Expert Systems With Applications 186 (2021) 115771

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Analysis of sentiment in tweets addressed to a single domain-specific Twitter account: Comparison of model performance and explainability of predictions

Krzysztof Fiok [a], Waldemar Karwowski [a], Edgar Gutierrez [a,b,*], Maciej Wilamowski [c]

[a] Department of Industrial Engineering and Management Systems, University of Central Florida, Orlando, FL 32816, USA
[b] Center for Latin-American Logistics Innovation, LOGyCA, Bogota 110111, Colombia
[c] University of Warsaw, Faculty of Economic Sciences, Warsaw, Poland

ARTICLE INFO

Keywords:
Natural language processing
Deep learning
Sentiment analysis
Machine learning
Explainability
Twitter

ABSTRACT

Many institutions and companies find it valuable to know how people feel about their ventures; hence, scientific research in sentiment analysis has been intensely developed over time. Automated sentiment analysis can be considered as a machine learning (ML) prediction task, with classes representing human affective states. Due to the rapid development of ML and deep learning (DL), improvements in automatic sentiment analysis performance are achieved almost every year. Since 2013, Semantic Evaluation (SemEval) has hosted a worldwide community-acknowledged competition that allows for comparisons of recent innovations. The sentiment analysis tasks focus on assessing sentiment in Twitter posts authored by various publishers and addressing multiple subjects. Our study aimed to compare selected popular and recent natural language processing methods using a new data set of Twitter posts sent to a single Twitter account. For improved comparability of our experiments with SemEval, we adopted their metrics and also deployed our models on data published for SemEval-2017. In addition, we investigated if an unsupervised ML technique applied for the detection of topics in tweets can be leveraged to improve the predictive performance of a selected transformer model. We also demonstrated how a recent explainable artificial intelligence technique can be used in Twitter sentiment analysis to gain a deeper understanding of the models' predictions. Our results show that the most recent DL language modeling approach provides the highest quality; however, this quality comes at reduced model transparency.

1. Introduction

The concept of analyzing sentiment in Twitter data is almost as old as Twitter itself, hence natural language processing (NLP) techniques used in this regard changed over time. Researchers improved machine learning (ML) classifiers, and practitioners experimented with a plethora of user and tweet-related features. Achievements of deep learning (DL), including models from the transformer family, were also leveraged, enabling every developer of a particular expert system with an abundance of viable up to date model solutions. However, there is always room for improvement. To provide a proper context for our contributions to the sentiment analysis of Twitter, we begin with a brief review of previous achievements in the field.

1.1. Historical view of sentiment analysis in Twitter

In 2009, Go et al. (2009) demonstrated the usefulness of the n-gram language model (LM) in combination with the naïve Bayes (NB) classifier for dividing tweets into positive, neutral, and negative classes. Shortly thereafter, (Pak and Paroubek, 2010) demonstrated a very similar research approach. In 2011, Agarwal et al. (2011) proposed the addition of parts-of-speech tags to the feature set and introduced "a tree kernel to obviate the need for tedious feature engineering." Based on these improvements, the authors reported new state-of-the-art performance. Kouloumpis et al. (2011) demonstrated that the use of lexicon and "micro-blogging" features is beneficial in three-class sentiment analysis in Twitter. In 2012, Wang et al. (2012) defined a four-class sentiment analysis task in Twitter, adding an "unsure" class to the previous three classes. That work addressed a specific topic, i.e.,

* Corresponding author.
E-mail addresses: fiok@ucf.edu (K. Fiok), wkar@ucf.edu (W. Karwowski), edfranco@mit.edu (E. Gutierrez), mwilamowski@wne.uw.edu.pl (M. Wilamowski).

https://doi.org/10.1016/j.eswa.2021.115771
Received 8 July 2020; Received in revised form 12 August 2021; Accepted 12 August 2021
Available online 21 August 2021

presidential elections in the USA, and utilized a simple uni-gram LM and an NB classifier. Saif et al. (2012) developed a useful approach for adding new features, namely, by searching key phrases and creating tweet labels; for example, if "iPhone" was found in the tweet text, the data instance was tagged as "Apple product." The authors demonstrated this approach for a task with two classes, i.e., "positive" and "negative." Again, the NB classifier was used as a machine learning (ML) classifier. Ghiassi et al. (2013) addressed a brand-related sentiment analysis task in Twitter and proposed five classes of sentiment: −2: strongly negative, −1: mildly negative, 0: neutral, 1: mildly positive, 2: strongly positive. That study made use of a feature set with an optimized size and different ML classifiers, i.e., a dynamic artificial neural network and support vector machines (SVMs). Severyn and Moschitti (2015) addressed a SemEval-2015 twitter sentiment analysis task by using a deep convolutional neural network (CNN). Their CNN was first trained in an unsupervised manner to create its own token representations and was later fine-tuned on a relatively small corpus of tweets. A survey of sentiment analysis methods in Twitter reported by Giachanou and Crestani (2016) identified various lexicon-based, ML, and graph-based methods. Most algorithms were based on human-designed features from WordNet (Miller, 1998), n-grams, lexicon features, term frequency (TF), and tweet-specific terms such as hashtags. This survey also identified a typical preprocessing scenario applied before feature extraction, namely removal of stop words, uniform resource locators (URLs), @ and # signs removal, and spell correction (i.e., elongated words). Some studies, such as that by Singh and Kumari (2016), have focused explicitly on the importance of text preprocessing in the analysis of Twitter data. Pagolu et al. (2016) compared two feature extraction techniques, namely Word2ved (Mikolov et al., 2013) and n-gram, both combined with a random forest (RF) classifier for the goal of predicting stock market movements based on three-class Twitter sentiment analysis. A systematic review of sentiment analysis in Twitter was presented by Kumar and Jaiswal (2020), who found that more recent works tend to more frequently use DL methods. An example of this trend is the study by Alharbi and de Doncker (2019), which applied a CNN to data from SemEval-2016. The team that won task 4 sub-tasks C and E (Cliche, 2017) in the SemEval-2017 five-class sentiment analysis competition utilized an ensemble of DL models. This team trained ten CNNs and ten long-short term memory (LSTM) networks using features provided by Word2vec, Glove (Pennington et al., 2014), and FastText (Bojanowski et al., 2017). The experiments demonstrated the superior performance of Word2vec and FastText, and the inferences used for final submission were obtained through soft voting. The authors also reported task-specific tricks that improve performance for sub-tasks using tweets that have a topic label in addition to a sentiment label. In the preprocessing phase, the authors added the topic at the end of a tweet if the tweet did not originally contain the topic word; subsequently, in the embedding phase, they extended each word embedding by another embedded space of dimension 5, which indicated whether the embedded word was part of the topic. More recent works, such as that by Potamias et al. (2019), have followed the trending use of a transformer model family in natural language processing (NLP). In this context, the concept of predicting stock markets was again explored via sentiment analysis (Sousa et al., 2019). In this case, the task was addressed with the bidirectional encoder representations from transformers (BERT) method (Devlin et al., 2018). The improved prediction quality provided by BERT was demonstrated against NB and SVM classifiers fed bag-of-words features and TF features and a CNN-based method called textCNN, as proposed by (Rosenthal et al., 2019). Transformer-based Twitter sentiment analysis has also been demonstrated in other languages, including Spanish (González et al., 2019) and Italian (Gambino & Pirrone, 2019).

