Dealing with Critical Issues in Emails: A Comparison of Approaches for Sentiment Analysis

Bernd Markscheffel(✉) and Markus Haberzettl

Department of Information and Knowledge Management, Technische Universität Ilmenau, Ilmenau, Germany
{bernd.markscheffel,markus.haberzettl}@tu-ilmenau.de

Abstract. The customer service of larger companies is constantly faced with the challenge of mastering the daily flood of incoming emails. In particular, the effort involved in dealing with critical issues, such as complaints, and the insufficient resources available to deal with them can have a negative impact on customer relations and thus on the public perception of companies. It is therefore necessary to assess and prioritise these concerns automatically, if possible. It is therefore necessary to evaluate and prioritise these concerns automatically if possible. The sentiment analysis as the automatic recognition of the sentiment in texts enables such prioritisation. The sentiment analysis of German-language e-mails is still an open research problem and till now there is no evidence of a dominant approach in this field. The aim of this article is to compare three approaches for the sentiment analysis of German emails:

The first approach (A) is based on the combination of sentiment lexicons and machine learning methods. The second (B) is the extension of approach A by further feature extraction methods and the third approach (C) is a deep learning approach based on the combination of Word Embeddings and Convolutional Neural Networks (CNN). A gold standard corpus is generated to compare these approaches. Based on this corpus, systematic experiments are carried out in which the different method combinations for the approaches are examined.

The results of the experiments show that the Deep Learning approach is more effective than classical approaches and generates better classification results.

Keywords: Sentiment analysis · Machine learning · Feature extraction methods · Deep learning · KNIME

1 Introduction

1.1 Problem Description

Only 127 years have elapsed between Samuel Morse's milestone in the history of communication systems when he sent the Bible quote "What hath God wrought?" and Ray Tomlinson's keyboard line "QWERTYUIOP" as content of the first email. Meanwhile emails

S. Hammoudi et al. (Eds.): DATA 2019, CCIS 1255, pp. 23–36, 2020.
https://doi.org/10.1007/978-3-030-54595-6_2

are not to be dismissed from the daily life any longer, even more, enterprises see in emails the preferential communication channel in particular for customer service [1, 2]. The increasing amount of emails arriving daily at customer service poses a challenge for the prompt processing of customer concerns in companies [2]. Automated prioritization is necessary in order to identify and prioritize critical concerns to avoid the risk of negative effects on the perception of companies. One form of prioritization is the sentiment, the emotionally annotated mood and opinion in an email [3]. A sentiment is also an approach to solving further problems such as the analysis of the course of customer contacts, email marketing or the identification of critical topics [4]. Linguistic data processing (LDV) approaches are used to automatically capture sentiment [5].

Although the number of published research papers is increasing, sentiment analysis continues to be an open research problem [6, 7]. In especial view, there is a lack of in approaches specifically for the German language, whereby the automated classification of polarity in the categories positive, negative and neutral is of particular interest [8–10]. In research, methods of machine learning have prevailed over knowledge- and dictionary-based methods to determine polarity [8]. The reason for this is that machine learning methods approach human accuracy and are not restricted by the other two approaches (e.g. lack of dynamics in relation to informal language) [11, 12]. Knowledge- and dictionary-based methods define the rules manually. In contrast to that, machine learning represents the fully automated inductive detection of such rules using algorithms developed for this purpose [12]. So far, no machine learning method or procedures and approaches based on it have been identified as dominant - another reason why sentiment analysis is today still an open research problem [3, 5, 13].

1.2 Research Questions

There are several solutions for this problem. One approach for the classification of polarity is seen in the combination of sentiment dictionaries and machine learning methods [14] – experiment A. Further potential is considered in the combination of such lexicons and learning methods with other methods of feature extraction – experiment B. This paper is a significant extension of our previous conference paper [27] where we have discussed only these two approaches as a hybrid of classical methods. Stojanowski et al. [15] see the Deep Learning approach in the context of sentiment analysis through the automation of feature extraction as more robust and flexible than the classical procedures mentioned above, especially when used in different domains (language, text structure) – our experiment C.

The main aim of this paper is to compare these approaches for German-language emails at the document level. We will answer the questions: do machine learning methods based on sentiment lexicons generate better results in the context of sentiment analysis if the lexicon is combined with other methods of feature extraction and how does a Deep Learning solution based on the combination of Word Embeddings and CNN compare to the results of experiments A and B. The paper is structured as follows. Section 2 describes the methodology of our research, Sect. 3 presents and compares the results of our several experiments before we summarize and give an outlook on future work in Sect. 4.

