Folia Oeconomica
Acta Universitatis Lodziensis
ISSN 0208-6018 e-ISSN 2353-7663
www.czasopisma.uni.lodz.pl/foe/
2(353) 2021

DOI: https://doi.org/10.18778/0208-6018.353.03

Adam Piotr Idczak
University of Łódź, Faculty of Economics and Sociology, Department of Statistical Methods
Łódź, Poland, adam.idczak@uni.lodz.pl

Sentiment Classification of Bank Clients' Reviews Written in the Polish Language

Abstract: It is estimated that approximately 80% of all data gathered by companies are text documents. This article is devoted to one of the most common problems in text mining, i.e. text classification in sentiment analysis, which focuses on determining the sentiment of a document. A lack of defined structure of the text makes this problem more challenging. This has led to the development of various techniques used in determining the sentiment of a document. In this paper, a comparative analysis of two methods in sentiment classification, a naive Bayes classifier and logistic regression, was conducted. Analysed texts are written in the Polish language and come from banks. The classification was conducted by means of a bag-of-n-grams approach, where a text document is presented as a set of terms and each term consists of n words. The results show that logistic regression performed better.

Keywords: sentiment analysis, opinion mining, text classification, text mining, logistic regression, naive Bayes classifier

JEL: C81, M31

1. Introduction

Approximately 80% of all data gathered by companies has textual form (Sullivan, 2001), such as e-mails, memos, reports, research, reviews, strategy, and marketing plans, etc. All of these textual forms provide a rich and extensive source of valuable (but undiscovered) information. The amount of available data is overwhelming, hence analysing data manually by analysts might be ineffective or even impossible. On the other hand, such a collection of data cannot be processed with typical techniques because of their unstructured form. Fortunately, there are several *text mining* applications available for deriving high-quality information from text documents. This creates an opportunity to take advantage of data to improve decision-making processes in companies.

Text classification in *sentiment analysis* is one of text mining applications which can provide answers to questions such as: "Do clients like my product (or service)?" or "Which aspects of my product (or service) do clients like or not?" It is also helpful in tracking and evaluating customer satisfaction. This type of text analysis focuses on detecting an author's attitude (called *sentiment*) toward entities and their attributes.

In this paper, sentiment classification of bank clients' reviews written in the Polish language is examined in a comparative analysis of two methods. In Section 2, sentiment analysis and document sentiment classification are introduced. The next section presents the idea of a bag-of-n-gram approach, a naive Bayes classifier and logistic regression. Section 4 contains an algorithm for the evaluation of the above-mentioned methods, a data overview, and the results of the comparison conducted. Finally, conclusions are stated at the end.

2. Sentiment analysis

Sentiment analysis (*opinion mining*) focuses on analysing textual data in order to assess an author's attitude toward entities and their attributes. This type of analysis is interdisciplinary in its nature, as it combines research and applications in such fields as: *natural language processing* (NLP), *data mining, web mining*, and *information retrieval*. It is presumed that the terms *sentiment analysis* and *opinion mining* were first introduced in (Dave, Lawrence, Pennock, 2003; Nasukawa, Yi, 2003) respectively, but research regarding *sentiment* and *opinion* emerged a few years earlier (Wiebe, 2000; Das, Chen, 2001; Tong, 2001; Morinaga et al., 2002; Pang, Lee, Vaithyanathan, 2002; Turney, 2002).

It is worth mentioning that there is no clear distinction between *sentiment analysis* and *opinion mining* among researchers and practitioners. In this paper, these two terms will be used interchangeably.

Sentiment analysis can be performed with respect to its granularity level (Liu, 2015):

1) document level – the objective is to classify a whole opinion document into *positive* or *negative sentiment*;
2) sentence level – the main task is to assign sentiment (positive or negative) to each sentence. Sentences without an opinion are considered as *neutral*;
3) aspect level – this type of analysis is focused on finding opinions concerning entities or their aspects and then assigning sentiment to them; for example, opinion *I love this restaurant, but the prices are too high* has overall positive sentiment, but it does not mean that the author of the opinion is positive about all aspects of the restaurant; thus, to obtain such details, one needs to apply aspect level analysis.

