FACTA UNIVERSITATIS (NIŠ)
Ser. Math. Inform. Vol. 33, No 5 (2018), 683–696
https://doi.org/10.22190/FUMI1805683L

COMPARISON OF THE INFLUENCE OF DIFFERENT NORMALIZATION METHODS ON TWEET SENTIMENT ANALYSIS IN THE SERBIAN LANGUAGE

Adela Ljajić, Ulfeta Marovac and Milena Stanković

Abstract. Given the growing need to quickly process texts and extract information from the data for various purposes, correct normalization that will contribute to better and faster processing is of great importance. The paper presents the comparison of different methods of short text (tweet) normalization. The comparison is illustrated by the example of text sentiment analysis. The results of an application of different normalizations are presented, taking into account time complexity and sentiment algorithm classification accuracy. It has been shown that using cutting to n-gram normalization, better or similar results are obtained compared to language-dependent normalizations. Including the time complexity, it is concluded that the application of this language-independent normalization gives optimal results in the classification of short informal texts.

1. Introduction

Normalization is an important step in text preparation for any type of machine processing. Normalization can be language-independent and language-dependent. Language-dependent normalization better preserves text properties and reduces the word to morphologically correct form. The problems of language-dependent normalization are unavailability and robustness of lexical resources and complexity of the normalization algorithms. Language-independent normalization reduces words to forms that do not necessarily have to be morphologically correct. On the other hand, specific lexical resources are not required for the use of language-independent normalizations. Cutting the word to the character n-grams of a certain length, as a way of language independent normalization, can have its advantages in particular text processing. This normalization is much faster than linguistic normalization and is preferred if it achieves satisfactory processing precision. This paper presents the effect of text normalization on the classification of short texts (tweets) based

Received October 10, 2018; accepted November 27, 2018
2010 *Mathematics Subject Classification.* Primary xxxxx; Secondary xxxxx, xxxxx

on the sentiment. Three types of normalization have been processed, two linguistic (stemming and lemmatization) and one language-independent (cutting words to character n-grams). The rest of the paper is organized as follows. The paper begins with a description of related work on different normalization methods for sentiment analysis in Section 2. Section 3. contains information about the dataset. Section 4. describes different normalization methods. Section 5. shows the results of the sentiment lexicon normalization. The experimental results and time complexity are included in Section 6. Finally, we conclude and present suggestions for future work in Section 7.

2. Related work

The sentiment analysis has been an ongoing topic of research recently. It seeks to determine the attitude expressed in the text. The sentiment can be analyzed at the level of the whole text, sentences or one aspect of the text [16]. The sentiment can be expressed discreetly (positive, negative and neutral) or on a scale from positive to negative. There are corpus-based and lexicon-based approaches to determine the polarity of the corresponding text [19]. The corpus-based approach (supervised approach) uses the methods of machine learning over a marked set of data. The lexicon-based approach (unsupervised approach) determines the polarity based on the sentiment lexicon. The sentiment lexicon contains words that can have discrete values (-1, 0, 1 or positive, neutral, negative) or values on a scale (eg. from -10 to +10). Sentiment lexicons with discrete values are Bing Liu's Opinion Lexicon [4] and MPQA Subjectivity Lexicon [21]. Sentiment lexicon containing the value of the polarity that has a specific scale is SentiWordNet [18]. When it comes to methods that determine the sentiment, most researchers use supervised learning methods [3], although a considerable number of approaches provide analysis by methods of unsupervised lexicon-based [19] and [15] or combined semi-supervised learning [2]. Although a machine-based approach gives better classification, a lexicon-based approach takes precedence in situations where a set of marked data is not available, and when a classifier training time is crucial. In the sentiment analysis, there are challenges such as the treatment of phenomena of negation, sarcasm, irony, and others. The sentiment analysis is closely related to the language. A sentiment analysis in the Serbian language was made for a set of newspaper articles [14], film reviews [1] and a set of tweets [8]. Normalization of text is a part of every kind of text processing and sentiment analysis. The effect of normalization on various problems of text processing is different. The normalization results vary depending on the type of text to which they are applied and the language in which the text is written. There are a small number of papers concerning the normalization of short texts in the Serbian language and its related languages (Bosnian, Croatian and Montenegrin). Linguistic normalization of the texts in the Serbian language was performed by D. Vitas et al. [20], who described tools and resources for the processing of texts in the Serbian language. In addition to linguistic normalization, normalization by stemming can be done using stemmers. The authors of the paper [1] dealt with the impact of morphological, stemming and word embedding normalization on the sentiment classification of text in the Serbian language. They used a movie review corpus and found that stemmer gives better results compared to lemmatizer and that adding bag-of-words attributes increases the accuracy of classification methods. The application of the word embedding method (which requires lemmatization) and the string kernel method (which does not require any normalization) in the sentiment analysis of informal short texts in the Croatian language is shown by L. Rotim et al. [17]. Their results show that word embedding outperforms string kernels, which in turn outperform word and n-gram bag-of-words baselines. Alternative methods of normalization are known, such as cutting off the same length and n-gram analysis [11].

