Marcin PIETRAS[1]

West Pomeranian University of Technology (1)

doi:10.15199/48.2015.02.26

Sentence sentiment classification using fuzzy word matching combined with fuzzy sentiment classifier

Abstract. *This article focuses on semantic tagging of content in terms of sentimental meaning which may often lead to ambiguities between the primary sense of the word and its meaning in a particular expression. To address this issue, a specially modified Levenshtein distance algorithm for suffix-mitigation was used to measure similarity of words. Sentence sentiment classification was based on fuzzy logic approach and a fuzzy classifier. The presented method was experimentally tested with the sentimental analysis of selected sentences in the Polish language. Limitations of the presented method and possible improvements are discussed.*

Streszczenie. *Artykuł skupia się na semantycznym tagowaniu zawartości tekstu w kategoriach znaczenia sentymentalnego, które często może prowadzić do niejednoznaczności między pierwotnym wydzwiekiem słowa i jego znaczeniem w danej wypowiedzi. Aby zmierzyć się z tym zagadnieniem zastosowano specjalnie zmodyfikowany algorytm na odległość Levenshteina z łagodzeniem znaczenia końcówki fleksyjnej wyrazu do pomiaru podobieństwa słów. Sentymentalna klasyfikacja zdań została oparta na logice rozmytej i podejścia rozmytego klasyfikatora. Przedstawiona metoda została eksperymentalnie sprawdzona z sentymentalnej analizy wybranych zdań w języku polskim. Ograniczenia prezentowanej metody oraz możliwe ulepszenia są również omówiane.* ***(Klasyfikacja wydźwięku sentymentalnego zdań przy użyciu rozmytego dopasowania wyrazów w połączeniu z rozmytym klasyfikatorem znaczenia sentymentalnego).***

Keywords: Fuzzy logic; Words matching; Levenshtein distance; Sentimental estimation.
Słowa kluczowe: Logika rozmyta; Dopasowanie słów; Odległość Levenshteina; Estymacja wydźwięku sentymentalnego.

Introduction

Automated analysis of the sentimental meaning of the text provides a wealth of useful information not only about the content but also about the author. In recent years, the research on this subject has gained on importance [1]. Undoubtedly, it is related to the development of standards used to describe the content in the Internet that allow computers to semantically categorize the processed information. Due to its sentimental value, the content classification [2] is a difficult issue that requires machines to calculate what is anger or adoration.

These kind of abstract ideas are hard to clearly define. A recent research carried out in [3, 4] has allowed to estimate sentimentally Polish words thanks to the use of Corpus-Based Lexeme Sentiment Estimation and the semantic orientation–pointwise mutual information. This paper describes an attempt to classify entire sentences based on the numerical parameters defined by the sentimental meaning of words. For this purpose, fuzzy sentiment classifier for sentence sentimental meaning and fuzzy word matching algorithm to specify the degree of word similarity was developed.

Motivation

Automatic determination of content sentimental category expands the possibilities of human-machine interaction at the level of information adjustment and information presentation, including artificial empathy [5]. By the recognition of the sentimental content presented to the user, the program could be able to make a classification of what kind of emotions are provoked in the user. Such information is useful in the field of behavioural targeting and contextual advertising [6].

For this purpose the use of fuzzy rules and fuzzy classifier to analyze the sentimental content for whole sentences is examined. In the matter of matching of Polish words a modified Levenshtein dynamic programming algorithm for words similarity comparison is proposed. These technique combined together provides an interesting solution, whose verification is a motivation for conducting this examination. Nevertheless, providing a proof, that the modified Levenshtein distance can be beneficial for suffix-mitigated words identification, is also a key reason for this research.

Sentence sentiment classification system development

The study was entirely conducted in the Matlab simulation environment, using Fuzzy Logic Toolbox [7].