2.1. Document sentiment classification

Document sentiment classification is one of the most studied topics in the field of sentiment analysis. Its task is to assess the overall sentiment about an entity based on the opinion document evaluating the entity. In other words, the goal of document sentiment classification is to assign one label (positive, negative or neutral) to a document. Document sentiment classification does not take into account all aspects in the opinion document or seek sentiments regarding them, hence it is considered as document level analysis. There is a great deal of research devoted to sentiment classification studying various types of data and various types of techniques. Turney (2002) used the data from Eopinios.com website that contain reviews sampled from four domains: reviews of cars, banks, movies, and travel destinations. He calculated Semantic Orientation (*SO*) of a term by means of the number of hits returned from the query engine[1] with the reference to words *poor* and *excellent*:

$$SO(term) = \log_2 \left(\frac{hits(term\ NEAR\ \text{"excellent"})\, hits(\text{"}poor\text{"})}{hits(term\ NEAR\ \text{"poor"})\, hits(\text{"}excellent\text{"})} \right). \tag{1}$$

The document is labelled as positive if averaged *SO* was positive, otherwise the document was labelled as negative. Pang, Lee, and Vaithyanathan (2002) used film reviews from the Internet Movie Database (IMDb). Their study utilises mostly unigrams and bigrams with term presence as features. Na, Khoo, and Wu (2005) examined unigrams and unigrams with part-of-speech (POS) tags with different weighting schemes (term presence, term frequency, and term frequency inverse document frequency) using on-line product reviews downloaded from the Review Centre (https://www.reviewcentre.com/). Many researchers appreciate messages (*tweets*) from Twitter as a source of data, e.g. Asur and Huberman (2010) classified

1 AltaVista Advanced Search engine.

film reviews (tweets) from Twitter using an n-grams approach in order to improve forecasting box-office revenue of movies. Tweets regarding the Irish Great Election in 2011 were utilised in a uni-gram approach. Hanbury and Nopp (2015) employ sentiment analysis in risk assessment for Eurozone banks. The authors evaluated CEO letters and Outlook sections (usually part of management report) by means of sentiment finance-oriented words. Such a finance-specific list of words comes from Loughran and McDonald's (2011) work. Selected studies with methods and accuracy are given in the Table 1.

Table 1. Selected studies on sentiment classification

No.	Author/Authors	Data set	Method	Accuracy (%)
1	Turney (2002)	Reviews of: – cars – banks – films – tours	Semantic Orientation	 84.0 80.0 65.8 70.5
2	Pang, Lee, and Vaithyanathan (2002)	Film reviews	NB[a]	81.0[g]/77.3[h]
			ME[b]	80.4[g]/77.43[h]
			SVM[c]	82.9[g]/77.13[h]
3	Na, Khoo, and Wu (2005)	On-line product reviews	SVM[c]	75.5[g]
4	Asur and Huberman (2010)	Tweets with film reviews	DynamicLMClassifier	98.0
5	Bermingham and Smeaton (2011)	Tweets regarding the Irish Great Election in 2011.	MNB[d] ADA-MNB[e] SVM[c] ADA-SVM[f]	62.94 65.09 64.82 64.28
6	Hanbury and Nopp (2015)	CEO letters	NB[a]	70.3[i]/75.0[j]
			SVM[c]	70.3[i]/79.2[j]
		Outlook sections of Eurozone banks	NB[a]	56.3[i]/70.4[j]
			SVM[c]	70.3[i]/70.4[j]

[a] Naive Bayes.
[b] Maximum Entropy.
[c] Support Vector Machines.
[d] Multinominal Naive Bayes.
[e] Adaboost M1 Multinominal Naive Bayes.
[f] Adaboost M1 Support Vector Machines.
[g] Unigram (binary).
[h] Bigram (binary).
[i] Lexicon-based approach.
[j] Document frequency and information gain.

Source: own elaboration

3. Classification algorithms

To employ a particular classification algorithm, the opinion documents analysed were expressed in bag-of-n-grams fashion. In this kind of document representation, a document consists of a set of terms (features) where n stands for the number of words in this particular term, e.g. uni-gram, bi-gram, etc. Given this, the documents can be presented as the following document-term matrix (DTM):

$$\boldsymbol{x} = \left[x_{ij}\right], \tag{2}$$

where:
x – is the document-term matrix,
x_{ij} – is the number of times that the j-th term occurred in the i-th document,
$i = 1, \ldots, I$ (I is the total number of documents in a training set),
$j = 1, \ldots, J$ (J is the total number of terms in a training set).