3. Dataset

The increasing use of social networks and the text availability make them popular for research. In this paper, the experimental dataset consists of tweets in the Serbian language. Tweets are short, informal texts that contain a lot of incorrectly written words, use of slang, irony, and sarcasm. If we take all these into consideration, we can conclude that normalization and classification of such informally written texts is a very demanding task. Liu et al., [7] created special systems for normalizing such texts. Tweets were collected using the Twitter Streaming API in the period from 30 November 2016 to 30 June 2017. The dataset was manually labeled by three people, two men, and one woman. Background of annotators are the following: a doctor of medicine, an electrical engineer and a student of the Serbian language and literature. In case of disagreement on tweet marking between any two or all three annotators, the tweet is thrown out of the set. The final dataset consists of 7663 tweets, 4193 of which are marked negative, 2625 neutral and 818 positives.

4. Normalization

Normalization is considered the process consisting of two phases. In the first phase, a tokenization that is linguistically independent and specific to the type of data is performed. The tokenizer deals with words that appear in tweets but do not carry the meaning (such as retweet, via, etc.), spaces in the text, numbers, dates, and punctuation characters in such a way that output tokens are only those that affect the meaning of the text. The second phase of text normalization is partially or completely linguistically dependent. In this paper, it involves removing the stop words specific for the Serbian language and reducing different forms of the words to their base. Reduction of the number of different types of words appearing in the dataset is done in three ways: stemming (ST), normalization by using morphological lexicon (MN) and cutting to the character of 4-grams, 5-grams, 6-grams and 7-grams (4G, 5G, 6G, 7G).dgsd

4.1. Stemming

Stemming belongs partly to linguistic, and partly to the heuristic approach of reducing different word forms to their root. Stemming removes the word suffixes by cutting them to the root of the word. There are publicly available stemming algorithms in the Serbian language. The paper used a stemming algorithm for the Serbian language described by N. Miloević [13]. This stemmer consists of a list of irregular verbs: moći/can, hteti/want, jesam/I am, and biti/to be. For each form of these verbs (a total of 68 infections), the word is reduced to its morphological root. Stemmer also contains a list of 289 suffixes and substitutions that are added if the suffix is taken from the word. Stemmer modifies the first words containing letters with diacritic characters, so the letters with a diacritic sign are replaced with two letters: "š" is modified to "sx", "č" to "cx", "ć" to "cy", "đ" to "dx" and "ž" to "zx". Further, if the word belongs to the list of one of the 4 irregular verbs, then it is shifted to the root; otherwise, the longest suffix contained in the word is found, and the word is modified according to the rule for that suffix. If the word does not contain any of the suffixes, then the stemmer returns the original word. This stemmer is set to stem words longer than 4 characters and those with more than 3 characters after the suffix is taken away. Stemmers created by Kešelj and Šipka [5] are also based on rules with suffixes. The algorithm for stemming of Ljubešić [10] is rule-based and achieves F1 97.64