Knowladge Database

As a result of scientific research in [3, 4], a database of Polish words with assigned sentimental value has been created. In the database, every word is described by four parameters:

- Neutral, determines whether a given word carries a sentimental value at all.
- GeneralNegPos, specifies the type of sentimental value of a word.
- PreciseNegPos, determines the degree of sentimental value expression.
- SOPMI, Semantic orientation, defines the semantic orientation of a word with relation to the environment.

All parameters values were normalized from 0 to 1, where the sentimental meaning of 0 refers to Negative and of 1 refers to Positive. These parameters are, however, often mutually exclusive. This is due to imperfect extraction methods and, as it is emphasized in [3], obtained sentimental scores are sometimes ambiguous.

Fuzzy Word Matching

The procedure of fuzzy words matching is based on a modified algorithm of the Levenshtein distance [8]. The proposed by W. Levenshtein metric of dissimilarity of strings (words) has been modified by accounting for the word quasi core constancy and the distance of examined characters from the beginning of the word. For simplicity, the first three letters of a word are regarded as the quasi core. If any of these first three letters in a compared pair of words turns out to be different, the compliance value will be equal to zero. Otherwise, the modified Levenshtein distance will be calculated (see Figure 1). For normalization purpose the result is divided by number of letters in the longer word.

Taking into consideration that in the Polish grammar, the result of conjugation and declination mostly affects the suffix of a word, the compliance of compared letters is weighted by the distance of this letter from the beginning of the word. Therefore, words' similarity calculation will be less affected by the differences between the last letters than the differences between letters in the middle of the word. As noted in [9], the weight modification of Levenshtein

algorithm results in obtaining a function not being a metrics (in the strict sense). However, as far as the recognition of Polish words is concerned, this modification allows for a better matching of inflected words.

```
double SuffixMitigatedLevenshtein(char w1[1..x], char w2[1..y])
   declare double matrix[0..x, 0..y]
   for i from 0 to x
         matrix[i, 0] := i
   for j from 0 to y
         matrix[j, 0] := j

   for i from 1 to x
         for j from 1 to y
   if w1[i] = w2[j] then cost := 0 else cost := 1
   cost := cost * x / j//suffix mitigation modification
   matrix[i, j] := minimum( matrix[i-1, j] + 1,
                            matrix[i, j-1] + 1,
                            matrix[i-1, j-1] + cost)
return matrix[x, y]
```

Fig.1. Pseudo-code for modified algorithm of Levenshtein distance calculation

For example, the comparison result of the word *rozwiązać* (Eng. solve) with *rozwinąć* (Eng. develop) is equal to *0.6667*, whilst the comparison of the word *rozwiązać* with its conjugated forms *rozwiązali*, *rozwiąż* and *rozwiązałem* yielded results of respectively *0.8000*, *0.7778* and *0.8182*. The fuzzy word matching procedure is performed for each of examined words in order to find words with assigned sentimental value. Table 1 presents matching results for strings in which letters change on diverse positions. Reference string is “ooooooooo” while character ‘x’ simulates the differences between letters.

Table 1. Results for strings comparison in reference to “ooooooooo”, in which changes of characters occurs

String	Matching score	String	Matching score
“ooooooooo”	1.0000	“oooooooxx”	0.7889
“oooooooox”	0.9000	“ooooooxxx”	0.6639
“oooooooxo”	0.8889	“oooooxxxx”	0.5210
“ooooooxoo”	0.8750	“ooooooxxo”	0.7639
“oooooxooo”	0.8571	“oooooxxxo”	0.6210
“ooooxoooo”	0.8333	“ooooxxxxo”	0.4544
“oooxooooo”	0.8000	“ooooooooox”	0.9091
“ooxoooooo”	0.8000	“oooooooooxx”	0.8333
“oxoooooooo”	0.0	“ooooooooo”	0.9000
“xoooooooo”	0.0	“oooooooo”	0.8000
“xoooooooo”	0.0	“ooooxxxxx”	0.1544