This brief review of research in the field of sentiment analysis in Twitter indicates that researchers have proposed preprocessing solutions that have successfully addressed the specifics of Twitter language in previous works. Moreover, various authors have demonstrated their solutions on data sets from different Twitter domains. Regarding prediction quality, it can be concluded that the evolution of methods used for Twitter sentiment analysis mirrors the overall progress of ML and DL in NLP. Early works benefited from simpler ML classifiers fed hand-crafted, lexicon-based, or TF features, while later solutions introduced progress by creating vector representations of tweets using deep neural networks (DNNs), i.e., CNNs, and LSTMs fed token-level features from LMs such as Glove, Word2vec, or FastText. Finally, the recently introduced transformer models has also been applied in the analysis of Twitter posts, for example, regarding hate speech (Mishra & Mishra, 2019), fake news (Schwarz et al., 2020), and sentiment (Song et al., 2020; Ibrahim, 2019).

1.2. Explaining model predictions doesn't seem popular in Twitter sentiment analysis

At the same time, we note that none of the reviewed works follow the recent trend in artificial intelligence (AI) of providing interpretable, clear explanations along with model predictions that would allow for an improved understanding of the analyzed phenomenon. The so-called explainable AI (XAI) methods developed for explaining ML model predictions can also be used in the Twitter sentiment analysis domain. Adadi and Berrada (2018) and (Barredo Arrieta et al., 2019) provide a thorough review of various methods designed to allow improved understanding of ML and DL model predictions. From numerous possible approaches, local interpretable model-agnostic explanations (LIME), the variant called submodular pick LIME (SP-LIME) (Ribeiro et al., 2016), and Shapley additive explanations (SHAP) (Lundberg and Lee 2017) are among the more popular XAI techniques that can be used in the context of ML.

1.3. Aims and contribution of this study

Given the abundance of possible choices regarding the definition of models and features to be used in Twitter sentiment analysis, we decided to 1) compare selected NLP models, including recent transformer models, on a new data set (sentiment@USNavy) of Twitter posts addressed to a single Twitter account; and 2) present how the quality of these models compare to results achieved by teams that participated in the SemEval-2017 competition. We hope that carrying out this goal will facilitate the selection of model solutions for AI practitioners.

We also hypothesize that XAI techniques will be beneficial to Twitter sentiment analysis. As research is frequently focused on prediction quality and comparisons of various models, our study may be one of the first to explore the application of a recent XAI technique to automated sentiment analysis in Twitter. Therefore, our third aim was to show how recent XAI techniques can be used in Twitter sentiment analysis to better understand the models' predictions.

Another aspect of our study that might be valuable to AI practitioners is our specific approach to selecting data for the new data set. The rationale for our belief is based on three assumptions: 1) when sentiment analysis is carried out for posts addressed to various accounts, the applied models can benefit from structured information accompanying those accounts as the specifics of the targeted account could be informative for the trained models. In a single account scenario, the models cannot benefit from such account-related features, making the prediction task more difficult, 2) AI practitioners often have to deal with limits of available data and could be forced to use only enterprise-targeted comments, which are always less abundant. Training a high-quality model with limited data-availability is still challenging, and 3) specifics of topics and language covered in enterprise posts can be mirrored by the specific language of responding users. In the analyzed case, Tweets often contain language particular to the US Navy domain, for example, acronyms like "BZ" (Bravo Zulu) or "SAPR survey" (Sexual Assault Prevention and Response survey) because the authors of analyzed posts often exhibit some interest and history of relations with US Navy. Fitting and later applying a model to such specific data is likely

to provide higher-quality results than using a general sentiment model to highly domain-specific data.

In our main analysis we do not focus on text preprocessing, training parameter optimization, or other task-specific tricks known to improve the final results (Cliche, 2017). Instead, we followed a rather out-of-the-box approach using model parameters borrowed from other researchers, as we preferred to demonstrate baseline results that can definitely be improved.

However, to demonstrate that improving those results is possible, we show that optimizing the training process of recent transformer models is possible and beneficial, and we present a side experiment in which a single parameter value is optimized for a selected model. Also, because we found that various researchers (Ren et al., 2016; Xiang & Zhou, 2014; Cliche, 2017) demonstrated that utilizing topic-related information may increase the performance of sentiment prediction realized with selected NLP models, we carry out another side experiment devoted to investigating if topic-related information extracted through unsupervised ML methods can increase the performance of sentiment prediction based on a selected recent transformer model.