4.2. Normalization with the morphological lexicon

Lemmatization is the process of reducing different forms of a single word to its linguistic root - lemma. Different forms of one and the same word occur when this word appears in different grammatical cases, grammatical gender, grammatical number, and grammatical tense or grammatical person. Lemmatization is used in the morphological analysis of the text; the morphological lexicon is used for the process of word reduction. The morphological lexicon contains all the forms of the word and word lemma. The lemma is derived from the form of words and other labels needed to uniquely map to its corresponding lemma. The additional tags of the word include the aforementioned information about the grammatical case, grammatical gender, grammatical number, grammatical tense or grammatical person and other characteristics - depending on the type of the word that is reduced to the lemma. For the application of lemmatization, a corpus with labeled word POS tag is required. Due to the absence of such corpus, normalization is applied in this paper by using the morphological lexicon that takes the first lemma for the corresponding word form, without taking into account the characteristics of the word. This normalization has defects in relation to real lemmatization. Since lemmatization is performed by using a large number of rules (each word is considered separately), it is accordingly more complex and time-consuming in comparison to stemming. The morphological lexicon of Krstev et al. is used for lemmatization [6]. It consists of 3,630,613 entries for 85,721 lemmas covering 11 PoS: 646

867 nouns, 2 315 640 adjectives, 654 159 verbs, 3233 adverbs, 4 794 numerals, 83 conjunctions, 218 interjections, 169 prepositions, 5 321 pronouns, 103 particles and 26 abbreviations.

4.3. Cutting to n-gram

An alternative way of normalization, which does not require any lexical resources is normalization by cutting the word into the first n characters. For n, the values of 4, 5, 6, and 7 are taken. This way of text normalization certainly results in information loss, and it may happen that different words with different sentiments are reduced to the same n-gram (ambicija/ambition, besplatno/free (positive words) and ambis/ambis, bespomoćno/helpless (negative words) are reduced to the same 4-grams - ambi, besp). However, the advantages of reduction of a large number of word forms with the same first n characters may bring greater benefits in comparison to losses (e.g.ljubav/love, ljubavni/love, ljubavisati/love, ljubavi/love are reduced to the same 4 grams ljuba/love). The gains and losses obtained by this normalization were experimentally shown (in Section 5.). In the example of the Twitter sentiment classification, this normalization was experimentally shown to positively affect the accuracy of classification.

5. Normalized sentiment lexicons

The sentiment lexicons contain words that are marked positive or negative. They serve in the sentiment analysis in the lexicon-based method. The sentiment lexicon used in this paper as the starting lexicon consisted of 5632 words (reduced to the morphological root), 4058 of which were negative and 1574 positive words [12]. The three described normalizations were used, and three resulting lexicons were obtained and used in a dataset, normalized by one of the three normalizations (stemming, lemmatization or cutting to n-grams).

5.1. Normalization of sentiment lexicon

The application of normalization to sentiment lexicons affects the total number of words in the lexicon as well as their quality. By using linguistic dependent normalizers (stemmer and lemmatizer), the words with the same or similar meaning are reduced to their common root. Using n-gram analysis, words from a lexicon are cut into n-grams, without taking into account the meaning of the word. Due to the characteristics of the corresponding normalizers, the resulting lexicons have a significantly different number of words [11]. The small numbers of words with different sentiments are transformed to the same root, due to which they become contradictory. Normalizations based on language rules produce a lower number of such words. Contradictory data will be excluded from the sentiment lexicons. Table 5.1 shows the results after normalization of words in lexicons and removal of contradictory words. The number of words in the lexicons is displayed as well as the total number of different roots obtained after normalization. For better comparison, the results are presented in cases when no normalization is applied (NN) and the results of the application of different normalizations: stemming (ST), morphological

Table 5.1: Number of words in normalized lexicons after the removal of contradictory words and number of different roots to which they are reduced by normalizations

Type of normalization	**NN**	**ST**	**NM**	**4G**	**5G**	**6G**	**7G**
Number of different words	5632	5596	5632	4139	5116	5481	5576
Number of different bases	5632	5218	5632	2271	3506	4283	4803

vocabulary normalization (NM), 4-grams (4G), 5-grams (5G), 6-grams (6G) and 7-grams (7G).