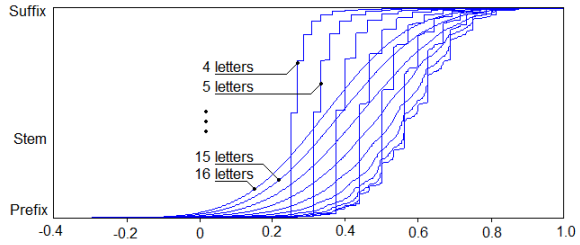

Fig. 2. Words similarity comparison by using normalized SuffixMitigatedLevenshtein method for all possible combination of words ranging in size from 4 to 16 letters

Graphical representation of the words matching functions for normalized SuffixMitigatedLevenshtein method for all possible combination of words ranging in size from 4 to 16 letters is presented in Figure 2. Differences in the letters in the suffix part of the word have a smaller impact on the similarity estimation. For shorter words, the matching functions have staircase shape, where any difference in the letters at the stem side significantly affects on similarity evaluation. Naturally, for longer words the matching function is smoother, and essentially takes a sigmoidal shape. Side effect of weight modification of Levenshtein algorithm results in obtaining negative values for words comparison. However, this does not matter, while only positive values are taken into consideration.

Fuzzy sentiment classifier

To mitigate some sentimental parameter ambiguities, the result of the entire sentence is processed by the fuzzy classifier (see Figure 3), where the classification decision is achieved by using linguistic variables: Negative, Neutral or Positive. Fuzzy classifier implements the Mamdani fuzzy model which is based on a set of decision rules and the usage of linguistic operators [10].

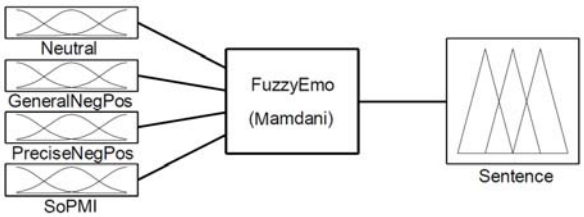

Fig. 3. Fuzzy Inference System for the sentiment classification

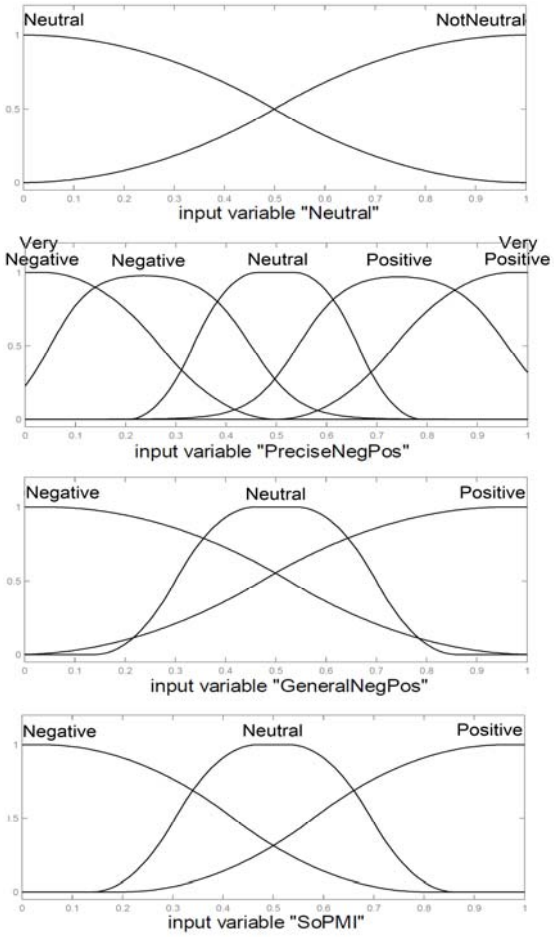

Fig. 4. Membership functions for fuzzification stage

Input fuzzification stage

Numerical values related to a linguistic variable are subject of fuzzification, whereby they are mapped to a specified fuzzy set. Input stage "Neutral" has two fuzzy sets (NotNeutral and Neutral). Input parameters "SoPMI" and "GeneralNegPos", have three fuzzy sets each (Negative, Neutral and Positive). On the contrary, the input stage "PreciseNegPos" contains 5 fuzzy sets (Very Negative,

Negative, Neutral, Positive and Very Positive). The shapes of membership functions for all fuzzification stages are presented in Figure 4. The notation of fuzzy sets for each attribute is derived from the words sentimental database. In contrast, the shapes of membership functions were chosen manually to provide rather evenly coverage.