2 Methodology

2.1 Literature Analysis and Related Work

The several machine learning and feature extraction methods to be identified for the different approaches are determined by a systematic literature analysis according to Webster and Watson [16]) and is additionally supplemented by Prabowo and Thelwall [17] when structuring the findings. The complete methodology and the results of the literature analysis, the determined machine learning methods, the identified relevant feature extraction methods and a comprehensive presentation of related work are described by Haberzettl and Markscheffel [18].

2.2 Implementation

We have implemented these to be compared approaches with the Konstanz Information Miner (KNIME) in version 3.5.2.25. The data required for implementation are acquired according to the Gold Standard requirements of Wissler et al. (2014). The results of the approaches will then be compared using identified quality criteria, which have been recognized in the context.

2.3 Data Acquisition

Text data are unstructured data. For the real classification, process it is necessary convert it into structured data. This data is collected in a corpus and split into a training - and a test data set for the analysis process. As no suitable, freely accessible corpus is available for this task, a separate corpus must be created and coded that meets the requirements of the Gold Standard.

For this purpose, 7,000 requests from private customers to the customer service of a company in the telecommunication sector are used. Since a full survey is not possible due to the manual coding effort and no information on the distribution of polarity in the population is available, this sample was determined based on a simple random selection. Coding by only one expert should be rejected, especially in view of the Gold Standard requirements. The argumentation for a higher data consistency due to this is to be critically considered especially in light of the subjectivity of the sentiment, because sentiment is interpreted differently by different persons, for example, due to different life experiences [4, 19, 20]. This characteristic has to be reflected in the corpus. The following parameters, therefore, apply to the coding: Emails should be evaluated from the writer's point of view and categorized exclusively as an entire document. In addition, only subjective statements are relevant for determining positive or negative sentiments. The coding was therefore carried out in three steps:

1. The sample was divided into seven equally sized data sets. These sets were coded by six different experts who had previously received a codebook with instructions (the assignment of the groups was random in each phase; no reviewer coded a document twice). In addition to the general conditions, the codebook contains the class scale to be used and instructions for the classification of the classes (1 – very positive …5 – very negative and 6 – Mixed, which contains positive and negative elements).

2. The sets were again coded by different experts due to the subjective interpretation of the sentiment. This expert had no information about the previous coding.
3. All emails were identified, which were coded differently in each of the previous steps. These emails were assigned to a new expert for the set, who performed a third encoding.

After coding, the corpus is divided into a training and test data set in a stratified manner with a ratio of 70:30. Then the emails are converted into documents.

2.4 Data Preprocessing

The emails are already pre-processed in the source system:

– Personal customer data (name, address, etc.) have been anonymized and replaced.
– HTML tags, meta data (sender, IDs, etc.), attachments have been deleted and
– Message histories in the emails are removed.

Nevertheless, there is a large number of non-text elements to be found and have to be eliminated. The pre-processing workflow consists of the following steps:

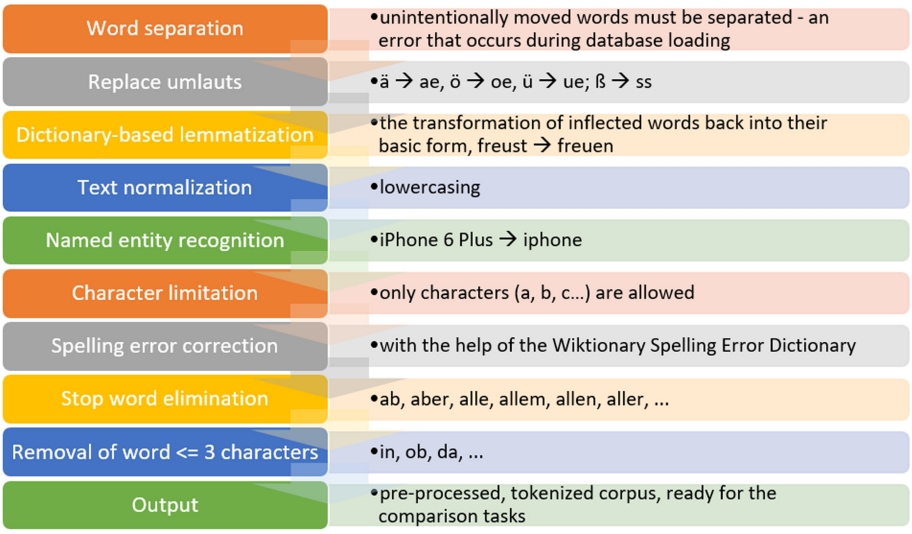

Fig. 1. Data preprocessing workflow

2.5 Feature Extraction and Selection

In a next step, we have to extract features from this corpus. Features are defined as numerically measurable attributes and properties of data. In the context of text mining,