The number of different words in nominalized lexicons decreases due to the exclusion of contradictory words, which is best expressed in normalization by cutting to 4-grams. Stemming is also part of the word ejected. The number of different normalized roots to which the words from the basic lexicon are reduced is the smallest in cutting to 4 grams, which increases with the length n. The number of occurring sentiment words in the data set is calculated for lexicons. The total number of sentiment words is: for stems 10777; for lemmas 12920; for 4-grams 22466; for 5-grams 15284; for 6-grams 10697; for 7-grams 7874. The distribution of the number of occurrences of the sentiment in the corpus of polarity is shown in Figure 5.1. The number of occurrences of terms from normalized lexicons in tweets from the corpus is the largest for 4-grams and 5-grams and for the lemmas. This distribution indicates that words cut to 4-grams are best mapped in tweets, which is expected due to the number of different word forms, beginning with the same 4-gram. What is visible from the chart and Table 5.1 is that the effect of normalization is reduced with the increase of n length, so that by cutting to 7-grams, we obtain those that are inclined towards results without normalization.

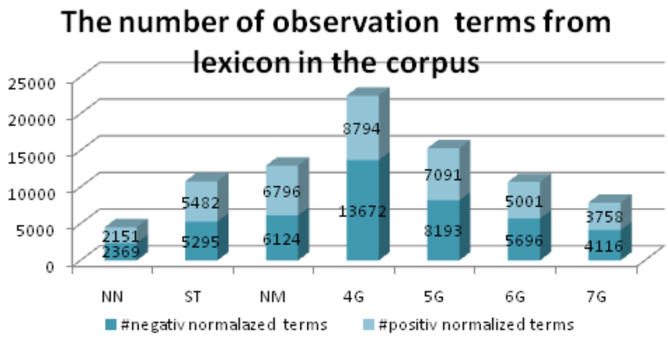

FIG. 5.1: The number of observation terms from lexicon in the corpus

In the next chapter, the quality of collected lexicon is verified by examining whether these words from normalized lexicon appear in tweets with the corresponding polarity, and how such normalized lexicon affects the classification by sentiment.

6. Results and discussion

The results of normalization are presented in two directions. In the first direction, the analysis of sentiment lexicons is done in the way that provides validation over the marked set of tweets, so the appearance of specific words in tweets by class is calculated. The second direction of results is based on the sentiment analysis of a normalized dataset using normalized sentiment lexicons.

6.1. Validation of the sentiment lexicon over a tweet dataset

The validity of lexicon is based on a set of tweets that are normalized by corresponding normalization. For each normalized root of sentiment word (stem, lemma or n-gram), the calculation (the number of occurrences of that word in tweets with the positive and negative sentiment) is done. If the sentiment word occurs within the negation scope, its appearance is counted as having appeared in a tweet with the opposite polarity. Normalization of the score is done by dividing the number of occurrences with the number of tweets from that class (the data set is unbalanced). The score is calculated by the formula (6.1) with n as a number of affirmative occurrences in positive tweets, and nn number of occurrences with negation in positive tweets; m is a number of affirmative occurrences in negative tweets, and nm is the number of occurrences with negation in negative tweets.

$$score = (n - nn)/num\_positive\_tw - (m - nm)/num\_negativ\_tw. \tag{6.1}$$

By classifying the sentiment words based on whether they appear more in positive or in negative tweets, the effect of normalization on sentiment analysis is tested. The sentiment is assigned to the word in the following way:
positive- if they appear more in positive than in negative tweets, (score>0).
negative- if they appear more in negative than in positive tweets, (score<0).
Sentiment word need not be classified (when score=0) for two reasons. The first reason is that sentiment word does not appear in the corpus, and second is that sentiment word appears equally in positive and negative tweets; this latter case is rare. Table 6.1 shows the classification results of positive sentiment words, negative sentiment words and all sentiment words from lexicons when different types of normalization are applied. Formula (6.2) presents precision (Pre), as a the number of correctly classified sentiment words($num\_corectly\_classified$) divided by the total number of sentiment words classified as belonging to the corresponding sentiment class ($total\_num\_classified$). Formula (6.3) presents recall (Rcall) as the number of correctly classified sentiment word($num\_corectly\_classified$) divided by the total number of sentiment words that actually belong to the corresponding sentiment class($total\_num$). Measure F1 uses a combination of Precision and Recall presented in formula (6.4), giving more relevant results with an unbalanced dataset.

$$Pre = num\_corectly\_classified/total\_num\_classified \tag{6.2}$$

Table 6.1: The obtained precision and recall for different types of used normalizations. The corresponding maximums are labeled red.