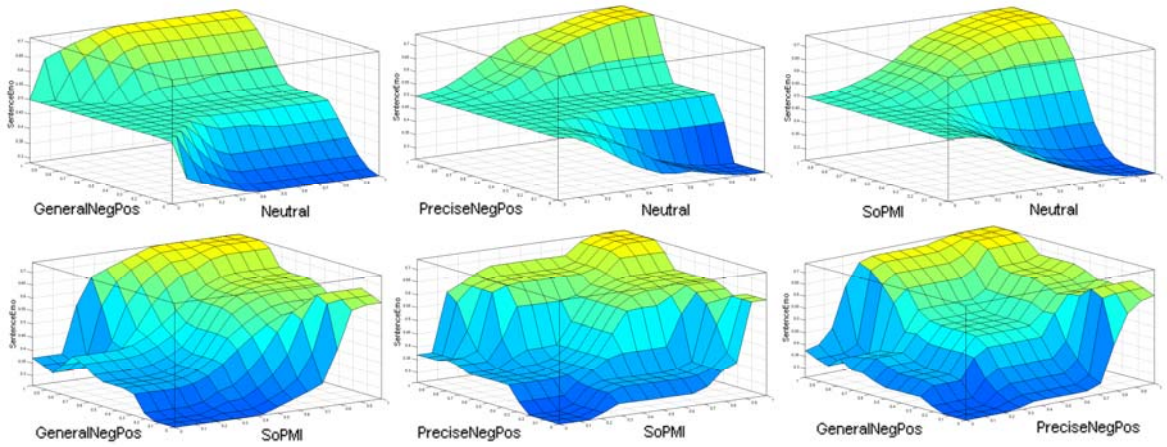

Fig. 5. Decision-making surfaces, which is depend on the system of two input parameters

Output defuzzification stage

By the aggregation of all conclusion rules, the membership of (Output) model is obtained, which determines the final fuzzy set: Negative, Neutral or Positive.

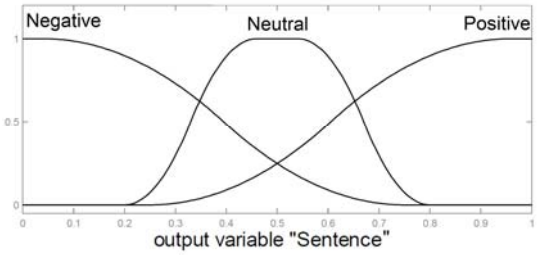

Fig. 6. Membership functions for defuzzification stage

The output fuzzy set is mapped onto a crisped value using the Centroid defuzzification technique. The shape of output membership function is presented in Figure 6.

Fuzzy inference stage

At this stage, input parameters are processed by a set of rules. As a result, a suitable fuzzy set, which is a conclusion of adopted decision rules, is found. The presented classifier uses 15 decision rules, and assigns five rules in order to determine the sentimental meaning (Negative, Neutral or Positive) of the examined sentence. Rules established for fuzzy classification of the sentence sentimental value:

