focus on automatic supervised stance classification of tweets. Given test datasets of tweets from five various topics, we try to classify the stance of the tweet authors as either in FAVOR of the target, AGAINST it, or NONE. We apply eight variants of seven supervised machine learning methods and three filtering methods using the WEKA platform. The macro-average results obtained by our algorithm are significantly better than the state-of-art results reported by the best macro-average results achieved in the SemEval 2016 Task 6-A for all the five released datasets. In contrast to the competitors of the SemEval 2016 Task 6-A, who did not use any char skip ngrams but rather used thousands of ngrams and hundreds of word embedding features, our algorithm uses a few tens of features mainly character-based features where most of them are skip char ngram features.

**Keywords:** Skip character ngrams · Skip word ngrams · Social data
Short texts · Stance classification · Supervised machine learning · Tweets

## 1 Introduction

Sentiment analysis is the computational study of people's opinions, appraisals, attitudes, and emotions toward entities, individuals, issues, events, topics and their attributes [1]. Stance classification is a sub-domain of sentiment analysis. Stance classification is defined as the task of automatically determining from text whether the text author is in favor of, against, or neutral towards the given target. This task is challenging due to the fact that the available social data contains on the one hand, informal language, e.g., emojis, hashtags, misspellings, onomatopoeia, replicated characters, and slang words and on the other hand, personalized language.

Stance detection is becoming more and more important in many fields. For instance, stance studies can be helpful in detecting electoral issues and understanding how public stance is shaped [2]. Furthermore, stance detection is critical in situations in which a quick detection is needed, such as disaster detection and violence detection [3].

During the last fourteen years, there has been active research concerning stance detection. Most studies focus on debates in online social and political public forums [4–7], congressional debates [8–10], and company-internal discussions [11, 12].

In this study, we explore another field, the field of stance detection in tweets. Twitter as one of the leading social networks presents challenges to the research community since tweets are short, informal, and contain many misspellings, shortenings, and slang


Y. Altun et al. (Eds.): ECML PKDD 2017, Part III, LNAI 10536, pp. 266–278, 2017.
https://doi.org/10.1007/978-3-319-71273-4_22

words. To perform the stance classification tasks we use the popular char/word unigrams/bigrams/trigrams features. Furthermore, we use hashtags, orthographic, and sentiment features that are assumed to contain important social information. We also use char/word skip ngram features.

Skip ngrams are more general than ngrams because their components (usually characters or words) need not be consecutive in the text under consideration, but may leave gaps that are skipped over [13]. The idea behind skip ngram features is to generate features that occur more frequently, which allow overcoming, at least partially, problems such as noise (e.g., misspellings) and sparse data (i.e., most of the data is fairly rare), by considering various skip steps. For the char sequence ABCDE, as an example, in addition to the traditional bigrams AB, BC, CD, and DE, we can define the following skip-bigrams with the skip step of "one": AC, BD, and CE. The main disadvantage of the skip ngram features (for various string and skip lengths) is that their number is relatively high.

The main contribution of this study is the implementation of successful stance classification tasks for short text corpora based mainly on a limited number of char ngrams features in general and char skip ngrams in particular. To the best of our knowledge, we are the first to perform such successful stance classification. The macro-average results obtained by our algorithm are significantly better than the best macro-average results achieved in the SemEval 2016 Task 6-A [14] for all the five released datasets of tweets in the supervised framework.

The rest of this paper is as follows: Sect. 2 presents relevant background on stance classification and skip ngrams. Section 3 describes the applied feature sets. Section 4 presents the examined corpus, the experimental results, and their analysis. Finally, Sect. 5 summarizes the research and suggests future directions.

## 2 Relevant Background

### 2.1 Stance Classification

A shared task held in NLPCC-ICCPOL 2016 [15] focuses on stance detection in Chinese microblogs. The submitted systems were expected to automatically determine whether the author of a Chinese microblog is in favor of the given target, against the given target, or whether neither inference is likely. The authors point that different from regular tasks on sentiment analysis, the microblog text may or may not contain the target of interest, and the opinion expressed may or may not be towards the target of interest. The supervised task, which detects stance towards five targets of interest, has had sixteen team participants. The highest F-score obtained was 0.7106.