		NN	ST	NM	4G	5G	6G	7G
ACC negative words	Pre	82%	81%	80%	84%	82%	81%	80%
	Rcall	16%	27%	25%	60%	45%	35%	28%
	F1	26%	41%	38%	70%	58%	49%	41%
ACC positive words	Pre	79%	67%	66%	52%	60%	64%	69%
	Rcall	13%	24%	19%	47%	37%	28%	22%
	F1	22%	35%	30%	49%	46%	39%	34%
ACC all words	Pre	81%	77%	77%	75%	76%	76%	77%
	Rcall	15%	26%	23%	57%	43%	33%	26%
	F1	25%	39%	36%	65%	55%	46%	39%

$$Rcall = num\_corectly\_classified/total\_num \tag{6.3}$$

$$F1 = 2 * Pre * Rcall/(Pre + Rcall) \tag{6.4}$$

From the obtained results, it can be concluded that n-grams (n < 7) are well classified by sentiment (the classification has the best F1 score). The reason is that a larger number of n-grams were found in the set of tweets compared to stemms and lemmas. Being informal texts, tweets often contain misspelled words that are rarer at length up to 6 letters. On the other hand, the Serbian language, being morphologically rich is difficult to process, and a large number of words are found in forms that are not adequately processed by stemmer and lemmatizer, hence such sentimental words cannot be found in the sentiment lexicon. Testing the improvement of classification of sentiment words by cutting them to n-grams versus stemmer and lemmatizer is done using Mc Nemar's test. We made a correlation matrix for classification by using n-gram analysis and lemmatization and 4-gram analysis and stemming. In both cases, the value of p <0.0001 was found, i.e. cutting on n-grams had statistically significant influence on the improvement of sentiment word classification.

6.2. Application of normalized lexicon to tweet sentiment analysis

The influence of the three normalization methods was tested on Twitter sentiment analysis. Normalized lexicons and normalized data set were used to determine the sentiment in two experiments. The first experiment classified tweets based only on the words found in the sentiment lexicon. In another experiment, the methodology for learning Multinomial Logistic Regression (MLR) was used for classification by sentiment. Figure 6.1 shows the system architecture from collecting data, through normalization to sentiment analysis.

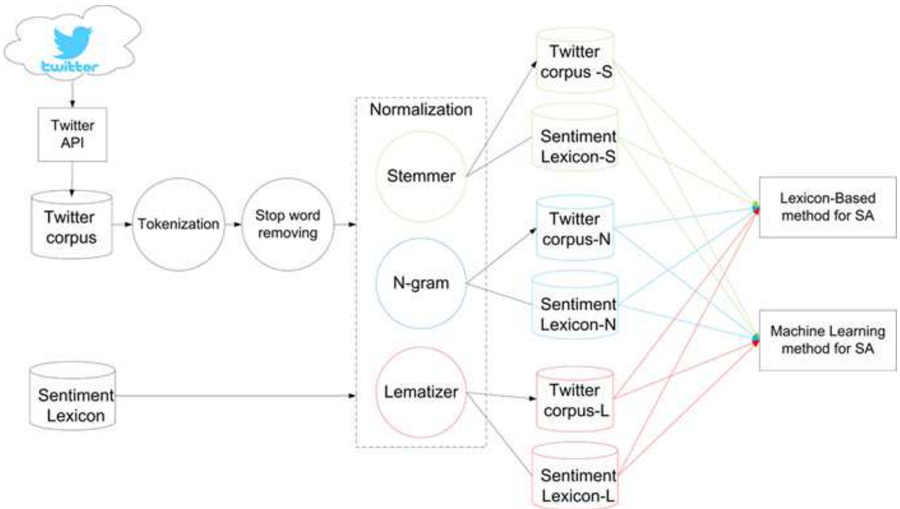

FIG. 6.1: System architecture from collecting tweets, through the normalization process to the sentiment analysis