feature extraction should be understood as the *structuring process of unstructured data.* The extraction is split into two parts: Features are generated on the one hand by direct conversion of texts or tokens and on the other hand by applying the feature extracting methods identified and introduced by Haberzettl, and Markscheffel [18]. Table 1 collects the several feature extraction methods used in our approach.

Table 1. Feature extraction methods [18]

n-Gramm	n-G
Term frequency - Inverse document frequency	TF-IDF
Term presence	TP
Term frequency	TF
Part of speech tagging	POS
Modification feature	MF
Negation	NEG
Pointwise Mutual Information (PMI)	PMI
Sentiment Dictionary (SM)	SM
Category (Cat)	CAT
Corpus specific	COR

2.6 Sentiment Lexicon

Sentiment dictionaries are dictionaries in which words are assigned to a polarity index. Sentiment dictionaries are context-sensitive, i.e. words and values contained in them apply primarily to the context in which they were created. Since no suitable dictionary exists for the context of German-language emails, such a dictionary had to be created. For resource reasons, an automated, corpus-based approach was pursued.

According to SentiWS [21] a generation on co-occurrence based rules is chosen. Pointwise Mutual Information (PMI) is used as a method for the analysis of co-occurrence and thus for the determination of semantic orientation [21–23]. In our specific case, two million uncoded emails were acquired from the same database as the corpus. Random sampling made the selection. All emails were pre-processed according to the process described in Fig. 1. For all words contained in these emails the semantic orientation {positive, negative} was determined on the basis of the PMI [21, 22], i.e. for each word its similarity to previously defined positive or negative seed words is calculated. For each of the 93,170 words identified, a threshold value for clipping the lexicon SO-PMI $\in [-0{,}13; 0{,}08]$ was determined by manual checking, taking into account the Zipf-distribution, so that the final lexicon consists of 1,704 negative and 955 positive words. Table 2 shows a cut-out of the sentiment dictionary with its top ten positive and negative normalized PMI-values, whereby the normalization is within the boundaries $-1 < \text{PMI} < 1$.

Table 2. Cut-out of the sentiment dictionary SentiMail (SM) [27].

Positive term	Scaled PMI	Negative term	Scaled PMI
herzlich	1	betruegen	−1
empathisch	0,9786	verarschen	−0,983
beglueckwuenschen	0,9589	andrehen	−0,9798
angenehm	0,954	dermassen	−0,9743
bedanken	0,9259	vertrauensbruch	−0,9628
kompliment	0,9156	scheiss	−0,9336
danke	0,9148	anluegen	−0,9263
sympathischen	0,9134	abzocke	−0,9233
sympathisch	0,8956	taeuschung	−0,9181
nervositaet	0,878	geschaeftsgebaren	−0,9137

3 Experiments and Results

For experiment A and B various machine learning methods were identified and introduced by Haberzettl, and Markscheffel [18, 27]. In Table 3 we have collected the several identified machine-learning methods used in approach A and B.

Table 3. Machine-learning methods used in experiment A and B [18, 27].

Support Vector Machine	SVM
Artificial Neural Network	ANN
Naive Bayes	NB
Logistic Regression or Maximum Entropy	LR or ME
k-nN nearest neighbor	k-nN

The implementation of the machine-learning methods in combination with the above introduced feature extracting methods was done with different libraries of Weka integration of KNIME (e.g. LibSVM, NaiveBayesMultinominal) or it could directly implemented as nodes (LR Logistics (3, 7), k-nN). The ANN was realized by a multi-layer perceptron starting from our multi-class case. A layer and M/2 (M = feature) neurons in this layer were chosen as a starting point and then successively increased to M + 2 neurons.