We believe that our work adds value to the field of Twitter sentiment analysis by introducing the following contributions:

1) Publication of the sentiment@USNavy data set for fine-grained classification of sentiment on Twitter data (Fiok, 2020),
2) Demonstration of the quality of selected LMs, including recent state-of-the-art transformer models, on the sentiment@USNavy data set and SemEval-2017 task 4 data set,
3) Presentation of the quality–explainability trade-off with selected LMs for sentiment classification in Twitter by means of state-of-the-art XAI techniques, and
4) Investigating the influence of topic-related information extracted through unsupervised ML methods on sentiment prediction based on a selected recent transformer model.

It is simple to compare our work with the renown SemEval (2017 edition, task 4 sub-task C and E) (Rosenthal et al., 2019) due to the adoption of the same metrics and the deployment of our models on a data set used in that competition. For easy reproduction of our experiments, we have published an applicable Python3 code (Fiok, 2020).

2. Methods

This section provides information regarding the analyzed data, metrics, models used for feature extraction and classification, cross-validation procedure, statistical analysis, and methods used to explain the model predictions.

2.1. Analyzed data

In this work, we analyzed posts by Twitter users directed to a single Twitter entity, namely, the official @USNavy account. Our tweet search covering the period from January 2011 to December 2019 was conducted on January 20, 2020 and resulted in a total of 130,688 tweets. The annual numbers of gathered tweets increased over time, as shown in Fig. 1.

2.1.1. Data set preparation: preprocessing and filtering tweets

This work is not focused on Twitter-specific data preprocessing tricks; therefore, we aimed to adopt a basic approach regarding text preparation and filtering. For all gathered tweets, we applied a preprocessing procedure, which began with the steps described in Fiok et al. (2020), i.e., we converted all images, retweets, and URLs to predefined tokens of "_IMAGE", "_RETWEET", and "_URL". Next, we removed all tweets that consisted only of those tokens, which resulted in the deletion of 26,523 tweets. We also removed 199 tweets that were posted by the official @USNavy account to itself. Although some users posted in

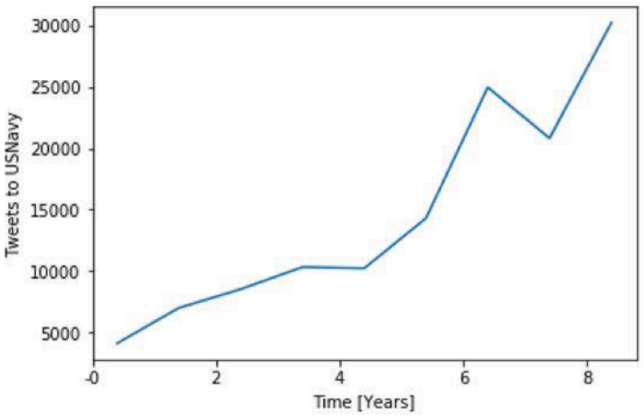

Fig. 1. Yearly number of tweets posted to the official @USNavy account.

foreign languages, our work was focused on English tweets; thus, we introduced an index describing the amount of signs not included in American standard code for information interchange (ASCII) defined as proportion of non-ASCII signs to the total sign number in a tweet (see formula 1). If a given tweet was rated with a non-ASCII index greater than 0.2, that tweet was removed. This procedure resulted in the deletion of 834 tweets. The defined procedure was unable to remove all foreign language tweets, including those in Spanish, Turkish, or Polish. For these languages, we defined an additional filtering function based on language-specific characters published at (Language recognition chart, 2019). With this solution, we removed an additional 1,905 non-English tweets. In the last filtering step, we removed 2,405 tweets that included only the "_IMAGE" token and less than four characters. Overall, the described filtering procedures resulted in the removal of 31,866 tweets.

$$\text{non-ASCII index} = \frac{1 - number\ of\ ASCII\ characters}{Total\ number\ of\ characters}$$

From the remaining 98,822 tweets, we randomly selected 5,000 posts for manual sentiment labeling. These tweets constitute the sentiment@USNavy data set.

2.1.2. The SemEval-2017 data set

To provide a straightforward comparison of the quality of models presented in this study, we utilized the data published for SemEval-2017 task 4 sub-tasks C and E (henceforth termed the SemEval-2017 data set). For this trial, we downloaded publicly available training and testing data splits from (SemEva2l-2017 Task 4, 2020). The SemEval-2017 data set was extensively described in (Rosenthal et al., 2019). For the purpose of this study, we revise its main features as follows: a) it comprises 6 000 training and 20 632 testing data instances, b) the data set was labeled according to a five-level scale of tweet sentiment, and c) the data instances were obtained from different sources and at different times.

It is important to mention that teams participating in the SemEval-2017 competition were allowed to utilize additional sources of data, including tweets published for previous SemEval competitions. However, in our study, we used only the data published specifically for SemEval-2017.

2.2. Labeling the sentiment data set

In our study, we adopted a five-level scale of tweet sentiment, similar to that proposed in SemEval-2017, with classes defined as: 0: very negative, 1: negative, 2: neutral, 3: positive, and 4: very positive.

The selected 5,000 tweets were labeled as follows:

1) Three researchers manually and independently labeled all posts.
2) We computed the Krippendorf alpha (Krippendorff, 2011) annotator agreement measure and obtained a low value of 0.592.

3) To obtain a unified sentiment value that could be used in sentiment prediction tasks, we applied the following algorithm:
   a) If a post received the same sentiment value from all annotators, the sentiment value was retained.
   b) If two out of three annotators agreed on the sentiment value and the sentiment value given by the third annotator differed only by 1, the majority sentiment value was retained.
   c) In all other cases, the tweets were manually labeled by a fourth annotator.

The above procedures resulted in 1,934 sentiment labels with full annotator agreement, 2,223 with majority agreement, and 843 tweet labels determined by a fourth annotator.