In the first experiment, the quality of the prediction for three normalization methods was performed by a lexicon-based method. The advantage of this classification method is that it does not require training and is independent of the dataset. Although the results are worse than in the machine learning approach, this algorithm gives us a better insight into the impact of different normalizations of the sentiment analysis. The sentiment is calculated according to formula 6.5. The sum of numbers of positive sentiment terms and negative sentiment terms within the negation scope that appear in tweets is given in the sumPos attribute. The sum of numbers of negative sentiment terms and positive sentiment terms in the negation scope from that appear in tweets is given in the sumNeg attribute [9].

$$D_{it} = \begin{cases} positive & \text{if sumPos>sumNeg} \\ neutral & \text{if sumPos=sumNeg} \\ negative & \text{if sumPos<sumNeg} \end{cases} \tag{6.5}$$

The results obtained by this method are given in Table 6.2. The table contains results when no normalization, stemming, morphologic dictionary or cutting to n-grams (n = 4,5,6 and 7) are applied. The results of the 3-class (3K) classification of

Table 6.2: Correctly Classified tweets using Lexicon-Based method depending on normalization

Normalization	NN	ST	NM	4G	5G	6G	7G
Lexicon-Based 3K	47.09%	50.51%	48.98%	49.25%	49.10%	50.13%	49.15%
Lexicon-Based 2K	33.45%	52.17%	50.73%	59.69%	53.54%	49.51%	43.76%

Table 6.3: Correctly Classified tweets using machine lerning method depending on normalization

Normalization	NN	ST	NM	4G	5G	6G	7G
MLR-3K	59.86%	64.17%	62.45%	59.23%	60.61%	62.49%	61.80%
MLR-2K	84.71%	85.27%	84.25%	83.96%	84.45%	84.47%	84.71%

tweets in positive, neutral and negative are presented, as well as the classification for 2-class (2K) positive and negative. If we only look at the classification of positive and negative tweets, we get 4-gram normalization giving the best accuracy. However, neutral tweets distort the classification quite a lot, so when classifying a group of tweets with three classes, normalization using stemmer gives the best results. In this case, cutting to n-gram finds a large number of n-grams with the sentiment in tweets, even in neutral, which classifies them into positive or negative. Neutral tweets often carry a part of the sentiment that is not clearly defined, and this can be solved only by introducing the classification of tweets into several sentiment groups. On the other hand, by cutting the word on n-grams, a set of sentiment words are lost if, after normalization, they are infiltrated into a group of contradictory ones. The omission of these sentiment words distorts the sentiment analysis classification. The classification quality is shown using a percentage of accurately classified tweets using formula (6.6).

$$(6.6) \qquad Correct\_classif = num\_corect\_classif\_tweets/total\_num\_of\_tweets$$

In the second experiment, supervised machine learning was performed using MLR in 10-fold cross-validation (Table 6.3). The attributes used for this method are the following: sumNeg, sumPoz, the number of words in negation scope, the number of words in the tweet. Here we see a significant increase in the results obtained for all three normalization methods, where stemming achieved the best result. The normalization by cutting to the n-grams, in this case, is the best for 6-grams and is more accurate than the normalization using the morphological lexicon.

6.3. Complexity of algorithms

Large amounts of data available for processing require techniques to quickly achieve results. In order to measure the complexity of sentiment analysis algorithms, the time complexity of determining the sentiment for one tweet has been shown, by using all three types of normalization. Differences in complexity of the sentiment algorithm in different modes of normalization are reflected in the size of lexical resources or the number of rules, used to normalize tweet and determine sentiment by formula (6.7). The first part of normalization does not depend on the way words are reduced and it will not be considered. Only the part that is specific for each normalization is considered. Sentiment analysis algorithm does not directly depend on normalization, but indirectly through the size of sentiment lexicon which is obtained by normalization. How the complexity of sentiment analysis algorithm depends on normalization is presented through the complexity of sentiment lexicon used (6.8).

$$iT = tweet\_normalization + determ\_senti \tag{6.7}$$

$$determ\_senti = num\_of\_words\_in\_tweet * num\_of\_words\_in\_senti\_lex \tag{6.8}$$