The organizers of the SemEval 2016 Task 6-A [14] released five datasets of tweets in the supervised framework. The goal of this task was to classify stance towards five targets: "Atheism", "Climate Change is a Real Concern", "Feminist Movement", "Hillary Clinton", and "Legalization of Abortion" while taking into account that the targets may not explicitly occur in the text. This corpus is the corpus we used in this study. The best results achieved in this task will be compared to our results.

### 2.2 Skip Ngrams

Guthrie et al. [13] examine the use of skip-grams to overcome the data sparsity problem, which refers to the fact that language is a system of rare events, so varied and complex, that even using an extremely large corpus, we can never accurately model all possible strings of words. The authors examine skip-gram modelling using one to four skips with various amount of training data and test against similar documents as well as documents generated from a machine translation system. Their results demonstrate that skip-gram modelling can be more effective in covering trigrams than increasing the size of the training corpus.

Jans et al. [16] were the first to apply skip-grams to predict script events. Their models (1) identify representative event chains from a source text, (2) gather statistics from the event chains, and (3) choose ranking functions for predicting new script events. Predicting script events using 1-skip bigrams and 2-skip bigrams outperform using regular ngrams on various datasets. They estimate that the reason for these findings is that the skipgrams provide many more event pairs and by that better capture statistics about narrative event chains than regular ngrams do.

Sidorov et al. [17] introduce the concept of syntactic ngrams (sn-grams), which enables the use of syntactic information. In sn-grams, neighbors are defined by syntactic relations in syntactic trees. The authors perform experiments for an authorship attribution task (a corpus of 39 documents by three authors) using SVM, NB, and J48 for several profile sizes. The results show that the sn-gram technique outperforms the traditional word ngrams, POS tags, and character features. The best results (accuracy of 100%) were achieved by Sn-grams with the SVM classifier.

Fernández et al. [18] perform supervised sentiment analysis in Twitter. They show that employing skip-grams instead of single words or ngrams improves the results for five datasets including Twitter and SMS datasets. This fact suggests that the skip-grams approach is promising.

Dhondt et al. [19] improve the classification of abstracts from English patent texts using a combination of unigrams and PoS filtered skip-grams. Skip-grams with zero (bigrams) up to two skips were found to be efficient informative phrases and especially noun-noun and adjective-noun combinations make up the most important features for patent classification.

## 3 The Features

In this research, we implement 36,339 features divided into 18 feature sets. Some of these feature sets (e.g., quantitative and orthographic) have been already implemented in previous classification studies [20, 21]. Table 1 presents general details about these feature sets. In a case, where less features are found for a certain feature set than the number assigned to this set then this set contains the number of found features.

The hashtag set contains the following 105 features: frequencies of the top 100 occurring hastags normalized by the # of words in the tweet, # of hashtags in the tweet normalized by the # of the words in the tweet, # of occurrences of 27 positive NRC [22] sentiment words used in hashtags normalized by the # of the words in the tweet, # of

occurrences of 51 negative NRC sentiment words used in hashtags [22] normalized by the # of the words in the tweet, # of occurrences of 14,459 positive NRC words used in hashtags normalized by the # of the words in the tweet, and the # of occurrences of 27,812 negative NRC words used in hashtags normalized by the # of the words in the tweet.

**Table 1.** General details about the feature sets.

| # of feature set | Name of feature set | # of features | # of feature set | Name of feature set | # of features |
|---|---|---|---|---|---|
| 1 | hashtag | 105 | 10 | PoS Tags | 36 |
| 2 | sentiment | 6 | 11 | character unigrams | 1000 |
| 3 | quantitative | 5 | 12 | character bigrams | 1000 |
| 4 | emojis | 21 | 13 | character trigrams | 1000 |
| 5 | orthographic | 122 | 14 | word unigrams | 1000 |
| 6 | long words | 11 | 15 | word bigrams | 1000 |
| 7 | stop words | 11 | 16 | word trigrams | 1000 |
| 8 | onomatopoeia | 11 | 17 | skip character ngrams | 15000 |
| 9 | slang | 11 | 18 | skip word ngrams | 15000 |

The sentiment set contains the following 6 features: normalized count of positive/negative sentiment emotion words according to the NRC lexicon [22], normalized counts of positive/negative sentiment words according to the Bing–Liu lexicon [23], and normalized count of positive/negative sentiment words according to the MPQA lexicon [24].