2.3. Metrics and comparability with SemEval-2017

To ensure that our experiments can be compared with previous research, we introduced metrics acknowledged in the renown SemEval competition. Because the sentiment@USNavy data set was labeled using a five-grade sentiment scale, the most similar SemEval tasks were sub-tasks 4C and 4E from the 2017 competition. For sub-task 4C, SemEval-2017 used the macro-averaged mean absolute error (MMAE) as the decisive metric. To provide more information, the classic non-class weighted mean absolute error (MAE) was also published. For the MMAE, we proposed our own implementation according to formula (2) (Rosenthal et al., 2019), and for the MAE, we used the implementation from scikit-learn (version 0.22.1) Python package (Sklearn, 2020). For sub-task 4E, SemEval-2017 relied on the Earth mover's distance (EMD, also known as the Wasserstein distance); to compute this parameter, we used the implementation from the scipy (version 1.4.1) Python package

Table 1
FE models used in our study.

Group	Feature Extraction	Model	Applicability	Short Description	Source
I	EFEs	TF	Training required	A technique for extracting features, which is popular due to its simplicity and speed, based on computing the frequency of token occurrences in text entities, e.g., tweets. According to Beel et al. (2013), 83% of recommender systems use a TF model.	Not available
		Linguistic inquiry and word count (LIWC)	Out-of-the-box	An acknowledged lexicon-based method dated back to 2001, created for automatic extraction of psychologically related information from text.	Pennebaker et al. (2001)
		Sentiment analysis and social cognition engine (SEANCE)	Out-of-the-box	A tool from 2017 that "contains a number of pre-developed word vectors developed to measure sentiment, cognition, and social order" and that extracts features from text based on various previously developed lexicon-based methods, i.e., the valence aware dictionary for sentiment reasoning (VADER) (Hutto & Gilbert, 2014).	Crossley et al. (2017)
II	DL without training on task-specific data (token embedding mean taken as tweet-level embeddings)	Robustly optimized BERT pretraining approach (RoBERTa)	Out-of-the-box	A pre-trained DL method that uses RoBERTa large LM to create token-level embeddings (embeddings from last four model heads are used). To achieve tweet-level vector representation, token embeddings are simply averaged. This model was implemented in Flair v. 0.4.5 (Akbik et al., 2019).	Liu et al. (2019)
		FastText	Out-of-the-box	A pre-trained DL method that uses token embeddings provided by a FastText LM trained previously on a Twitter corpus. To achieve tweet-level vector representation, token embeddings are simply averaged.	(Bojanowski et al., 2017)
		Universal sentence encoder (USE)	Out-of-the-box	A pre-trained DL method for creating vector representation at the sentence (tweet) level. Here, the "universal-sentence-encoder-large 5" version was used.	Cer et al. (2018)
III	DL with training on task-specific data (token embeddings converted into tweet-level embeddings by a trained LSTM)	Bidirectional LSTM with FastText	Training required	This FE uses a FastText model to create token-level embeddings, which are further used by a bidirectional two-layer LSTM with 512 hidden states to create tweet-level embeddings.	Bojanowski et al. 2016
		Bidirectional LSTM with RoBERTa	Training required	This FE uses a RoBERTa large LM to create token-level embeddings (embeddings from the last four model heads are used), which are further passed to a bidirectional two-layer LSTM with 512 hidden states to create tweet-level embeddings.	Liu et al. (2019)
	DL with fine tuning on task-specific data (tweet-level embeddings provided by built-in-transformer LM classification [CLS] output)	Fine-tuned RoBERTa large	Training required	This FE uses a fine-tuned RoBERTa large LM and its [CLS] output to obtain tweet-level embeddings.	Liu et al. (2019)
		Fine-tuned BERT large uncased	Training required	This FE uses a fine-tuned BERT large LM and its [CLS] output to obtain tweet-level embeddings.	Devlin et al. (2018)
		Fine-tuned BERT large cased	Training required	This FE uses a fine-tuned BERT large LM and its [CLS] output to obtain tweet-level embeddings.	Devlin et al. (2018)
		Fine-tuned XLNet large cased	Training required	This FE uses a fine-tuned generalized autoregressive pretraining for language understanding (XLNet) LM and its [CLS] output to obtain tweet-level embeddings.	(Yang et al., 2019)
		Fine-tuned BART large CNN	Training required	This FE uses a fine-tuned denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension (BART) large CNN LM and its [CLS] output to obtain tweet-level embeddings.	Lewis et al. (2019)
		Fine-tuned XLM-R	Training required	This FE uses a fine-tuned cross-lingual language model (XLM) RoBERTa large (XLM-R) and its [CLS] output to obtain tweet-level embeddings.	Conneau et al. (2019)
		Fine-tuned XLM MLM en 2048	Training required	This FE uses the fine-tuned XLM version "MLM en 2048" and its [CLS] output to obtain tweet-level embeddings.	Lample and Conneau (2019)

(Scipy, 2020). Because of its popularity, we also provide the F1 macro score, as implemented in scikit-learn.

$$MAE^M(h, Te) = \frac{1}{|C|}\sum_{j=1}^{|c|}\frac{1}{|Te_j|}\sum_{x_i \in Te_j}|h(x_i) - y_i|$$

The macro-average mean absolute error MAE^M is used as a classification measure: The sets (h, Te) represents the predicted and the original labels sentiment for test documents, respectively. The item x_i belongs to the set of the sentences Te_j, that is to say $(x_i \in Te_j)$, whose true sentiment class is C_j, y_i represents their correct or original label and $h(x_i)$ is its predicted sentiment label. Therefore, the absolute value of the "distance difference" $|h(x_i) - y_i|$ between classes represents how far are the predicted to the real class label. For example, the distance between the labels for the item x_i Highly negative (predicted) and Neutral (original) is 2. The summations are over all items that belongs to Te and over all class C_j. C denotes sentiment classes or unique labels in Te for "macroaveraging".

2.4. Models for feature extraction

In our study, we distinguish three types of feature extractor (FE) models: 1) explainable FEs (EFEs); 2) pre-trained DL FEs that do not require training on task-specific data; and 3) trainable DL FEs that require training on task-specific data. For a list of selected FEs, please refer to Table 1. The proposed division allows us to focus on comparing the usability of recent NLP techniques in conjunction with XAI tools, which sheds light on the rationale of model decisions. Also, distinction of the DL FE group that does not require any training allows us to assess the quality of out-of-the-box approaches. Additionally, by adopting a group of DL approaches trained on task-specific data, we can also observe the state-of-the-art quality in sentiment analysis.