1. The normalization with a stemmer, as a linguistic dependent normalization, depends on the number of rules used in stemming and the size of the stemmed sentiment lexicon. Based on previously presented values, it is obtained as described in detail in Subsection 5.1. and Subsection 4.1. that:
tweet_normalization = number_of_words_in_tweet * number_od_rules_in_stemmer= m * (68 + 289)
determining_sentiment = m * 5218
iT = m * 5575

2. The use of morphological lexicon is the most expensive process due to robust lexical resources it uses. The size of morphological lexicon determines the complexity of normalization. For the normalization of morphological lexicon based on formulas (6.7) and (6.8), the following complexity is obtained:
tweet_normalization = number_of_words_in_tweet *number_od_rules_in_lemmatizer= m * 3,630,613
determining_sentiment = m * 5632
iT = m * 3636245

3. Cutting to n-grams requires the fewest resources, as shown in Table 6.4. The complexity of normalization is reduced to the number of words in the tweet. The sentiment lexicon normalized by cutting into n-grams is also smaller than the stemmed and lemmatized lexicon. Depending on length n, the complexity by cutting to n-grams is:
tweet_normalization = number_of_words_in_tweet * number_of_ngram_rules= m * 1

The obtained results indicate that normalization by cutting to n-grams is the least required and the fastest normalization algorithm and gives better results than lemmatization. If we compare it with stemming, the results are also satisfactory,

Table 6.4: Complexity of sentiment analyses in case of cutting to n-gram normalization

n	determining_sentiment	iT
4	m*2271	m*2272
5	m*3506	m*3507
6	m*4283	m*4284
7	m*4803	m*4804

since the classification into two classes of sentiments is always satisfactory. The problem occurs more due to the nature of sentiment analysis, i.e. unclearly defined type of neutral tweets that mainly carry both sentiments words by nature.

7. Conclusion

Cutting to n-grams maps a great number of words in tweets, so the number of accurately classified tweets is large. The problem arises with words that are thrown out of the sentiment lexicon because they are reduced to words with the opposite sentiment, therefore they do not participate in the sentiment analysis. Another problem is that neutral tweets contain the sentiment word that makes them difficult for classification. The results show that cutting off sentiment words into n-grams gives good results in classifying sentiment words in tweets, especially due to the informal form of tweet writing. Taking into account the accuracy of classification, the minimum of lexical resources, and the simplicity of application, cutting to n-grams is a method that has the advantage over linguistic dependent normalization in the Twitter sentiment analysis. In linguistic dependent normalizers, the use of stemmers takes precedence over the normalization with the morphological lexicon, both due to low complexity of the algorithm and the best result in the tweet sentiment classification in the 3-class dataset. In order to improve results, the sentiment analysis algorithm itself should be improved. Improving the result is possible using domain sentiment lexicon with sentiments that are also validated on the appropriate corpus. The introduction of several degrees in the sentiment analysis would significantly solve the problem of neutral tweet classification by sentiment.

REFERENCES

1. V. Batanović and B. Nikolić: *Sentiment classification of documents in Serbian: The effects of morphological normalization.* In:Proceedings of the 24th Telecommunications Forum (TELFOR), Belgrade, Serbia, 2016, pp. 1–4.
2. N. F.F. da Silva, L.F.S. Coletta and E.R. Hruschka: *A Survey and Comparative Study of Tweet Sentiment Analysis via Semi-Supervised Learning.* ACM Computing Surveys, v.49, n.1, 2016, pp. 15:1–15:26.

3. C.W. Hsu, C.C. Chang, C.J. Lin and M. Stede: *A practical guide to support vector classification* Technical Report, Department of Computer Science and Information Engineering, University of National Taiwan, Taipei, 2003, pp. 1–12.

4. M. Hu and B. Liu: *Mining and summarizing customer reviews*. In:Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Seattle, Washington, USA, 2004, pp. 168–177.

5. V. Kešelj and D. Šipka: *A Suffix Subsumption-Based Approach to Building Stemmers and Lemmatizers for Highly Inflectional Languages with Sparse Resources*. INFOtheca, v.9, n. 12, 2008, pp. 23a-33a.