The quantitative set contains the following 5 features: # of characters in the tweet, # of words in the tweet, the average length in characters of a word in the tweet, # of sentences in the tweet, and the average length in words of a sentence in the tweet.

The emoji set contains the following 21 features: the # of emojis in the tweet normalized by the # of the characters in the tweet and frequencies of the top 20 occurring emojis normalized by the # of words in the tweet.

The orthographic set contains 122 features. Due to space limitation, we shall present some of them as follows: # of question marks/# of exclamation marks in the tweet/# of pairs of apostrophes in the tweet/# of legitimate pairs of brackets normalized by the # of characters in the tweet.

The "long words" set contains the following 11 features: # of elongated words (i.e., words that at least one of their letters repeats more than 3 times) normalized by the # of the words in the tweet, and frequencies of the top 10 occurring long words (words that their length is more than 10 characters) normalized by the # of the words in the tweet.

The stop words set contains the following 11 features: # of stop words in the tweet normalized by the # of words in the tweet and frequencies of the top 10 occurring stop words normalized by the # of words in the tweet.

The onomatopoeia set contains the following 11 features: # of onomatopoeia words in the tweet normalized by the # of words in the tweet and frequencies of the top 10 occurring onomatopoeia words normalized by the # of words in the tweet.

The slang set contains the following 11 features: # of slang words in the tweet normalized by the # of words in the tweet and frequencies of the top 10 occurring slang words normalized by the # of words in the tweet.

The PoS Tags set contains frequencies of the 36 PoS tags (see the 'Penn Treebank Project' [25]) normalized by the # of PoS tags in the tweet implemented by the Stanford Log-linear Part-Of-Speech Tagger [26] described in Klein and Manning [27].

Each one of the character unigrams/bigrams/trigrams sets includes the frequencies of the top 1000 occurring character unigrams/bigrams/trigrams normalized by the suitable # of character series in the tweet. Each one of the word unigrams/bigrams/trigrams sets includes the frequencies of the top 1000 occurring word unigrams/bigrams/trigrams normalized by the suitable # of word series in the tweet.

The skip character n-grams set is divided into 15 feature subsets (and the same for the skip word n-grams set). The features included in these 30 sub-sets (1000 features for each sub-set) are defined for all possible combinations of continuous character/word series (of 3–7 characters/words) that enable skip steps (of 2–6 characters/words, respectively). We defined 30,000 features for these 30 sub-sets because we wanted to have 1000 features for each sub-set (similar to what was defined for the character/word unigrams/bigrams/trigrams sets). We did not know which combinations of character/word series and skips will be successful; therefore we decided to define 30 possible combinations. The main reason why we enabled such a big number of features is because we assume that some of these skip ngram features might be very useful to overcome problems that characterized tweets such as noise (e.g., misspellings) and sparse data (i.e., most of the data is fairly rare).

## 4 The Experimental Setup

The examined corpus is the corpus of the SemEval 2016 Task 6-A [14] mentioned above. It includes tweets divided to 5 datasets: Legalization of Abortion, Hillary Clinton, Feminist Movement, Climate Change, and Atheism. Each one of the topics contains tweets with stance class that be labeled into one of three possibilities: FAVOR, AGAINST and NONE. Table 2 presents the distribution of stances in the five supervised datasets. To enable reproducibility, in the next paragraphs, we detail the algorithm, the experiments and their results (in addition to the details in the previous section).

**Table 2.** Distribution of stances in the five supervised datasets.

| Target | # total | % of instances in train set | | | | % of instances in test set | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | # train | Favor | Against | Neither | # test | Favor | Against | Neither |
| Abortion | 933 | 653 | 18.5 | 54.4 | 27.1 | 280 | 16.4 | 67.5 | 16.1 |
| Climate | 564 | 395 | 53.7 | 3.8 | 42.5 | 169 | 72.8 | 6.5 | 20.7 |
| Feminist | 949 | 664 | 31.6 | 49.4 | 19.0 | 285 | 20.4 | 64.2 | 15.4 |
| Clinton | 984 | 689 | 17.1 | 57.0 | 25.8 | 295 | 15.3 | 58.3 | 26.4 |
| Atheism | 733 | 513 | 17.9 | 59.3 | 22.8 | 220 | 14.5 | 72.7 | 12.7 |
| All | 4163 | 2914 | 25.8 | 47.9 | 26.3 | 1249 | 24.3 | 57.3 | 18.4 |