Features from matrix (2) can be transformed in various ways (Pang, Lee, Vaithyanathan, 2003; Na, Khoo, Wu, 2005):

1) term presence (binary):

$$x_{ij}^{*} = \begin{cases} 0, & when\ x_{ij} = 0 \\ 1, & when\ x_{ij} > 0 \end{cases}, \tag{3}$$

2) term frequency (TF):

$$x_{ij}^{*} = x_{ij}, \tag{4}$$

3) term frequency inverse document frequency (TFIDF):

$$x_{ij}^{*} = \begin{cases} 0, when\ x_{ij} = 0 \\ (1 + \log\left(x_{ij}\right)) * \log\left(\dfrac{I}{df_j}\right), when\ x_{ij} > 0 \end{cases}, \tag{5}$$

where:
I – is the number of all documents,
df_j – is the number of documents where the j-th term occurred.

3.1. Naive Bayes

Bayes' rule (Domański, Pruska, 2000) for document sentiment classifications defines conditional probability that the $\boldsymbol{x}_i$ document belongs to the C_k class:

$$\mathrm{P}(C_k|\boldsymbol{x}_i) = \frac{p_k f(\boldsymbol{x}_i \mid C_k)}{\sum_{k=1}^{K} p_k f(\boldsymbol{x}_i \mid C_k)}, \tag{6}$$

where:
C_k – is the k-th class, $k = 1, \ldots, K$,
$\boldsymbol{x}_i$ – is the i-th document with J features,
p_k – is the *a priori* probability that the document belongs to the C_k class,
$f(\boldsymbol{x}_i|C_k)$ – is a probability of occurrence of the $\boldsymbol{x}_i$ document, given it belongs to the C_k class.

A naive Bayes (NB) classifier assigns the $\boldsymbol{x}_i$ document to the class C_k if equation (7) is satisfied:

$$\mathrm{P}(C_k|\boldsymbol{x}_i) = \max_k \mathrm{P}(C_k|\boldsymbol{x}_i), \tag{7}$$

which is equivalent for:

$$\mathrm{P}(C_k|\boldsymbol{x}_i) = \max_k \left[p_k f(\boldsymbol{x}_i|C_k)\right]. \tag{8}$$

The above-mentioned classification rule assumes that terms $\boldsymbol{x}_j$ are independently distributed given the k-th class:

$$f(\boldsymbol{x}|C_k) = \prod_{j=1}^{J} f(x_j|C_k). \tag{9}$$

In order to train a naive Bayes classifier, p_k will be calculated using relative-frequency estimation:

$$\hat{p}_k = \frac{n_k}{I}, \tag{10}$$

where n_k is the number of documents given that belong to the k-th class, while $f(\boldsymbol{x}_i|C_k)$ will be calculated using relative-frequency estimation (for term presence or TF):

$$\hat{p}(x_j = x_{ij} \mid C_k) = \frac{n_{ijk}}{n_{jk}}, \tag{11}$$

or fitting a normal distribution (for TFIDF):

$$\hat{f}(x_j|C_k) = \left(\widehat{\sigma_{\mathrm{jk}}}\sqrt{2\pi}\right)^{-1} * \exp\left(-\frac{\left(x_j - \widehat{\mu_{\mathrm{jk}}}\right)^2}{2\widehat{\sigma_{\mathrm{jk}}^2}}\right), \tag{12}$$

where:
n_{ijk} – frequency of the i-th value of the j-th term in the k-th class,
n_{jk} – frequency of the j-th term in the k-th class,

$\widehat{\mu_{jk}}$ – mean of TFIDF for the *j*-th term in the *k*-th class,

$\widehat{\sigma_{jk}}$ – standard deviation of TFIDF for the *j*-th term in the *k*-th class.