2.5. ML models and computing machine

The FE models' output was forwarded to a gradient boosting (GB) ML classifier from the XGBoost (version 1.0.2) Python package (XGboost, 2020) with the parameters as presented in Table 2. The undisclosed parameters were set to default values proposed in the package.

All experiments were coded in Python (version 3.7) and were performed on the same computer, which was equipped with a single NVIDIA Titan RTX 24 GB RAM GPU. Most NLP- and DL-related computing, e.g., training of LMs with LSTMs and fine-tuning of transformer models, was performed with the use of the Flair (version 0.4.5) Python package (Akbik et al., 2019), and pre-trained models were obtained from the Transformers (version 2.8.0) Python package (Transformers, 2020).

2.6. Cross-validation

To minimize any bias due to the relatively small data samples, our experiments were cross-validated when possible. To define the cross-validation procedure, we considered the adopted FE methods. On one hand, approaches from group III, i.e., including FE methods based on DL with training on task-specific data, require cross-validation during the stage in which the FEs are trained (henceforth called the first training stage). On the other hand, all analyzed approaches require cross-validation during ML classification (henceforth called the second training stage). Therefore, we concluded that for the sentiment@USNavy data set, a two-stage approach for cross-validation was necessary. In this case, we applied a five-fold cross-validation, with the assumption that the test splits were the same for both training stages, i.e., these data were not presented to the models during training. In the first training stage, the training data were divided five times into DL_train and DL_validate splits. In the second training stage, the DL_train and DL_validate splits were combined to form the training split for ML classification.

For the SemEval-2017 data, we were obliged to proceed differently. In this case, the test and training splits were predefined by the authors of the competition; thus, in the second training stage, no cross-validation was possible for approaches utilizing FEs from groups I or II. However, for the group III FEs, we performed five-fold cross-validation by dividing the original training set into DL_train and DL_validate splits. ML classifiers were trained according to the data split five times, regardless of the FE type.

Table 2
Training parameters of the adopted XGBoost classifier.

Parameters	Value
objective	multi:softprob
n_jobs	24
learning_rate	0,03
max_depth	10
subsample	0,7
colsample_bytree	0,6
random_state	2 020
n_estimators	250
tree_method	gpu_hist

2.7. Training of group III FEs

DL FEs that used a bidirectional LSTM for creating tweet-level vector representations were trained in the same manner as in Fiok et al. (2020), and when DL LMs from the transformer model family were fine-tuned to later use their particular classification [CLS] output, we have utilized another set of parameters. All DL FE training parameters with values specific to this study are presented in Table 3, and the undisclosed parameters were set to the default values proposed by the Flair framework.

As with other models, we did not optimize the above training parameters; instead, these values were established based on previous research. However, to demonstrate that parameter optimization is a challenging task and to show the extent of its advantages for transformer model performance, we performed a side experiment illustrating the influence of five selected MBS values for RoBERTa large when applied to the SemEval-2017 data set.

2.8. Statistical analysis of results

As some of the tested models provided very similar results, we decided to conduct bootstrap statistical analysis to assert their significance. We focused on the decisive MMAE metric and carried out the following procedure: 1) bootstrap the distribution of the metric value for the best performing model. In all cases, we have resampled and computed the metric value 10 000 times, and 2) compute the statistical significance of differences from all other models. The resulting p values were used to mark the obtained MMAE value according to the standard

Table 3
Training parameters of DLFE utilizing bidirectional LSTMs and fine-tuned transformer models.

	Parameters	Value
LSTM training	initial learning rate	0.1
	minimal learning rate	0.002
	annealing rate	0.5
	mini-batch size (MBS)	8
	hidden size	512
	shuffle data during training	TRUE
	optimizer	SGD
Fine-tuning	initial learning rate	3e-06
	MBS	8
	maximum number of epochs	4
	minimal learning rate	3e-06
	patience	3
	optimizer	Adam

approach regarding p-value significance levels, i.e., 1) when p was greater than 0.1, we assumed not statistically significant differences between models, 2) between 0.1 and 0.01 we marked the MMAE value with an '*' sign to denote weak significance, 3) between 0.01 and 0.001 '**' was used, and 4) if p was lower than 0.001 the '***' sign was used to denote highly significant differences.

2.9. Explaining model decisions

To provide an improved understanding of the rationale of the ML model predictions, we used SHAP (version 0.35.0), a state-of-the-art XAI technique. Specifically, we used the SHAP tree explainer (Lundberg et al., 2020) to generate visualizations of model-level explanations for several selected GB model variants.

We also demonstrate the use of a recent tool, BertViz (Bertviz., 2020), designed specifically for transformer LMs, which provides a visualization of connections between tokens that are identified by so-called "attention" mechanisms of these LMs in an analyzed sample tweet.

2.10. The influence of topic-related information on sentiment predictions

The concept of analyzing information regarding the topic of a tweet in combination with its sentiment is not new. In this context, at least two types of approaches have been developed to benefit from the knowledge of tweet topic: 1) (Xue et al., 2020; Si et al., 2013) demonstrate an approach in which they utilize tweet-related topic information extracted through unsupervised ML techniques such as Latent Dirichlet Allocation (LDA) (Blei et al., 2003) to discuss aspect-level sentiment, i.e., gain the ability to analyze not only general sentiments, but also the target of the sentiment, and 2) utilizing topic information as features in the ML sentiment prediction process to increase model performance. This approach was addressed by Cliche (2017), where the dataset-specific method to extracting topics was applied, and by Ren et al. (2016) and Xiang and Zhou (2014), who utilized features extracted by LDA together with various word embeddings. Focusing on the second type approach, all the mentioned researchers reported an increase in prediction performance. However, they all benefited from sentiment prediction models outside the deep learning transformer architecture. Since we were unable to find a study investigating if topics extracted from tweets through LDA can increase recent transformer models' prediction performance, we decided to address this in our work. For this purpose, we have used the Gensim Python Package (version 3.7.2) (Gensim, 2020). LDA is known to provide varying results depending on three key parameters that have to be chosen manually, i.e., number of topics that the documents should be divided into, and alpha and eta (also called "beta"). Therefore, it is required to perform a grid search over the above parameters and select the appropriate LDA model. Our study carried out a grid search covering 450 parameter combinations, including a number of topic groups ranging from 5 to 20. Often, the adopted LDA model's quality and resulting topics are measured by the coherence score, which is a measure of semantic similarity between keywords defining each topic. However, sole optimization of this metric can yield unconvincing results, for example, when coherence scores are very similar for models adopting 5, 10, and 20 topic groups. In such a situation, the researcher or AI practitioner should carefully consider other factors.