Basic baseline accuracy results for each one of the five datasets are computed using all the features (more advanced baseline accuracy results are the state-of-art results reported by the best macro-average results achieved in the SemEval 2016 Task 6-A). We performed extensive experiments using the WEKA platform [28, 29]. Using the same training and test sets as used by the SemEval 2016 Task 6-A, we applied eight variants of seven supervised machine learning (ML) methods with their default parameters, parameter tuning, and 10-fold cross-validation tests, three filter feature selection methods, and seven performance metrics, as follows:

For each dataset (Atheism, Climate Change, Feminist Movement, Hillary Clinton, and Legalization of Abortion), we perform the following steps:

1. Compute all the features from the training dataset
2. Apply the eight variants of the seven supervised ML methods (SMO with two different kernels, LibSVM, J48, Random Forest (RF), Bayes Networks, Naïve Bayes (NB), and Simple Logistics) using all the features to measure the baseline accuracy results.
3. Filter out non-relevant features using three filtering methods (Info Gain [30], Chi–square, and the Correlation Feature Selection method (CFS) [31].
4. Re-apply the eight variants of the seven supervised ML methods using the filtered features for all the 3 filtering methods.
5. Compute the accuracy, precision, recall, and F-Measure values obtained by the top three ML methods while performing various types of parameter tuning, e.g. increasing the # of iterations in RF, performing two experiments for SMO with two different kernels, the default Poly-Kernel, and the normalized Poly-Kernel. We saw that changing the kernel type to these two specific kernels resulted in better results. For LibSVM, we changed the kernel to be the linear kernel, the C value to 0.5 instead value of 1, and we tuned the 'normalize' and 'probabilityEstimates' options.
6. Given the test data, we apply the best ML method (according to the accuracy results) on the features filtered-in by the CFS selection method (found as the best feature selection method), and compute the following seven performance metrics: accuracy, precision, recall, F-Measure, ROC area, PRC area, and the macro-average result.

Due to space limitations, we present in the following sub-section detailed results for only one of the datasets of Task 6-A of SemEval 2016: legalization of abortion. A summary of the results for all the five datasets will be presented after that.

### 4.1 Results for the Legalization of Abortion Dataset

We applied the ML methods described above on all the features and on the filtered features using the three filtering methods. The accuracy results of the baseline version and three versions using the filtered features (Info Gain, Chi–square, and CFS) for each tested ML method for the training dataset of the Legalization of Abortion are presented in Fig. 1.

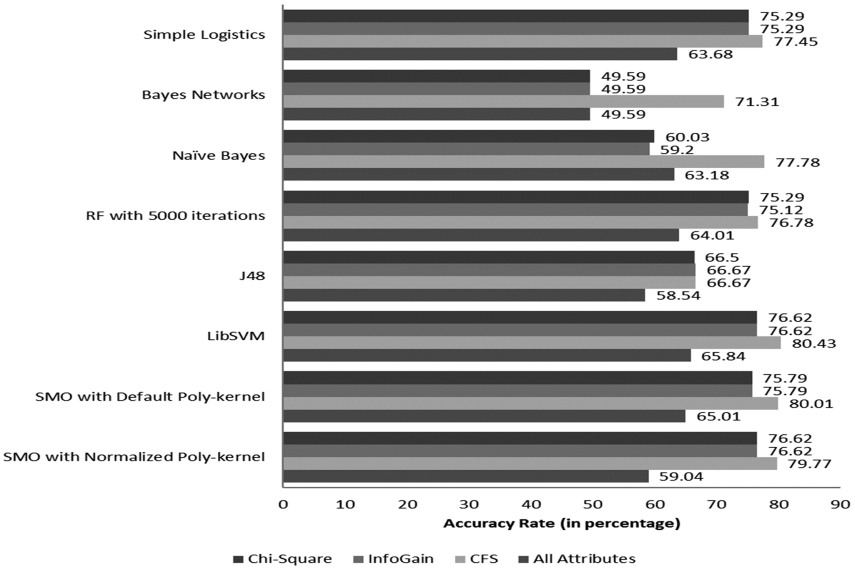


**Fig. 1.** Accuracy rates of the baseline and the filtered features for the abortion dataset.

From Fig. 1, we can see that for all ML methods, the best accuracy results are achieved using the CFS feature selection method. Moreover, in most cases the CFS results are significantly higher than the results obtained by the baseline version that uses all the features. We decided to perform additional experiments with the top three ML methods and to check other measures in addition to accuracy. In Fig. 2, we see the accuracy, precision, recall and F-Measure results of the top three ML methods.