3.2. Logistic regression

Let us assume that C be the Bernoulli random variable:

$$C \sim Bernoulli(p), \tag{13}$$

that can take one of two values:

$$C = \begin{cases} 0, \text{ when the sentiment of a document is negative}, \\ 1, \text{ when the sentiment of a document is positve}, \end{cases} \tag{14}$$

then the logistic regression (Hosmer, Lemeshow, Sturdivant, 2013) can be written as follows:

$$p = p(C = 0|\boldsymbol{x}_i) = \frac{e^{\beta_0 + \beta^t \boldsymbol{x}_i}}{1 + e^{\beta_0 + \beta^t \boldsymbol{x}_i}}, \tag{15}$$

where:
β_0 is an intercept and $\boldsymbol{\beta}$ is a vector of estimated parameters.

It is convenient to apply *logit transformation* on (15) to obtain some desirable properties of a linear model:

$$\ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta^t \boldsymbol{x}_i, \tag{16}$$

in particular, the above-mentioned equation is linear in its parameters, hence betas have a handy interpretation in terms of *odds ratio* $\left(\frac{e^{\beta_0 + \beta^t \boldsymbol{x'}_i}}{e^{\beta_0 + \beta^t \boldsymbol{x}_i}}\right) e^{\beta_0 + \beta^t \boldsymbol{x'}_i}$[2], i. e. if the x_j feature increases by 1 unit (ceteris paribus), the odds ratio will increase by e^{β_j}. This means that the odds that a document has positive sentiment (given the increased x_j) has increased (decreased) by $(e^{\beta_j} - 1) * 100\%$.

Probability $p(C = 0|\boldsymbol{x}_i)$ in (15) is a probability that the document $\boldsymbol{x}_i$ has positive sentiment, thus a probability that the document $\boldsymbol{x}_i$ has negative sentiment is calculated by the following equation:

$$p(C = 1|\boldsymbol{x}_i) = 1 - p(C = 0|\boldsymbol{x}_i). \tag{17}$$

The $\boldsymbol{x}_i$ document is classified as negative if the following equation is satisfied:

$$\mathrm{P}(C = 0|\boldsymbol{x}_i) = \max\left[p(C = 0|\boldsymbol{x}_i), p(C = 1|\boldsymbol{x}_i)\right], \tag{18}$$

otherwise, the document is considered as positive.

2 $e^{\beta_0 + \beta^t \boldsymbol{x'}_i}$ denotes the odds for the x_j feature to be increased by 1 unit.

Parameters from equation (15) can be estimated by means of the *maximum likelihood method* by maximising the following equation:

$$\mathrm{L}(\beta) = \prod_{i=1}^{I} p(C_1|\boldsymbol{x}_i)^{C_i} \left[1 - p(C_1|\boldsymbol{x}_i)\right]^{1-C_i}, \tag{19}$$

with respect to parameters β_0 and $\boldsymbol{\beta}$:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmax}}\, \mathrm{L}(\beta). \tag{20}$$

4. Evaluation

4.1. Experimental set-up

In order to evaluate a naive Bayes classifier and logistic regression in document sentiment classification, experiment is conducted in line with the algorithm presented in Figure 1. All calculations are made in R software. First, the documents analysed are read into the memory, and then they are initially processed, i.e. unwanted numbers, punctuations and words are deleted. Also, *lemmatisation* is a very important part of this step. The process of lemmatisation groups together the inflected forms of the word so that they can be analysed as a single item (*word's lemma*), e.g. *płakać* is lemma for *płakał*, *płakaliśmy*, *płacze*. It is especially important in the case of the Polish language, which is inflected. Lemmatisation is done by means of tm package in R. This step can have a crucial impact on features (and on the number of features) in the document-term matrix. For the purpose of this study, unigrams and bigrams will be considered. The DTM matrix is calculated by the use of hashmap, tm and tex2vec package. After the DTM is created, the three versions of the document-term matrix are calculated (binary, TF and TFIDF) employing RWeka and tm package. Then the matrix is used in 10-fold cross validation, according to Figure 1, where a naive Bayes classifier and logistic regression are learnt on a training sample and classification is evaluated on a validation sample. This part of algorithm is handled by e1071 and gmodels package. The classification is evaluated by means of accuracy:

$$accuracy = \frac{TP + TN}{I}, \tag{21}$$

where:
TP – the number of documents with positive sentiment classified as positive,
TN – the number of documents with negative sentiment classified as negative,
I – the number of all documents.