3.1 Evaluation

The results of the experiments and thus the classification itself are to be evaluated with the use of quality criteria. With the help of a confusion matrix, the results of the classification

can be divided according to positive and negative cases. The four resulting cases from the classification in the confusion matrix (true positive, true negative, false positive, false negative) allow the derivation of the following different quality criteria: Accuracy (ACC), Precision (PRE), Recall (REC) and F-Measure (F1) [24, 25]. The validity of the quality criteria is ensured by a 10-fold stratified cross-validation [26]. Accuracy is used as the decisive criterion for determining the best result due to the limitations discussed by Haberzettl, and Markscheffel in [18, 27].

3.2 Experiments and Results for the Sentiment Dictionary (A) and Feature Extraction (B)

In a first step, based on the approaches A and B, the sentiment lexicon to be used was first determined. For this purpose, all learning methods were trained on the features of the two used lexicons (SentiWS [21] and SentiMail-SM, see Sect. 2.6) and the combination of both. The result is the result of experiment A. Figure 2 shows the corresponding workflow implemented with KNIME for experiments A and B [27].

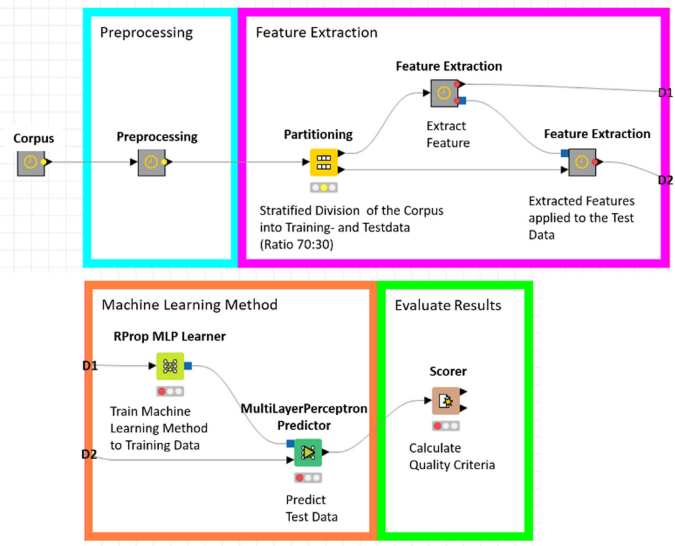

Fig. 2. KNIME workflow for the experiments A and B [27].

The results of the first experiment are obvious (see Table 4): For each learning method, the combination of both sentiment lexicons is the best alternative with regard to each quality criterion. Only the precision at NB is better with SentiWS - probably, measured by the recall, due to the simple assignment of the emails to the most frequented class (neutral). Particularly, with regard to the exactness (Precision, Recall, F1-Measure), the combination of both lexicon is dominant [27]. Table 4 shows a compilation of the results.

So, out of the results of experiment A both sentiment lexicon were selected from the results of A. It should be noted that the SentiMail (SM) lexicon, created within the context, produces better results in direct comparison with SentiWS (SW) - this substantiates the need for context-dependent sentiment dictionaries. The rank assigned according to Accuracy indicates that the best result for experiment A is the combination of ANN and both sentiment dictionaries. This result is also confirmed by the remaining quality criteria (F1 is to be weighted higher than the Precision outlier is) [27].

Table 4. Comparison of the sentiment lexicons SentiMail (SM) and SentiWS (SW) as feature extraction method and the best result (R), evaluated according to accuracy for approach A [27].

	R	ACC	PRE	REC	F1	
SVM	2	83,19%	83,26%	71,43%	75,87%	SMSW
	5	80,41%	80,18%	63,76%	69,14%	SM
	9	78,44%	74,70%	62,25%	65,86%	SM
ANN	1	83,82%	82,26%	74,55%	77,78%	SMSW
	4	81,44%	79,68%	68,20%	72,49%	SM
	7	79,17%	73,64%	65,98%	68,83%	SM
NB	12	75,67%	68,47%	67,81%	67,16%	SMSW
	14	74,47%	67,45%	63,95%	63,92%	SM
	15	72,89%	71,59%	47,41%	49,75%	SM
ME	3	82,73%	82,22%	70,97%	75,21%	SMSW
	6	80,33%	79,35%	64,22%	69,32%	SM
	10	78,32%	74,30%	61,74%	65,72%	SM
KnN	8	79,14%	75,01%	69,23%	71,65%	SMSW
	11	77,58%	71,81%	65,75%	68,22%	SM
	13	75,09%	67,88%	61,70%	64,08%	SM