The three accuracy results in Fig. 2 in descending order are: LibSVM with optimized setting (80.43%), SMO with the poly-kernel (80.01%), and SMO with the normalized poly-kernel (79.77%). We can see that the values of the other measures for these three ML methods are also rather similar. There are no significant differences between the results of the three ML methods. Nevertheless, the LibSVM ML method obtained the highest results for all the four measures. Using the filtered features, LibSVM achieved a 14.59% increase over the basic baseline.

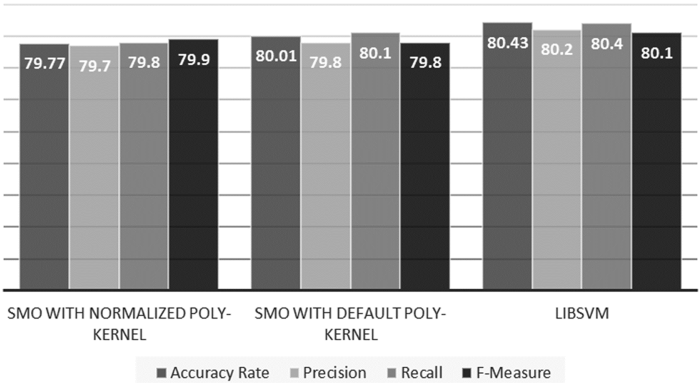


**Fig. 2.** Accuracy, precision, recall and F-measure results of the top three ML methods for the abortion dataset.

The application of the CFS feature selection method on all the features lead to a reduced set of 167 features. 125 features (81%) belong to the skip char ngram feature sets, 30 features (18%) are char ngrams (unigram, bigram and trigram) feature sets, and only 2 features are word unigrams.

**Test Data Results.** The test data for the Legalization of abortion dataset contains 280 tweets. 189 tweets (68%) are likely AGAINST the target, 46 tweets (16%) are likely in FAVOR of the target, and 45 (16%) are NONE of the above.

Based on the results obtained for the training test, we applied the CFS method (the best feature selection method) on the test data, and then we applied the LibSVM method with optimized parameters (the best ML method). The application of the CFS method on all the features lead to a reduced set of 100 features. Again, the dominant feature sets are the feature sets that belong to the char skip ngram features, with 77% of the features. Moreover, almost all the selected features (93%) are character-based features (char skip ngrams, char unigrams, char bigrams, and char trigrams).

The application of the LibSVM method with optimized parameters on the 100 filtered features lead to the following results: accuracy (86.43), Precision (86.2), Recall (86.4), F-Measure (86.3), ROC area (0.93), and PRC area (0.91). The values of the ROC area and the PRC area indicate excellent classification performance.

To estimate the relative importance of each feature, we further applied the InfoGain feature selection method on the 100 filtered features. Analysis of the top 25 ranked features showed that 24 features are character-based. Of these 24 features, 18 features are skip char ngram features (9 skip char bigram features, 8 skip char trigram features, and 1 skip char quadgram feature), and 6 features are from the char ngram feature sets (3 char bigrams and 3 char trigrams). Only one feature is a word unigram.

Examples for a few of those top 25 ranked features are as follows. A skip char trigram feature “wmn”, which represents words such as “woman”, “women”, and “women’s” and hastags such as “#women” and “#womenforwomen”. A skip char bigram feature “lf”, which represents words such as “life”, “prolife”, and “pro-life” and hastags such

as "#everylifematters" and "#ProLifeYouth". A char bigram "wo", which is common to some frequent relevant words and hastags such as "woman", "women", "women's", "#women", and "work", and also of non-relevant frequent words such as "would". The only word unigram, which is among the top ranked features is "men", a group of people which also has what to tweet about abortion.