1.(Neutral==Neutral) & (GeneralNegPos==Neutral) & (PreciseNegPos==Neutral) & (SoPMI==Neutral) => (Sentence=Neutral) (1)
2.(Neutral==Neutral) & (GeneralNegPos==Neutral) & (PreciseNegPos==Negative) &(SoPMI==Neutral) => (Sentence=Neutral) (1)
3.(Neutral==Neutral) & (GeneralNegPos==Neutral) & (PreciseNegPos==Positive) &(SoPMI==Neutral) => (Sentence=Neutral) (1)
4.(Neutral==Neutral) & (GeneralNegPos==Neutral) & (PreciseNegPos==Neutral) & (SoPMI==Positive) => (Sentence=Neutral) (1)
5.(Neutral==Neutral) & (GeneralNegPos==Neutral) & (PreciseNegPos==Neutral) &(SoPMI==Negative) => (Sentence=Neutral) (1)
6.(Neutral==NotNeutral) & (GeneralNegPos==Negative) & (PreciseNegPos==VeryNegative) & (SoPMI==Negative) => (Sen-tence=Negative) (1)
7.(Neutral==NotNeutral) & (GeneralNegPos==Negative) & (PreciseNegPos==Negative) & (SoPMI==Negative) => (Sentence=Negative) (1)
8.(Neutral==NotNeutral) & (GeneralNegPos==Negative) & (PreciseNegPos==VeryNegative) & (SoPMI==Neutral) => (Sentence=Negative) (1)
9.(Neutral==NotNeutral) & (GeneralNegPos==Neutral) & (PreciseNegPos==VeryNegative) & (SoPMI==Negative) => (Sen-tence=Negative) (1)
10. (Neutral==NotNeutral) & (GeneralNegPos==Positive) & (PreciseNegPos==VeryPositive) & (SoPMI==Positive) => (Sentence=Positive) (1)
11. (Neutral==NotNeutral) & (GeneralNegPos==Positive) & (PreciseNegPos==Positive) & (SoPMI==Positive) => (Sentence=Positive) (1)
12. (Neutral==NotNeutral) & (GeneralNegPos==Positive) & (PreciseNegPos==VeryPositive) & (SoPMI==Neutral) => (Sentence=Positive) (1)
13. (Neutral==NotNeutral) & (GeneralNegPos==Neutral) & (PreciseNegPos==VeryPositive) & (SoPMI==Positive) => (Sentence=Positive) (1)
14. (Neutral==NotNeutral) & (GeneralNegPos==Positive) & (PreciseNegPos==Neutral) & (SoPMI==Positive) => (Sentence=Positive) (1)
15. (Neutral==NotNeutral) & (GeneralNegPos==Negative) & (PreciseNegPos==Neutral) & (SoPMI==Negative) => (Sentence=Negative) (1)

Dependencies between inference rules related to four input parameters form five-dimensional decision-making space, which is impossible to visualize directly. Therefore, Figure 5 presents (output) decision surface depending on the system of two Input parameters (other inputs are constant) in the three-dimensional space. Despite different shapes of decision-making surfaces, it is possible to distinguish the minima and maxima associated with the sentimental classification of words as Negative and Positive. Absolute minimum value obtained from fuzzy classifier is equal to *0.23* and absolute maximum value is equal to *0.77*. Transient states related to the Neutral sentiment class are also possible to differentiate and are roughly equal to *0.5*. The Decision-making surfaces have limited output value of fuzzy classifier to range *0.5±0.27*. Therefore, for better interpretation the result obtained from fuzzy classifier may possibly be scaled to the range *0.5±0.5*.

Sentence sentiment estimation

A particular sentence is considered here as group of words related to each other and acting as information medium understood only in the context of the entire sentence. Hence, the estimation of sentimental meaning will be based on the identification of sentimental parameters of single words occurring in the analyzed sentence and the calculation of the arithmetic mean. The obtained value is yet weighted (see formula (1)) by the word similarity metric derived from the modified algorithm of Levenshtein distance calculation (*SuffixMitigatedLevenshtein)*.

$$FuzzyIN = \frac{\left(\sum_{i=1}^{NoSw} WsP(i) \cdot SML(i)\right)}{NoSw} \quad (1)$$

where: *FuzzyIN*- Four inputs parameters for fuzzy sentiment classifier; *NoSw*- Number of the sentimental words in the examined sentence; *WsP*- Words sentimental parameters (four parameters) obtained from sentimental database [3] [4]; *SML*- Words similarity factor (scalar) obtained from normalized *SuffixMitigatedLevenshtein* procedure.