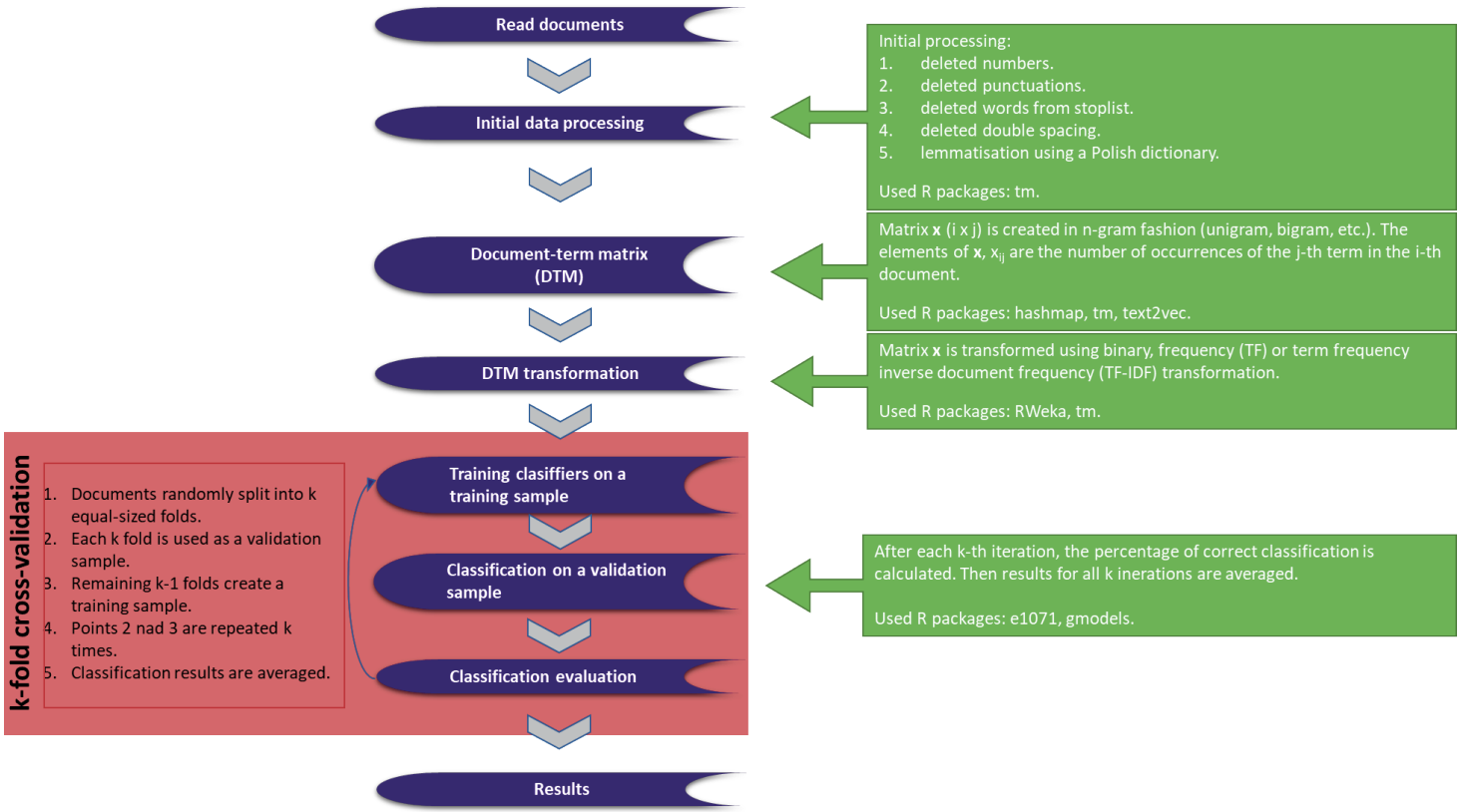

Figure 1. Algorithm
Source: own elaboration

4.2. The data

The data consist of 1,559 documents that are clients' reviews concerning one of Polish banks. Each document is labelled with positive or negative sentiment (positive or negative class). These labels were assigned manually by an opinion holder (by choosing a sad or happy face icon). There were 786 negative and 773 positive documents. Words with the highest frequency in each class (red for negative and green for positive) are shown in Figure 2.