The main conclusion from these results is that most of the top features are character ngrams and skip character ngrams. These features serve as generalized features that include within them semantically close words and hastags, and their declensions. These features allow to overcome problems such as noise and sparse data and enable successful classification.

**Comparison to the Contest Results.** In the contest, organized by the SemEval 2016 Task 6-A [14] for all the test datasets, the organizers used the macro-average measure as the evaluation metric for the task. The macro-average (also called *Favg*) is defined as:

$$Favg = (Ffavor + Fagainst) / 2 \tag{1}$$

where *Ffavor* and *Fagainst* are defined as follows:

$$Ffavor = 2PfavorRfavor / (Pfavor + Rfavor) \tag{2}$$

$$Fagainst = 2PagainstRagainst / (Pagainst + Ragainst) \tag{3}$$

Our results were: *Fagainst* = 90.7, *Ffavor* = 73.8, and *Favg* = 82.25. The score of 82.25 is significantly higher than the *Favg* results of all the 19 competitors, including the best *Favg* result (66.42) obtained by the baseline SVM-ngrams team using all the possible word ngrams (this team was not a part of the official competition) and the best *Favg* result (63.32) achieved by the DeepStance team (a part of the official competition) using ngrams, word embedding vectors, sentiment analysis features such as those drawn from sentiment lexicons [32], and stance bearing hashtags.

In contrast to the *Favg* scores of many of the competitors of the SemEval 2016 Task 6-A, that were obtained using thousands of ngrams and hundreds of word embedding features, our *Favg* score is significantly better mainly probably due to the use of the CFS feature selection method and the use of only 100 derived features where 93 of them are character-based features and 77 of them are skip char ngram features.

### 4.2 Summary of the Results for All Five Datasets

Table 3 presents a summary of the results of our algorithm for all the five test datasets and Table 4 presents a comparison of the *Favg* values and an analysis of our features.

General findings that can be derived from Table 3 are: (1) The best ML methods are the two SVM's versions and Naïve Bayes; (2) The best filtering method is CFS; (3) The number of the filtered features is relatively very small (between 53 to 111); and (4) The values of all measures are relatively high (around 85% and up) for all test datasets.

**Table 3.** Summary of the results of our algorithm for all the five test datasets.

| Data Set | Best ML method | Best filtering method | # of filtered features | Acc | Prc | Rec | F-M | ROC area | PRC area |
|---|---|---|---|---|---|---|---|---|---|
| Abortion | LibSVM | CFS | 100 | 86.43 | 86.2 | 86.4 | 86.3 | 0.93 | 0.91 |
| Climate | SMO norm. pol-kernel | CFS | 53 | 86.39 | 85.1 | 86.4 | 85.75 | 0.82 | 0.79 |
| Feminist | SMO default pol-kernel | CFS | 102 | 83.51 | 83.9 | 83.5 | 83.7 | 0.82 | 0.75 |
| Clinton | NB | CFS | 111 | 85.42 | 86.5 | 85.4 | 85.95 | 0.93 | 0.88 |
| Atheism | NB | CFS | 74 | 79.55 | 85.6 | 79.5 | 82.44 | 0.93 | 0.91 |

**Table 4.** Comparison of the *Favg* values and an analysis of our features.

| Data Set | % of skip char ngrams | % of char ngrams | Best team in Task 6-A | | *Favg* of our system |
|---|---|---|---|---|---|
| | | | Team | *Favg* | |
| Abortion | 77.0% | 93.0% | SVM-ngrams | 66.42 | 82.25 |
| Climate | 77.4% | 94.3% | IDI@NTNU | 54.86 | 65.1 |
| Feminist | 75.5% | 94.1% | MITRE | 62.09 | 79.45 |
| Clinton | 82.9% | 96.4% | TakeLab | 67.12 | 77.8 |
| Atheism | 78.4% | 93.2% | TakeLab | 67.25 | 80.95 |
| Average | 78.24% | 94.2% | – | 63.55 | 77.11 |

General findings that can be drawn from Table 4 are: (1) The average rate of the skip char ngram features is around 78%; (2) The average rate of all the character-based features is around 94%; and (3) The average value of our *Favg* (77.11) is significantly higher than the average value of *Favg* of the best teams in the five experiments (63.55).