3. Results and discussion

This section discusses the results obtained in the main experiment, side experiments, and the possible implementation of XAI for sentiment analysis in Twitter.

3.1. Main experiment

Table 4 shows that the performances of selected FE models on the sentiment@USNavy data set differ strongly between groups in all metrics. Generally, EFE models exhibit the lowest performance, DL FE models without training on task-specific data provide some improvements, and trained DL FE models obtain the best results. The results for selected FE models on the SemEval-2017 data set are presented in Table 5 and are in agreement with these conclusions.

We can also draw the following detailed observations:

1) For both data sets, the best results were obtained by the FE models that used the special classification [CLS] output of fine-tuned transformer models to create tweet-level vector representations.

Table 4
Comparison of model performance for the sentiment@USNavy data set. The best results are highlighted in bold. Considering the MMAE metric and the best Fine-tuned BART large CNN model, there was no statistically significant difference with the Fine-tuned RoBERTa large (p = 0,3295), and Fine-tuned XLM-R (p = 0,1604) models, whereas RoBERTa large LSTM performed poorer (p = 0,0093) and all other models significantly (p = 0,0) poorer.

@USNavy Data Set							
FE Group	FE Type	FE Model	MMAE	MAE	EMD	MCC	F1 Macro
I	EFEs	TF	0.832***	0.392	0.247	0.466	0.377
		LIWC	0.783***	0.394	0.145	0.466	0.389
		SEANCE	0.756***	0.377	0.155	0.479	0.395
II	DL without training on task-specific data	Pooled FastText	0.783***	0.373	0.2	0.487	0.379
		Pooled RoBERTa	0.681***	0.328	0.18	0.541	0.449
		USE	0.701***	0.325	0.182	0.541	0.41
III	DL with training on task-specific data (token embeddings converted into tweet-level embeddings by trained LSTM)	FastText LSTM	0.701***	0.353	0.114	0.522	0.445
		RoBERTa large LSTM	0.536**	0.297	0.058	0.588	0.561
	DL with fine-tuning on task-specific data (tweet-level embeddings provided by built-in-transformer LM [CLS] output)	Fine-tuned RoBERTa large	0.507	0.278	**0.05**	0.615	0.587
		Fine-tuned BERT large uncased	0.617***	0.302	0.088	0.588	0.518
		Fine-tuned BERT large cased	0.609***	0.297	0.097	0.585	0.491
		Fine-tuned XLNet large cased	0.582***	0.288	0.089	0.598	0.525
		Fine-tuned BART large CNN	**0.5**	**0.268**	0.051	**0.626**	**0.596**
		Fine-tuned XLM-R	0.515	0.269	0.062	**0.626**	0.592
		Fine-tuned XLM MLM en 2048	0.594***	0.281	0.085	0.604	0.496

Table 5
Comparison of model performance for the SemEval-2017 data set. The best results are highlighted in bold. Considering the MMAE metric, there was no statistically significant difference between the best Fine-tuned XLM-R and Fine-tuned RoBERTa large (p = 0,2016) models, whereas all other models achieved significantly (p = 0.0) poorer performance when compared to the best model.

SemEval-2017 Task 4							
FE Group	FE Type	FE Model	MMAE	MAE	EMD	MCC	F1 Macro
I	EFEs	TF	1.372***	0.723	0.699	0.046	0.138
		LIWC	1.167***	0.617	0.51	0.167	0.215
		SEANCE	1.133***	0.601	0.497	0.163	0.224
II	DL without training on task-specific data	Pooled FastText	1.129***	0.595	0.507	0.2	0.228
		Pooled RoBERTa	1.075***	0.576	0.505	0.23	0.248
		USE	1.01***	0.55	0.478	0.225	0.246
III	DL with training on task-specific data (token embeddings converted into tweet-level embeddings by trained LSTM)	FastText LSTM	1.013***	0.559	0.393	0.213	0.275
		Roberta large LSTM	0.763***	0.496	0.301	0.308	0.382
	DL with fine-tuning on task-specific data (tweet-level embeddings provided by built-in-transformer LM classification [CLS] output)	Fine-tuned RoBERTa large	0.662	**0.452**	0.265	**0.344**	0.417
		Fine-tuned BERT large uncased	0.863***	0.483	0.316	0.294	0.34
		Fine-tuned BERT large cased	0.798***	0.472	**0.236**	0.311	0.365
		Fine-tuned XLNet large cased	0.753***	0.47	0.295	0.317	0.393
		Fine-tuned BART large CNN	0.685***	0.455	0.256	0.337	0.406
		Fine-tuned XLM-R	**0.656**	0.458	0.268	0.333	**0.433**
		Fine-tuned XLM MLM en 2048	0.732***	0.47	0.281	0.314	0.395

2) For both data sets, no single transformer model achieved the best results for all metrics; rather, the best FE varied for different metrics.
3) When DL FEs were used as out-of-the-box solutions, i.e., without any task-specific training, the USE and RoBERTa provided similar results, with the former achieving slightly higher performance.
4) Comparison of the EFE models indicates that SEANCE seems to slightly outperform LIWC and TF in both data sets. It should be mentioned, however, that the TF method is prone to performance changes when parameter tuning procedures are applied, and the latter two methods are totally non-trainable. Moreover, the TF method is known to provide good results with ML classifiers beyond the GB classifier used here.
5) The sentiment analysis task, as defined in the sentiment@USNavy data set, is easier than the analogical task for the SemEval-2017 data set, as all models in all metrics achieve better results for the former. Presumably, this is a result of the following facts: a) the instances in the sentiment@USNavy data set are probably more similar when the use of language is concerned as a result of the single domain that is addressed, b) the analyzed SemEval-2017 data set comprises of over 26 000 data instances gathered from different sources, which results in different ways of using language by different sources and thus less satisfactory prediction performance, and c) in the sentiment@USNavy data set the division of data between training and testing was different than in SemEval-2017. In the latter, a vast majority of data instances are included in the testing set, which creates a situation where the models are allowed to train on a small portion of data and are later tested extensively, influencing final metric values.