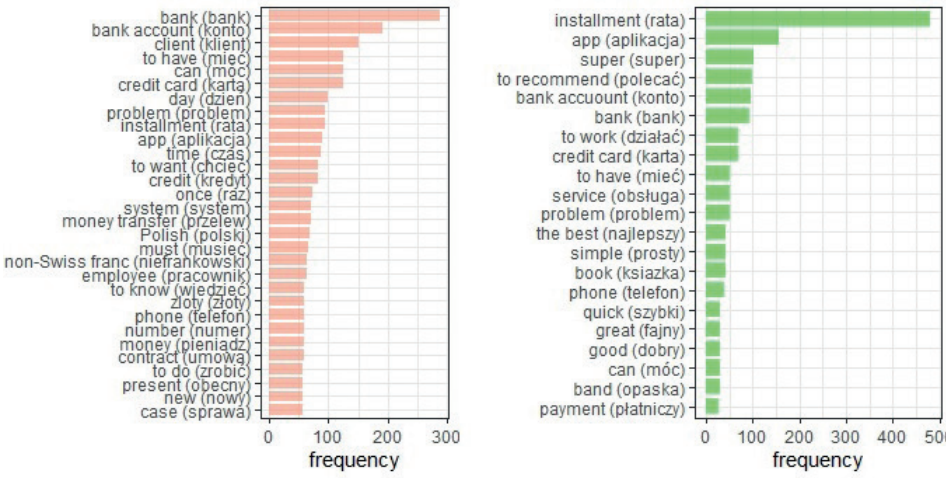

Figure 2. The most frequent words in each class (negative and positive)

Source: own calculations

4.3. Results

Figures 3 and 4 show results of classification of the above-mentioned data set for unigrams and bigrams respectively. Document sentiment classification was conducted by means of naive Bayes classifier (NB) and logistic regression (GLM). It turns out that the considered classification methods outperformed the 50% random-choice baseline and the results ranged from 51.06% to 82.81%. The highest accuracy was observed for logistic regression (unigram DTM with TFIDF) and the lowest was achieved for a naive Bayes classifier (bigram DTM with TFIDF).

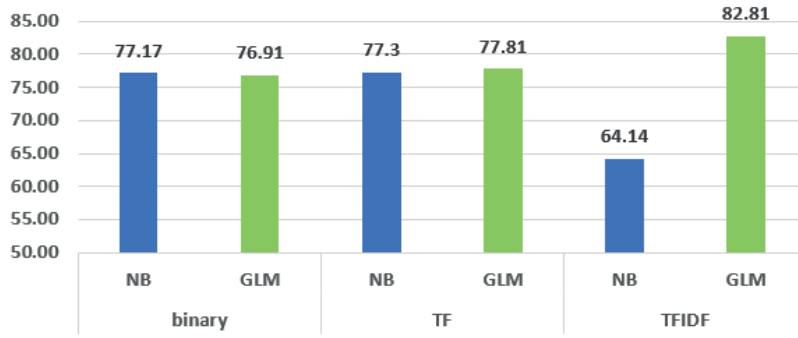

Figure 3. Accuracy (%) of unigrams

Source: own calculations

Results for unigrams are quite similar for binary and TF transformation and range from 76.91% to 77.81% but for TFIDF differences are greater, i.e. a Naive Bayes classifier with TFIDF (64.14%) performs worse than NB and GLM with binary or TF. Also, in terms of accuracy, NB is worse than any logistic regression. In fact, GLM with TFIDF has the highest percentage of correctly classified documents (82.81%).

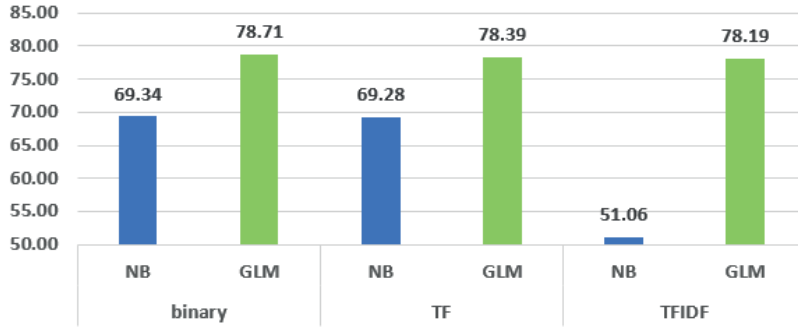

Figure 4. Accuracy (%) of bigrams

Source: own calculations

As for bigrams, logistic regression performed better than a naive Bayes classifier, yielding roughly 78% of correctly classified documents. Accuracy of NB was about –9 p.p. worse than GLM for binary and TF. NB with TFIDF has the lowest accuracy (only 51.06%), yielding performance only about 1 p.p. above the random-choice baseline.