On the one hand, it is not surprising that the best classification results are successful with char ngrams features (around 94% of the features) because tweets are much more characterized by characters than by words, tweets are known as relatively short (up to 140 characters), and they contain also various hashtags, typos, shortcuts, slang words, onomatopoeia, and emojis.

On the other hand, it is relatively surprising that the skip character ngrams (around 78% of the features) contribute the most to the success of the classification tasks. The skip character ngrams that can be regarded as a type of generalized ngrams (because they enable gaps that are skipped over) have been discovered as "anti-noise" features that perform very well in a noisy environment such as twitter corpora.

As mentioned before by Guthrie et al. [13], skip-grams enable to overcome the data sparsity problem (i.e., the text corpus is composed of rare text units) for machine translation tasks even for an extremely large corpus. Based on our experiments, skip character ngrams do not only enable to overcome the data sparsity problem (which characterizes

short text corpora) but also help to overcome noisy problems (e.g., misspellings, onomatopoeia, replicated characters, and slang words), which also characterize short text corpora.

## 5 Summary, Conclusions and Future Work

In this study, we present an implementation of stance classification tasks based mainly on a limited number of features, which contain mainly char ngrams features in general and char skip ngrams in particular. To the best of our knowledge, we are the first to perform successful stance classification using mainly skip character ngrams.

The macro-average results obtained by our algorithm are significantly higher than the state-of-art results reported by the best macro-average results achieved in the SemEval 2016 Task 6-A [14] for all the five released datasets of tweets in the framework of task-A (the supervised framework).

In contrast to the competitors of the SemEval 2016 Task 6-A, that did not use any char skip ngrams but rather used thousands of ngrams and hundreds of word embedding features, our algorithm uses a limited number of features (53–111) derived by the CFS selection method, mainly character-based features where most of them are skip char ngram features.

Our experiments show that two feature sets are very helpful for stance classification of tweets: (1) char ngrams features in general probably because tweets are much more characterized by characters than by words, tweets are relatively short (up to 140 characters), and contain also various typos, shortcuts, hashtags, slang words, onomatopoeia, and emojis and (2) skip character ngrams in particular probably because they serve as generalized ngrams that allow to overcome problems such as noise and sparse data.

In order to examine the usefulness of character ngrams in general and skip character ngrams in particular we suggest the following future research proposals: conducting additional experiments for larger social corpora of various types of short text files written in various languages based on more feature sets and applying additional supervised ML methods such as deep learning methods.

**Acknowledgments.** The authors thank three anonymous reviewers for their help and fruitful comments.

## References

1. Liu, B., Zhang, L.: A survey of opinion mining and sentiment analysis. In: Aggarwal, C., Zhai, C. (eds.) Mining Text Data, pp. 415–463. Springer, Boston (2012). https://doi.org/10.1007/978-1-4614-3223-4_13
2. Mohammad, S.M., Zhu, X., Kiritchenko, S., Martin, J.: Sentiment, emotion, purpose, and style in electoral tweets. Inf. Process. Manage. **51**(4), 480–499 (2015)
3. Basave, C., He, A.E., He, Y., Liu, K., Zhao, J.: A weakly supervised bayesian model for violence detection in social media (2013)