6. C. Krstev, R. Stanković, I. Obradović, D. Vitas and M. Utvić : *Automatic Construction of a Morphological Dictionary of Multi-Word Units*. In: Loftsson H., Rognvaldsson E., Helgadottir S. (eds) Advances in Natural Language Processing. NLP 2010. Lecture Notes in Computer Science, v. 6233. Springer, Berlin, Heidelberg, 2010.

7. F. Liu, F. Weng and X. Jiang: *A broad-coverage normalization system for social media language*. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1 (ACL '12), Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 1035–1044.

8. A. Ljajić and U. Marovac: *Improving Sentiment Analysis for Twitter Data by Handling Negation Rules in the Serbian language*. Computer Science and Information Systems,2018.

9. A. Ljajić, M. Stanković and U. Marovac: *Detection of Negation in the Serbian Language*. In:Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics, Novi Sad, Serbia, 2018, pp. 39:1-39:6.

10. N. Ljubešić, D. Boras and O. Kubeka: *Retrieving Information in Croatian: Building a Simple and Efficient Rule-Based Stemmer*. In: INFuture2007: Digital Information and Heritage, Zagreb, Croatia: Department for Information Sciences, Faculty of Humanities and Social Sciences, 2007, pp. 313320.

11. U. Marovac, A. Ljajić, E. Kajan and A. Avdić: *Similarity Search in Text Data for the Serbian language*. In: Proceedings of ICEST (2013), Ohrid, Macedonia, 2013, pp. 607–610

12. U. Marovac, A. Ljajić, E. Kajan and A. Avdić: *Towards the Lexical Resources for Sentiment-Reach Informal Texts-Serbian language Case*. 5th International Conference CONTEMPORARY PROBLEMS OF MATHEMATICS, MECHANICS AND INFORMATICS (CPMMI 2018), State University of Novi Pazar, 2018.

13. N. Milošević: *Stemmer for the Serbian language*. arXiv 1209.4471, 2012.

14. M. Mladenović, J. Mitrović, C. Krstev and D. Vitas: *Hybrid Sentiment Analysis Framework for A Morphologically Rich Language*.Journal of Intelligent Information Systems, v. 46:3, 2016, pp. 599–620.

15. A. Moreo, M. Romero, J. Castrp and J.M. Zurita: *Lexicon-based comments-oriented news sentiment analyzer system*. Expert Systems with Applications, 2012, pp. 9166–9180.

16. B. Pang and L. Lee: *Opinion Mining and Sentiment Analysis* .Found. Trends Inf. Retr. 2008, 1–135.

17. L. Rotim and J. Šnajder: *Comparison of Short-Text Sentiment Analysis Methods for Croatian*. In:Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing (BSNLP 2017), Valencia, Spain, 2017, pp. 69–75.
18. B. Stefano, A. Esuli and F. Sebastiani: *SentiWordNet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining*. In: Proceedings of the Seventh Conference on International Language Resources and Evaluation, European Language Resources Association, 2010, pp. 168–177 2200-2204.
19. M. Taboada, J. Brooke, M. Voll and M. Stede: *Lexicon-based methods for sentiment analysis* Computational linguistics, 2011, pp. 267–307.
20. D. Vitas, C. Krstev, I. Obradović, Lj. Popović, and G. Pavlović-Lažetić: *An Overview of Resources and Basic Tools for Processing of Serbian Written Texts* . In: Proceedings of the Workshop on Balkan Language Resources, 1st Balkan Conference in Informatics, 2003.
21. J. Wiebe, T. Wilson and C. Claire: *Annotating Expressions of Opinions and Emotions in Language*. Language Resources and Evaluation, 2005, pp. 165–210.

Adela Ljajić
State University of Novi Pazar
Department of Technical Sciences
36300 Novi Pazar, Serbia
acrnisanin@np.ac.rs

Ulfeta Marovac
State University of Novi Pazar
Department of Technical Sciences
36300 Novi Pazar, Serbia
umarovac@np.ac.rs

Milena Stanković
Faculty of Electronic Engineering
Department of Computer Science
18000 Niš, Serbia
milena.stankovic@elfak.ni.ac.rs