For example, the word *delikatnie* (Eng. gently) is matched with the reference word *delikatny* with similarity factor equal to *0.8*. For the word *delikatny* the parameters: Natural, GeneralNegPos, PrecisNegPos and SoPMI assume normalized values of *1.0*, *1.0*, *0.25* and *0.31* respectively. However, for the word *delikatnie* these values will be adjusted according to the modified Levenshtein distance and will assume values of *0.8*, *0.8*, *0.2* and *0.248* respectively. This procedure leads to the extinction of the sentimental value by increasing the difference between the words. When similarity factor tends to zero, parameters zeroing is performed which reinforces the "Neutral" set classification. The values of sentimental parameters calculated for each word in the sentence are grouped together and the arithmetic mean is calculated. The mean constitutes the input (four parameters) for fuzzy sentiment classifier.

Experiment

In the case study, some sentences in Polish were tested to verify the presented methodology. According to formula (1) each sentimental value parameter is multiplied by word similarity factor. Obtained results are presented in Table 2 and 3.

Positive sentence:

„W naszym domu rodzinnym było ciepło, życzliwie, troskliwie, chociaż bez luksusów"

(Eng. In our *family* home it was *warm*, *kindly*, *lovingly*, although without the *luxuries*.)

Table 2. Summary of the word matching and related to it the sentimental value

Estimator	*rodzina*	*ciepły*	*życzliwy*	*troskliwy*	*luksusowy*	Average
WordSimilarity	0.667	1.000	0.778	1.000	0.889	0.8667
Neutral	0.167	1.000	0.889	1.000	0.944	0.8000
GeneralNegPos	0.500	1.000	0.778	1.000	0.944	0.8667
PreciselNegPos	0.500	0.500	0.889	0.750	0.500	0.6278
SO-PMI	0.720	0.850	0.889	0.300	0.588	0.6696

Result of fuzzy sentiment classification: **Positive 0.7626**

Negative sentence:

„Mam dosyć infantylnych, egocentrycznych, pozbawionych empatii, brzydkich facetów, którzy nie potrafia dorosnąć"

(Eng. I've enough of *infantile*, *egocentric*, *devoid* of empathy, *ugly guys* who aren't able to *grow up*)

Table 3. Summary of the word matching and related to it the sentimental value

Estimator	*infantylny*	*egocentryczny*	*pozbawić*	*brzydki*	*facet*	*dorosnąć*	Average
WordSimilar.	0.833	0.867	0.667	0.778	0.714	1.000	0.810
Neutral	0.917	0.933	0.167	0.889	0.143	1.000	0.675
GenNegPos	0.083	0.067	0.500	0.111	0.143	0.000	0.151
PrecNegPos	0.083	0.067	0.167	0.111	0.143	0.000	0.095
SO-PMI	0.416	0.327	0.313	0.788	0.614	0.140	0.433

Result of fuzzy sentiment classification: **Negative 0.2337**

For sentence sentimental value estimation are considered only words with similarity factor larger than 0.6. This threshold is selected to assure proper word stem identification (see Figure 2). The average values of the estimators constitute corresponding inputs for fuzzy classifier. Obtained results for both sentences examples have almost reached the limits of decision-making surface, which indicates very high confidence in the judgment.

Discussion

The presented method of word matching and fuzzy classification of the sentimental meaning of sentences and paragraphs seems to be very promising. It exhibits a high degree of resistance to certain classification ambiguities of individual words in a sentence. Be that as it may, the method is at the upper layer of decision-making and is dependent on the proper classification of the sentimental parameters of each word. If a sentimental words database contains major errors, they will also affect the higher classification system intended for the sentimental sentence tagging. Hence, care should be taken to set up a sentimental words database correctly with appropriate parameters. The examined database of the sentimental words contains more than 3700 words, but still this number should be increased or even doubled for more precise sentiment recognition. Since word matching method based on the modified algorithm of Levenshtein distance is computationally complex - proportional to the product of string lengths of the words under comparison, it is reasonable to replace the algorithm with equivalent Look-Up-Table. For words ranging in size from 4 to 16 letters (see Figure 2) all possible words matching combination are covered by LUT with 131072 entries. Considering that the algorithm is to be used for words matching, the maximum length of compared strings is limited to the longest word in the dictionary. This improvement should significantly boost the application speed.