For the second experiment (B), the best lexicon for each learning method is used. In the following step, we had to determine which frequency is to be used for the unigrams. The background for this is the often cited comparison between term presence (TP) and relative term frequency (relTF), at which the term presence dominates [28]. For this purpose, each machine learning method was trained with all three-frequency types (TP, relTF, TF-IDF) in each case as well as the identified sentiment lexicons from the previous experiment step. Only the frequency, which achieves the best results according to Accuracy, was selected for each learning method. The results of the remaining 62 possible combinations of the feature categories for each learning method are evaluated, whereby each of these combinations must inevitably contain the sentiment dictionary and produces the results for experiment B, (see Table 5) [27].

Table 5. Comparison of term presence (TP) vs. TF-IDF vs. relative term frequency (relTF) as additional features for approach A = experiment B [27]

	R	ACC	PRE	REC	F1	
SVM	1	84,67%	80,93%	76,65%	78,59%	TP
	2	84,16%	84,38%	73,48%	77,73%	TF-IDF
	3	83,73%	83,99%	72,55%	76,93%	Rel TF
ANN	7	77,02%	67,17%	67,01%	67,06%	TP
	8	76,92%	67,40%	65,98%	66,65%	TF-IDF
	9	75,72%	65,48%	66,39%	65,91%	Rel TF
NB	4	81,32%	74,08%	78,26%	75,96%	TP
	5	78,83%	71,86%	72,95%	72,23%	TF-IDF
	6	77,87%	71,15%	70,51%	70,56%	Rel TF
ME	10	72,83%	61,79%	67,14%	63,75%	TP
	12	71,33%	59,98%	64,51%	61,77%	TF-IDF
	13	71,30%	60,04%	64,86%	61,91%	Rel TF
KnN	11	72,43%	70,51%	52,80%	51,34%	TP
	14	69,29%	58,60%	59,01%	54,93%	Rel TF
	15	68,28%	56,88%	60,59%	55,05%	TF-IDF

Table 5 illustrates that the values for term presence (TP) are better than the values for TF-IDF as well as to the relative term frequency (relTF). So, only term presence for unigrams was used for all machine learning methods. The accuracy of the previously best learning method (ANN) decreases by 6.8% points, while, for example, the accuracy of the SVM (F1-Measure) increases further. This mainly reflects the core characteristics of the SVM, which benefits significantly more from large feature vectors than other learning methods. Also noteworthy is the small difference between TF-IDF and relTF. Although four of the five learning methods achieved a higher accuracy with TF-IDF than with the relative term frequency, the results of the quality criteria between the two frequencies usually deviate only marginally. As Table 6 shows, the results of SVM as well as of NB and ME with approach B are significantly better with regard to Accuracy and F1-Measure than in approach A. In particular, the 6.6% points higher accuracy and the 9.78% points higher F1 measurement at NB should be highlighted. ANN and k-nN show no significant deviations from A, whereby the ANN generates marginally worse results with respect to almost all quality criteria than in approach A [27].

Table 6. Best results for experiment B (R measured by accuracy), i.e. for features in combination with SentiWS and SentiMail [27].

	R	ACC	PRE	REC	F1	
SVM	1	85,03%	81,22%	77,98%	79,49%	POS, Neg, n-G
ANN	2	83,64%	81,84%	74,79%	77,83%	TF
NB	4	82,27%	75,62%	78,44%	76,94%	POS, Booster, Neg, n-G
ME	3	83,28%	81,43%	72,52%	76,14%	TF, POS, Cat
KnN	5	79,95%	77,22%	68,26%	71,77%	TF

3.3 Experiments and Results for the Deep Learning Approach as a Combination of Word Embeddings and CNN (C)

As described above, Stojanovski et al. [15] see the advantage of deep learning in the context of sentiment analysis in automated feature extraction. Here they refer to the connection of Word Embeddings and Convolutional Neural Networks (CNN). This approach is our role model for Experiment C.