By adopting the metrics employed in SemEval-2017, we can compare our results to those obtained by the teams participating in that competition. By comparing our best performing fine-tuned transformer models based on the primary MMAE metric measured in the 4C SemEval-2017 sub-task, we find that XLM-R is in 6th place while based on the secondary MAE metric RoBERTa large is in 1st place out of 15 competitors (Rosenthal et al., 2019). For sub-task 4E, the best BERT large cased model takes 1st place based on the EMD measure. We again note that our models were not optimized with the aim of winning the SemEval-2017 competition; therefore, no extraordinary data set-specific preprocessing steps were taken, and no model parameter optimization was performed. Moreover, we used only the data published in the SemEval-2017 competition, whereas the actual competitors were allowed to use other data, i.e., data released for previous SemEval editions. In contrast, as already mentioned in the introductory section, the winning team, which used an ensemble of 20 DL FE models based on CNNs and LSTMs fed FastText embeddings, reported additional tricks to improve their final score (Cliche, 2017).

Importantly, our results are consistent between the two compared data sets, namely, the transformer models clearly achieve the best results. This finding indicates that we have most likely avoided data set-related bias in our study.

3.2. Side experiment

One should remember the possible bias in FE models from group III. For this group, no model-specific optimization of training parameters was performed, and thus, the adopted set of variables could favor one model over another. To illustrate the extent to which DL model performance can be hindered or improved by parameter selection, we conducted a small side experiment, i.e., an example effort to optimize a single model parameter. For the FE using the RoBERTa large LM, we evaluated the effect of five selected MBS values on performance. The results of this fivefold cross-validated side experiment are presented in Table 6. The results indicate that when considering a single metric, i.e., the MMAE, the differences exceed 0.054 points with the selected LM,

Table 6
Results of our side experiment. Example of MBS optimization for the RoBERTa large transformer model in the fine-tuning procedure for the SemEval-2017 data set. Considering the MMAE metric, there was no statistically significant difference between batch sizes of 8 and 4 (p = 0.2163), however all other variants compared to batch size of 8 were significantly different (p = 0.0).

MBS	MMAE	MAE	EMD	MCC	F1 Macro
4	0.671	0.444	0.257	0.344	0.418
8	**0.662**	0.452	0.265	0.344	0.417
12	0.686***	0.457	0.28	0.335	0.413
16	0.704***	0.452	0.269	0.336	0.397
32	0.725***	0.465	0.28	0.329	0.394

which could mean a difference of achieving the 6th or 7th place in the leaderboard of the SemEval-2017 competition. When the MMAE and EMD are considered, it can be observed that not all metrics change in the same manner as the MBS is varied from 4 to 8; here, the MMAE is better for MBS = 8, and the EMD is better for MBS = 4. Thus, we hypothesize that parameter optimization should be metric-specific, which is in agreement with other works such as that on NLP by Munson et al. (2005) and that on ML by Zhao et al. (2018). We also note that optimizing the DL LM parameters is time-consuming, as the side experiment results reported in Table 6 required a duration of almost 10 h to acquire via our computer.

3.3. XAI for sentiment analysis in Twitter

We would also like to discuss the explainability of selected FE methods. Models that are based on features understood by humans can easily benefit from XAI methods developed for ML models. An example of model-wise rationale for predictions provided by the SHAP technique and the SEANCE FE model used for the sentiment@USNavy data set are presented in Fig. 2. The features extracted by the SÉANCE model have meaningful names (precise definitions are available in (Crossley et al., 2017)) that allow utilizing SHAP explanations for formulating conclusions such as "the highest impact on model output was caused by the 'vader compound' feature and the 2nd-most informative feature was 'negative adjectives component'". This, in turn, allows the AI practitioner to optimize the final model, excluding features of marginal importance.

The provided visualization was created for a single model, i.e., it was computed for a single data set and thus is not cross-validated. This visualization allows us to observe the average extent to which the five most important features contributed to the decisions of the ML model. We believe that such presentations can be helpful, for instance, during feature engineering or discussions of model performance. For the model explanations presented in Fig. 2, we can conclude that VADER features play a very important role among all lexicon-based features extracted by SAENCE.

When DL FEs are considered, the SHAP method allows us to generate plots similar to those in Fig. 3. As the feature names provided by the USE model do not allow any specific human interpretation, probably the only possible conclusions based on this SHAP explanation could state that the impact of one feature was more significant than the others. Other DL FEs used in our study also do not allow humans to understand what each given feature represents. This fact indicates a serious limitation of high-quality DL FEs, namely, they do not enable explanations regarding rationale for ML model predictions. For text representations created by LSTMs based on LMs that provide simple static word embeddings (i.e., that do not change with context of the token in a sentence), it is possible to create instance-level visualizations of rationale for model predictions, as shown in Li et al. (2015) and Arras et al. (2017). Karpathy (2015) also showed that such visualizations are possible for character-level LMs with recurrent neural networks. Unfortunately, these prediction models do not provide state-of-the-art performance. In addition, the methods presented by the above-mentioned researchers are not popular; to our knowledge, there are no ready-to-use software packages that would allow easy application. For the recent complex context-aware methods used to create token representations, i.e., based on the transformer model proposed by Vaswani et al. (2017), Vig (2019) proposed a method for visualizing the focus of the so-called attention mechanism used in these models. The method enables one to inspect each layer and attention head of a transformer model, which, for an example of a RoBERTa large LM with 24 layers and 16 heads, results in 384 possible combinations for visualization. This method can offer tweet-level insights, as presented in Fig. 4, which allows one to verify whether the connections between tokens are correctly identified by the assessed model. When considering the tokens that are visibly connected with the [CLS] token, which indicates that they influence the tweet-level embeddings outputted by the transformer model for tweet classification, the tokens identified by the model are "Thank," "wonderful," "Dad," and "e" (beginning of "eulogy," which was unexpectedly divided into separate tokens). Among other observations, "sharing" and "memories" are rightfully strongly connected with each other, and the same phenomenon occurs for "your" and "Dad." From the above observations, we can conclude that the attention mechanisms correctly identified connections between tokens and, more importantly, the tokens influencing the [CLS] token are truly the key tokens while "for" or "the" were reasonably not connected with [CLS].