Conclusion

In the presented work a modified Levenshtein distance algorithm for suffix-mitigated words identification is investigated. The fuzzy sentiment classifier combined with fuzzy word matching algorithm proved its suitability for a proper sentence sentiment classification. Presented case study describes in details the process for sentences sentimental estimation. Nevertheless, more experiments have to be conducted with the use of more sophisticated sentimental words databases. This should also be the direction for further research. The proposed modification in Levenshtein distance seems to be useful for word matching for languages which conjugation and declination pertains mostly to the word suffix. Although the presented solution is intended for the Polish language, presumably it may be also applied in other languages.

REFERENCES

[1] Transactions, *IEEE Trans. Magn.*, 50 (2002), No. 5, 133-137
Cambria E., Havasi C., Hussain A.: SenticNet 2: A semantic and affective resource for opinion mining and sentiment analysis. In: AAAI FLAIRS, pp. 202-207, Marco Island 2012.

[2] Kim S.M., Hovy E.: Determining the sentiment of opinions. In Proceedings of the 20th International Conference on Computational Linguistics, ser. COLING '04. USA: Association for Computational Linguistics, pp. 1367-1373. Stroudsburg, PA, 2004.
[3] Wawer A.: Mining Co-Occurrence Matrices for SO-PMI Paradigm Word Candidates. In Proceedings of the Student Research Workshop at the 13th Conference of the European Chapter of the Association for Computational Linguistics, EACL'12 SRW, pp. 74-80. Avignon, France, April 2012.
[4] Wawer A., Rogozinska D.: How Much Supervision? Corpus-based Lexeme Senti-ment Estimation. In Data Mining Workshops, 2012 IEEE 12th International Confer-ence on. SENTIRE, ICDMW, pp. 724-730. Los Alamitos, USA, IEEE Computer So-ciety. 2012.
[5] Nair R., Tambe M., Marsella S.: The role of emotions in multiagent teamwork, in Who Needs Emotions? The Brain Meets the Robot, pp. 311-329. Oxford University Press, 2004.
[6] Li T., Liu N., Yan J., Wang G., Bai F., Chen Z.: A Markov chain model for integrating behavioral targeting into contextual advertising, KDD Workshop on Data Mining and Audience Intelligence for Advertising, pp. 1-9. ACM, 2009.
[7] Sivanandam, S. N., S. Sumathi, and S. N. Deepa.: Introduction to Fuzzy Logic Using MATLAB. Berlin: Springer, 2007.
[8] Levenshtein A.: Binary Codes Capable of Correcting Deletions, Insertions and Reversals. Soviet Physics Doklady, vol. 10. no. 8, pp. 707-710. 1966.
[9] Weigel A., Fein F.: Normalizing the Weighted Edit Distance. Proc. 12th IAPR Int'l Conf. Pattern Recognition, vol. 2, Conf. B: Computer Vision and Image Processing, pp. 399-402, Oct. 1994.
[10 Mamdani E. H.: Application of fuzzy logic to approximate reasoning using linguistic synthesis. IEEE Trans. Computers 26(12), pp. 1182-1191. 1977.

***Author**: mgr inż. Marcin Pietras, West Pomeranian University of Technology, Faculty of Computer Science and Information Technology, Department of Methods of Artificial Intelligence and Applied Mathematics, ul. Żołnierska 49, 71-210 Szczecin, Poland. E-mail: mpietras@wi.zut.edu.pl*