Word embeddings represent words or tokens as vectors which, together with the inherent syntactic and semantic information, make it possible to assign these words to certain contexts [28–30]. To generate this information or to determine the vectors, there are different procedures. One of the most popular is the Word2Vec model according to Mikolov et al. [30]. Stojanovski et al. [15] also use Word2Vec, which is why we have also chosen it for our Deep Learning approach C. Word2Vec contains two methods for calculating Word embeddings: Continuous Bag of Words (CBOW) and Skip-gram [31]. The generated word vectors serve as input for the CNN. Additionally, due to the context of document classification, the approach presented by Le and Mikolov [32] is used to apply Word2Vec to Doc2Vec documents. Doc2Vec creates an n-dimensional document representation instead of the previous word representation. The vector calculation principles analogous to Word2Vec are Distributed Memory (DM) and Distributed Bag of Words (DBOW) [31] in the Doc2Vec context. In our approach, the Deeplearning4java (DL4J) integration in KNIME is used for the calculation of Word embeddings and Doc2Vec vectors. CBOW and Skip Gram are applied to the 2 million data sets already used for the creation of the sentiment lexicon. While the word relations were generated unsupervised, DBOW and DM allow the monitored generation on the training data. In this case, the tag of the documents has to be replaced by the respective class of the document. Thus, Doc2Vec makes it possible to calculate the relationships of words in the context of the sentiment of the document.

For the comparison of the results, the contextually unrelated Word Embeddings pre-trained by Reimers et al. [33] are used. The training parameters correspond to the findings of Mikolov et al. [31], where the word or Doc2Vec vector length is $n = 300$. The architecture of CNN is based on Stojanovski et al. in the given context. The implementation of the CNN takes place by means of the Keras implementation in KNIME.

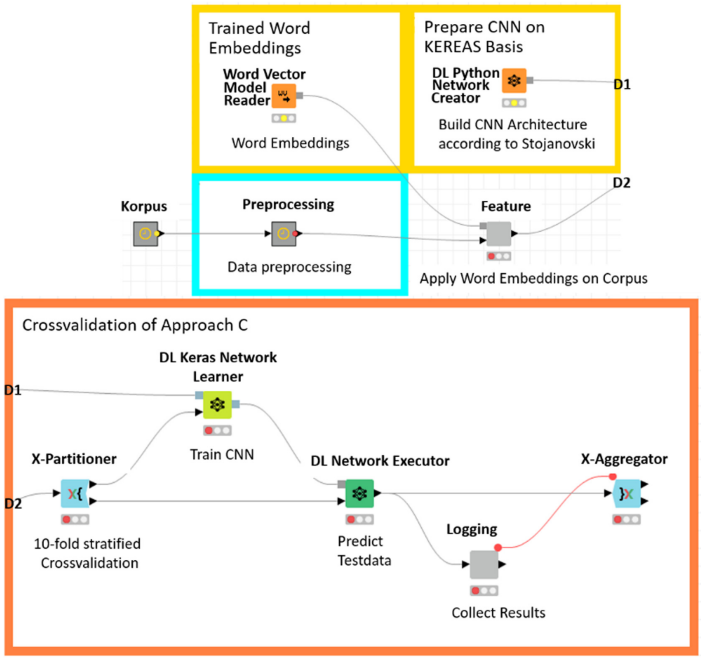

Fig. 3. KNIME workflow for the experiments C.

The experiments are performed by means of stratified 10-fold cross-validation. Figure 3 shows the architecture of Experiment C.

The results shown in Table 7, illustrate that Word Embeddings based on CBOW (with the exception of Precision) generate the best results. However, the difference to skip grams is marginal. This finding is also interesting about the fact that Mikolov et al. reported better results with Skip Grams. Furthermore, the assumption is confirmed that the embeddings created in the context are better than those created outside the context are.

Table 7. Results of the deep learning approach with the used word embeddings.

Embeddings	Accuracy	Precision	Recall	F1
CBOW	86,62%	84,11%	79,29%	81,46%
Skip Gram	86,05%	84,71%	77,77%	80,78%
DM	84,00%	79,54%	77,16%	78,18%
DBOW	83,73%	79,04%	76,77%	77,74%
Reimers	82,05%	79,43%	70,88%	74,14%

The Doc2Vec results should also be highlighted: With 4,658 emails, the training data was considerably lower compared to the two million emails at Word2Vec. Nevertheless, the results are already comparable with the results of approach A. Moreover, Le and Mikolov [32] expect DM to generate better results than DBOW.