4. Somasundaran, S., Wiebe, J.: Recognizing stances in ideological on-line debates. In: Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, pp. 116–124. Association for Computational Linguistics (2010)
5. Murakami, A., Raymond, R.: Support or oppose? Classifying positions in online debates from reply activities and opinion expressions. In: Proceedings of the 23rd International Conference on Computational Linguistics: Posters, pp. 869–875. Association for Computational Linguistics (2010)
6. Anand, P., Walker, M., Abbott, R., Tree, J.E.F., Bowmani, R., Minor, M.: Cats rule and dogs drool!: classifying stance in online debate. In Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis, pp. 1–9. Association for Computational Linguistics (2011)
7. Sridhar, D., Foulds, J., Huang, B., Getoor, L., Walker, M.: Joint models of disagreement and stance in online debate. In: Annual Meeting of the Association for Computational Linguistics (2015)
8. Thomas, M., Pang, B., Lee, L.: Get out the vote: determining support or opposition from Congressional floor-debate transcripts. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, pp. 327–335. Association for Computational Linguistics (2006)
9. Yessenalina, A., Yue, Y., Cardie, C.: Multi-level structured models for document-level sentiment classification. In: Proceedings of EMNLP, pp. 1046–1056 (2010)
10. Burfoot, C., Bird, S., Baldwin, T.: Collective classification of congressional floor-debate transcripts. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, pp. 1506–1515. Association for Computational Linguistics (2011)
11. Agrawal, R., Rajagopalan, S., Srikant, R., Xu, Y.: Mining newsgroups using networks arising from social behavior. In: Proceedings of WWW, pp. 529–535 (2003)
12. Rajendran, P., Bollegala, D., Parsons, S.: Contextual stance classification of opinions: a step towards enthymeme reconstruction in online reviews. In: Proceedings of the 3rd Workshop on Argument Mining, pp. 31–39. Association for Computational Linguistics, Berlin (2016)
13. Guthrie, D., Allison, B., Liu, W., Guthrie, L., Wilks, Y.: A closer look at skip-gram modelling. In: Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC-2006), pp. 1222–1225 (2006)
14. Mohammad, S.M., Kiritchenko, S., Sobhani, P., Zhu, X., Cherry, C.: SemEval-2016 task 6: detecting stance in tweets. In: Proceedings of SemEval, pp. 31–41 (2016)
15. Xu, R., Zhou, Yu., Wu, D., Gui, L., Du, J., Xue, Y.: Overview of NLPCC Shared Task 4: stance detection in Chinese microblogs. In: Lin, C.-Y., Xue, N., Zhao, D., Huang, X., Feng, Y. (eds.) ICCPOL/NLPCC -2016. LNCS (LNAI), vol. 10102, pp. 907–916. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-50496-4_85
16. Jans, B., Bethard, S., Vulić, I., Moens, M.F.: Skip n-grams and ranking functions for predicting script events. In: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pp. 336–344. Association for Computational Linguistics (2012)
17. Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., Chanona-Hernández, L.: Syntactic n-grams as machine learning features for natural language processing. Expert Syst. Appl. **41**(3), 853–860 (2014)
18. Fernández, J., Gutiérrez, Y., Gómez, J.M., Martınez-Barco, P.: GPLSI: supervised sentiment analysis in twitter using skipgrams. In: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), number SemEval, pp. 294–299 (2014)

19. Dhondt, E., Verberne, S., Weber, N., Koster, C., Boves, L.: Using skipgrams and pos-based feature selection for patent classification. Comput. Linguist. Neth. J. **2**, 52–70 (2012)
20. HaCohen-Kerner, Y., Beck, H., Yehudai, E., Rosenstein, M., Mughaz, D.: Cuisine: classification using stylistic feature sets and/or name-based feature sets. J. Am. Soc. Inform. Sci. Technol. **61**(8), 1644–1657 (2010)
21. HaCohen-Kerner, Y., Beck, H., Yehudai, E., Mughaz, D.: Stylistic feature sets as classifiers of documents according to their historical period and ethnic origin. Appl. Artif. Intell. **24**(9), 847–862 (2010)
22. Mohammad, S.M., Kiritchenko, S., Zhu, X.: National Research Council Canada (NRC) Hashtag unigram Lexicon (2013). http://saifmohammad.com/WebPages/SCL.html. Accessed 18 Apr 2017
23. https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html. Accessed 18 Apr 2017
24. http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/. Accessed 18 Apr 2017
25. https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html. Accessed 18 Apr 2017
26. http://nlp.stanford.edu/software/tagger.shtml. Accessed 18 Apr 2017
27. Klein, D., Manning, C.D.: Accurate unlexicalized parsing. In: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics, vol. 1, pp. 423–430. Association for Computational Linguistics (2003)
28. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, San Mateo (2005)
29. Hall, M.: Correlation-based feature selection for machine learning. Doctoral dissertation, The University of Waikato (1999)
30. Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: Icml, pp. 412–420 (1997)
31. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software. ACM SIGKDD Explor. Newsl. **11**(1), 10 (2009)
32. Kiritchenko, S., Zhu, X., Mohammad, S.M.: Sentiment analysis of short informal texts. J. Artif. Intell. Res. **50**, 723–762 (2014)