5. Conclusions

In this paper, a naive Bayes classifier and logistic regression were examined in document sentiment classification performed for the Polish language. This problem was found by researchers (Pang, Lee, Vaithyanathan, 2003) to be more challenging than traditional topic-based classification, which concerns keywords that help identify topics. Document sentiment classification is more complex because sentiment (rather than topics) can be expressed in a more subtle manner.

The results produced in section 4.3 indicate that the performance of naive Bayes classifier and logistic regression applied to the customer reviews written in Polish is high. In all cases, the accuracy is higher than the random-choice baseline, and it also fits in the accuracy that the researchers obtained in their studies (see Table 1). Logistic regression with TF-IDF yielded the highest accuracy, i.e. 82.81%.

When it comes to TFIDF transformation, the accuracy for a naive Bayes classifier was undoubtedly poorer than in the case of the other approaches. The reason for such drop in performance is that the distribution of TFIDF features does not necessarily follow the density function when $f(\boldsymbol{x}_i|C_k)$ is a normal distribution.

It is worth mentioning that estimates of parameters of the above-mentioned methods are highly influenced by sparsity of the DTM matrix. Thus, performance of considered classifiers is driven by non-occurrence rather than occurrence of features obtained from the training set. Saif, He and Alani (2012) proposed two effective approaches to deal with sparsity of the DTM matrix.

The results (considered as high in terms of accuracy) presented in this article cannot be generalised to all types of documents written in the Polish language due to the fact that: (1) each type of data has its own specific way of expressing sentiment, (2) most of document sentiment classification research is conducted on documents written in the English language, whereas the Polish language is inflected, which affects the DTM matrix and can possibly add some complexity to expressing the sentiment. All in all, it seems that more studies on documents in the Polish language are needed.

References

Asur S., Huberman B.A. (2010), *Prediction the Future with Social Media*, https://www.researchgate.net/publication/45909086_Predicting_the_Future_with_Social_Media [accessed: 10.02.2021].

Bermingham A., Smeaton A.F. (2011), *On Using Twitter to Monitor Political Sentiment and Predict Election Results*, "Proceedings of the Workshop on Sentiment Analysis where AI meets Psychology (SAAIP)", pp. 2–10, https://www.aclweb.org/anthology/W11-3702.pdf [accessed: 10.02.2021].

Das S., Chen M. (2001), *Yahoo! For Amazon: Extracting Market Sentiment from Stock Message Boards*, "Proceedings of APFA–2001".

Dave K., Lawrence S., Pennock D.M. (2003), *Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews*, "Proceedings of International Conference

on World Wide Web (WWW–2003)", https://www.researchgate.net/publication/2904559_Mining_the_Peanut_Gallery_Opinion_Extraction_and_Semantic_Classification_of_Product_Reviews [accessed: 10.02.2021].

Domański Cz., Pruska K. (2000), *Nieklasyczne metody statystyczne*, PWE, Warszawa.

Hanbury A., Nopp C. (2015), *Detecting Risks in the Banking System by Sentiment Analysis*, "Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing", pp. 591–600, https://www.aclweb.org/anthology/D15-1071.pdf [accessed: 15.02.2021].

Hosmer D.W., Lemeshow S., Sturdivant R.X. (2013), *Applied Logistic Regression*, 3rd ed., John Wiley & Sons, New Jersey.

Liu B. (2015), *Sentiment Analysis. Mining Opinions, Sentiments, and Emotions*, Cambridge University Press, New York.

Loughran T., McDonald B. (2011), *When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks*, "Journal of Finance", vol. 66, no. 1, pp. 35–65, https://www.uts.edu.au/sites/default/files/ADG_Cons2015_Loughran%20McDonald%20JE%202011.pdf [accessed: 19.02.2021].