4 Summary and Future Work

On the background of optimizing the analysis of the polarity of German-language emails at the document level, three approaches to sentiment analysis were compared in experiments: Approach A combines machine learning methods and sentiment dictionaries. Approach B extends this with additional feature extraction methods. Experiment C combines Word Embeddings and CNN. Measured against the quality criteria of the best results per approach, approach C dominates all cases (see Table 8).

Table 8. Comparison of the best results of the approaches A, B and C.

	Accuracy	Precision	Recall	F1
A	83,82%	82,26%	74,55%	77,78%
B	85,03%	81,22%	77,98%	79,49%
C	86,62%	84,11%	79,29%	81,46%

Approach C classifies polarity best, while approach B generates better results than approach A. It should be noted that this confirmation only applies to the best results. Within the experiments a dependence of the results on the respective method combination is visible. Thus, the results of a machine learning method based on sentiment lexicons are not necessarily better by adding further feature extraction methods. Furthermore, the deep learning approach, depending on the word embedding that is used, is not always better than an alternative from A or B.

However, the insights gained are still remarkably relevant for the customer service of companies. In particular, the Deep Learning approach is suitable for automatically identifying negative sentiment in customer e-mails in order to treat the respective concerns preferentially and to identify negative effects for the company at an early stage and, at best, to avoid them.

For further research, the investigation of the combination of machine learning methods itself offers further potential. A significant improvement of the approaches can be assumed in the inclusion of further linguistic specifics such as the recognition of sarcasm and irony.

References

1. Gupta, N., Gilbert, M., Di Fabbrizio, G.: Emotion detection in email customer care. In: Inkpen, D., Strapparava, C. (eds.) Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, pp. 10–16. Association for Computational Linguistics, Los Angeles (2010)

2. Radicati Group: Email statistics report, 2018–2022 - executive summary, pp. 1–4. https://doi.org/10.18356/1de36bac-en
3. Borele, P., Borikar, D.: An approach to sentiment analysis using artificial neural network with comparative analysis of different techniques. IOSR J. Comput. Eng. **2**(18), 64–69 (2016)
4. Nasukawa, T., Yi, J.: Sentiment analysis: capturing favorability using natural language processing. In: Proceedings of the 2nd International Conference on Knowledge Capture, 23–26 October, 2003, Florida, USA, pp. 70–77. ACM Press, New York (2003). https://doi.org/10.1145/945645.945658
5. Agarwal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, R.: Sentiment analysis of twitter data. In: Proceedings of the Workshop on Language in Social Media LSM 2011, Portland 2011, pp. 30–38, Stroudsburg (2011)
6. Bravo-Marquez, F., Mendoza, M., Poblete, B.: Meta-level sentiment models for big social data analysis. Knowl. Based Syst. **69**, 86–99 (2014)
7. Ravi, K., Ravi, V.: A survey on opinion mining and sentiment analysis: tasks, approaches and applications. Knowl. Based Syst. **89**, 14–46 (2015). https://doi.org/10.1016/j.knosys.2015.06.015
8. Scholz, T., Conrad, S., Hillekamps, L.: Opinion mining on a german corpus of a media response analysis. In: Sojka, P., Horák, A., Kopeček, I., Pala, K. (eds.) TSD 2012. LNCS (LNAI), vol. 7499, pp. 39–46. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32790-2_4
9. Steinbauer, F., Kröll, M.: Sentiment analysis for German Facebook pages. In: Métais, E., Meziane, F., Saraee, M., Sugumaran, V., Vadera, S. (eds.) NLDB 2016. LNCS, vol. 9612, pp. 427–432. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-41754-7_44
10. Waltinger, U.: GermanPolarityClues: a lexical resource for German sentiment analysis. In: Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., et al. (eds.) Proceedings of the 7th Conference on International Language Resources and Evaluation (LREC-10), pp. 1638–1642. European Language Resources Association, Valletta (2010)
11. Cao, Y., Xu, R., Chen, T.: Combining convolutional neural network and support vector machine for sentiment classification. In: Zhang, X., Sun, M., Wang, Z., Huang, X. (eds.) CNCSMP 2015. CCIS, vol. 568, pp. 144–155. Springer, Singapore (2015). https://doi.org/10.1007/978-981-10-0080-5_13
12. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. (CSUR) **34**(1), 1–47 (2002). https://doi.org/10.1145/505282.505283
13. Vinodhini, G., Chandrasekaran, R.M.: Sentiment analysis and opinion mining: a survey. Int. J. Adv. Res. Comput. Sci. Softw. Eng. **2**(6), 282–292 (2012)
14. Ohana, B., Tierney, B.: Sentiment classification of reviews using SentiWordNet. In: 9th IT & T Conference, October 2009, pp. 1–9, Dublin (2009) https://doi.org/10.21427/d77s56
15. Stojanovski, D., Strezoski, G., Madjarov, G., Dimitrovski, I.: Twitter sentiment analysis using deep convolutional neural network. In: Onieva, E., Santos, I., Osaba, E., Quintián, H., Corchado, E. (eds.) HAIS 2015. LNCS (LNAI), vol. 9121, pp. 726–737. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19644-2_60
16. Webster, J., Watson, R.T.: Analyzing the past to prepare for the future: writing a literature review. MIS Q. **26**(2), 13–23 (2002)
17. Prabowo, R., Thelwall, M.: Sentiment analysis: a combined approach. J. Informetr. **3**, 143–157 (2009). https://doi.org/10.1016/j.joi.2009.01.003
18. Haberzettl, M., Markscheffel, B.: A literature analysis for the identification of machine learning and feature extraction methods for sentiment analysis. In: Proceedings of the 13th International Conference on Digital Information Management (ICDIM 2018), pp. 385–391, Berlin (2018). https://doi.org/10.5220/0008114803850391

19. Bütow, F., Schultze, F., Strauch, L.: Semantic search: sentiment analysis with machine learning algorithms on German news articles (2017). http://www.dai-labor.de/fileadmin/Files/Publikatio-nen/Buchdatei/BuetowEtAl–SentimentAnalysis. Accessed 11 May 2019
20. Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., Kappas, A.: Sentiment strength detection in short informal text. J. Am. Soc. Inf. Sci. Technol. **61**(12), 2544–2558 (2010). https://doi.org/10.1002/asi.21416
21. Remus, R., Quasthoff, U., Heyer, G.: SentiWS - a publicly available German-language resource for sentiment analysis. In: Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., et al. (eds.) International Conference on Language Resources and Evaluation, pp. 1168–1171 (2010)
22. Turney, P.D.: Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: Isabelle, P. (eds.) Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Philadelphia 2002, pp. 417–424 (2002)
23. Turney, P.D., Littman, M.L.: Measuring praise and criticism. ACM Trans. Inf. Syst. **21**(4), 315–346 (2003). https://doi.org/10.1145/944012.944013
24. Cleve, J., Lämmel, U.: Data Mining. De Gruyter, Oldenbourg (2014)
25. Davis, J., Goadrich, M.: The relationship between Precision-Recall and ROC curves. In: Cohen, W., Moore, A. (eds.) Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, PA, pp. 233–240 (2006). https://doi.org/10.1145/1143844.1143874
26. Kohavi, R.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Mellish, C.S. (eds.) Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI-1995), Montreal, pp. 1137–1143, Morgan Kaufmann Publishers Inc., San Francisco (1995)
27. Markscheffel, B., Haberzettl, M.: Sentiment analysis of German emails: a comparison of two approaches. In: DATA 2019 - Proceedings of the 8th International Conference on Data Science, Technology and Applications, pp. 385–392, SCITEPRESS – Science and Technology Publications (2019)
28. Pang, B., Lee, L.: Opinion mining and sentiment analysis. FNT Inf. Retr. **2**, 1–135 (2008). https://doi.org/10.1561/1500000011
29. Liu, Y., Liu, Z., Chua, T.S., Sun, M.: Topical word embeddings. In: Proceedings of the 29th AAAI Conference on Artificial Intelligence, pp. 2418–2424. AAAI Press, Palo Alto (2015)
30. Neelakantan, A., Shankar, J., Passos, A., McCallum, A.: Efficient non-parametric estimation of multiple embeddings per word in vector space. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, pp. 1059–1069. Association for Computational Linguistics (2014). https://doi.org/10.3115/v1/d14-1113
31. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR http://arxiv.org/abs/1301.3781 (2013)
32. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: Xing, E.P., Jebara, T. (eds.) Proceedings of the 31st International Conference on Machine Learning - Volume 32 (ICML 2014), vol. 32, pp. 1188–1196, JMLR.org (2014)
33. Reimers, N., Eckle-Kohler, J., Schnober, C., Kim, J., Gurevych, I.: GermEval-2014: nested named entity recognition with neural networks. In: Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition, Hildesheim, pp. 117–120 (2014)