Morinaga S., Yamanishi K., Tateishi K., Fukushima T. (2002), *Mining Product Reputations on the Web*, "Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD–2002)", https://www.researchgate.net/publication/200044311_Mining_product_reputations_on_the_Web [accessed: 10.02.2021].

Na J.Ch., Khoo C., Wu P.H.J. (2005), *Use of negation phrases in automatic sentiment classification of product reviews*, "Library Collections, Acquisitions & Technical Services", no. 29, pp. 180–191, https://ccc.inaoep.mx/~villasen/bib/Use%20of%20negation%20phrases%20in%20automatic%20sentiment%20classification.pdf [accessed: 11.02.2021].

Nasukawa T., Yi J. (2003), *Sentiment Analysis: Capturing Favorability Using Natural Language Processing*, "Proceedings of the K-CAP–03, 2nd International Conference on Knowledge Capture", pp. 70–77, https://www.researchgate.net/publication/220916772_Sentiment_analysis_Capturing_favorability_using_natural_language_processing [accessed: 15.02.2021].

Pang B., Lee L., Vaithyanathan S. (2002), *Thumbs up? Sentiment Classification using Machine Learning Techniques*, "Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)", pp. 79–86, https://www.cs.cornell.edu/home/llee/papers/sentiment.pdf [accessed: 8.02.2021].

Review Centre, https://www.reviewcentre.com/ [accessed: 25.02.2021].

Saif H., He Y., Alani H. (2012), *Alleviating data sparsity for Twitter sentiment analysis*, [in:] *2nd Workshop on Making Sense of Microposts (#MSM2012): Big things come in small packages at the 21st International Conference on the World Wide Web (WWW'12), 16 Apr 2012, Lyon, France*, CEUR Workshop Proceedings (CEUR-WS.org), pp. 2–9, https://www.researchgate.net/publication/228450062_Alleviating_Data_Sparsity_for_Twitter_Sentiment_Analysis [accessed: 25.02.2021].

Sullivan D. (2001), *Integrating Data and Document Warehouses*, "DM Review Magazine", http://www.dmreview.com/article_sub_articleId_3697.html [accessed: 18.02.2021].

Tong R.M. (2001), *An Operational System for Detecting and Tracking Opinions in on-Line Discussion*, "Proceedings of SIGIR Workshop on Operational Text Classification".

Turney P. D. (2002), *Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews*, "Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)", pp. 417–424, https://www.researchgate.net/publication/248832100_Thumbs_Up_or_Thumbs_Down_Semantic_Orientation_Applied_to_Unsupervised_Classification_of_Reviews [accessed: 22.02.2021].

Wiebe J. (2000), *Learning Subjective Adjectives from Corpora*, "Proceedings of National Conference on Artificial Intelligence (AAAI–2000)", pp. 735–740, https://www.aaai.org/Papers/AAAI/2000/AAAI00-113.pdf [accessed: 13.02.2021].

Analiza sentymentu na podstawie polskojęzycznych recenzji klientów banku

Streszczenie: Szacuje się, że około 80% wszystkich danych gromadzonych i przechowywanych w systemach informacyjnych przedsiębiorstw ma postać dokumentów tekstowych. Artykuł jest poświęcony jednemu z podstawowych problemów textminingu, tj. klasyfikacji tekstów w analizie sentymentu, która rozumiana jest jako badanie wydźwięku tekstu. Brak określonej struktury dokumentów tekstowych jest przeszkodą w realizacji tego zadania. Taki stan rzeczy wymusił rozwój wielu różnorodnych technik ustalania sentymentu dokumentów. W artykule przeprowadzono analizę porównawczą dwóch metod badania sentymentu: naiwnego klasyfikatora Bayesa oraz regresji logistycznej. Badane teksty są napisane w języku polskim, pochodzą z banków i mają charakter marketingowy. Klasyfikację przeprowadzono, stosując podejście *bag-of-n-grams*. W ramach tego podejścia dokument tekstowy wyrażony jest za pomocą podciągów składających się z określonej liczby *n* wyrazów. Uzyskane wyniki pokazały, że lepiej spisała się regresja logistyczna.

Słowa kluczowe: analiza sentymentu, klasyfikacja dokumentów, textmining, regresja logistyczna, naiwny klasyfikator Bayesa

JEL: C81, M31

Received: 2021-01-17; verified: 2021-03-30. Accepted: 2021-06-30