We believe that the presented instance-level visualizations are helpful; however, these methods are only an initial step towards easy everyday usage. At present, computing and interpreting a single tweet-level visualization is highly time-consuming. We hope that future

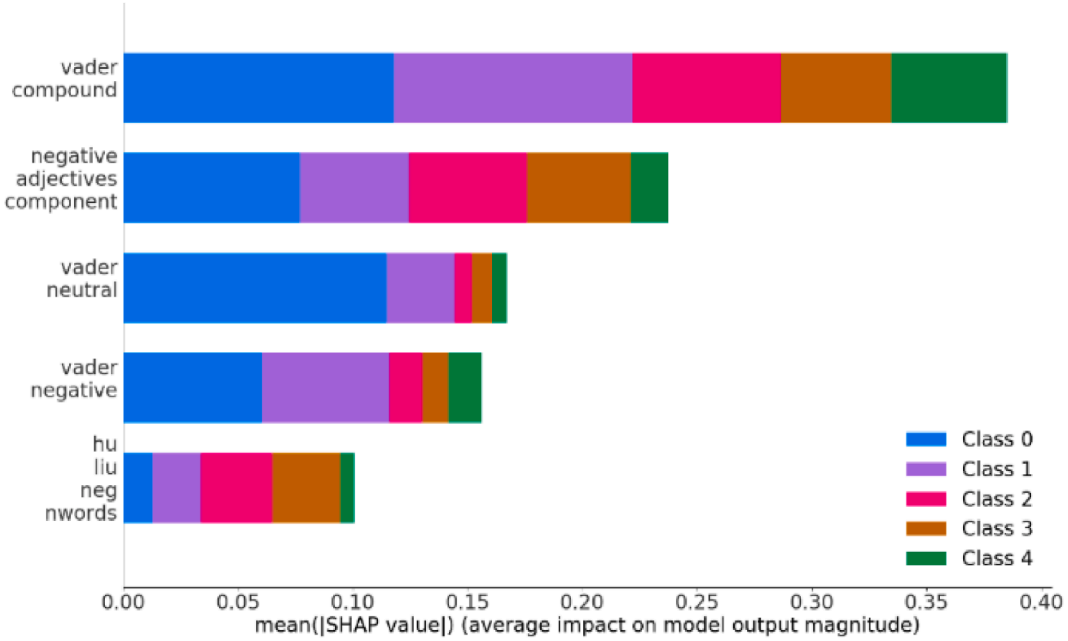

Fig. 2. SHAP explanations for a GB model trained on SEANCE features for the sentiment@USNavy data set.

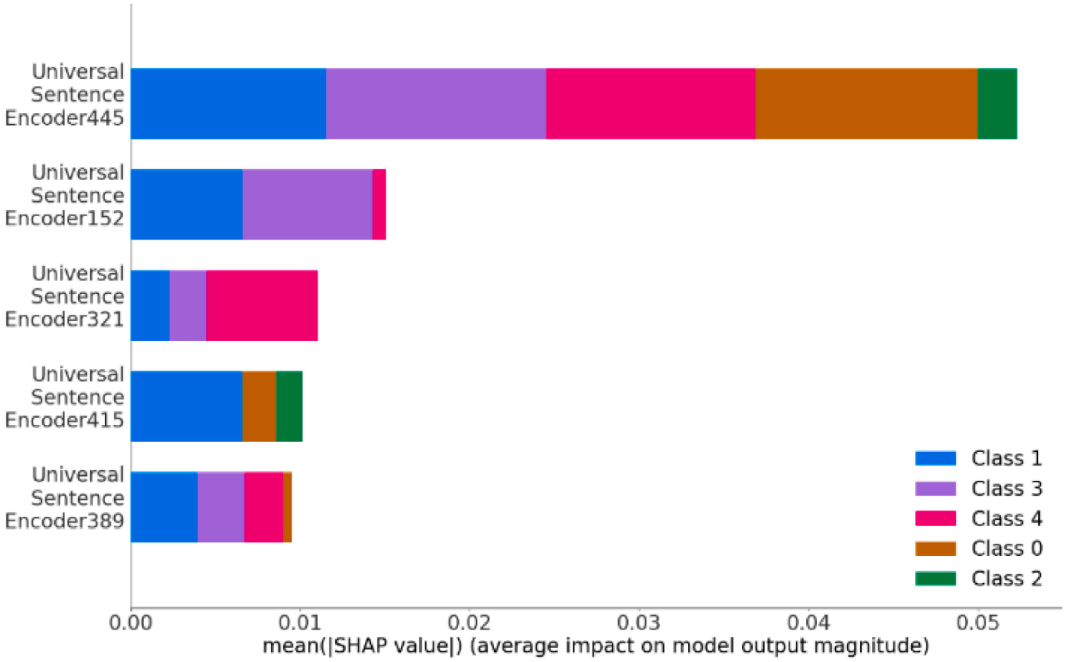

Fig. 3. SHAP explanations for a GB model trained on USE features used on the sentiment@USNavy data set.

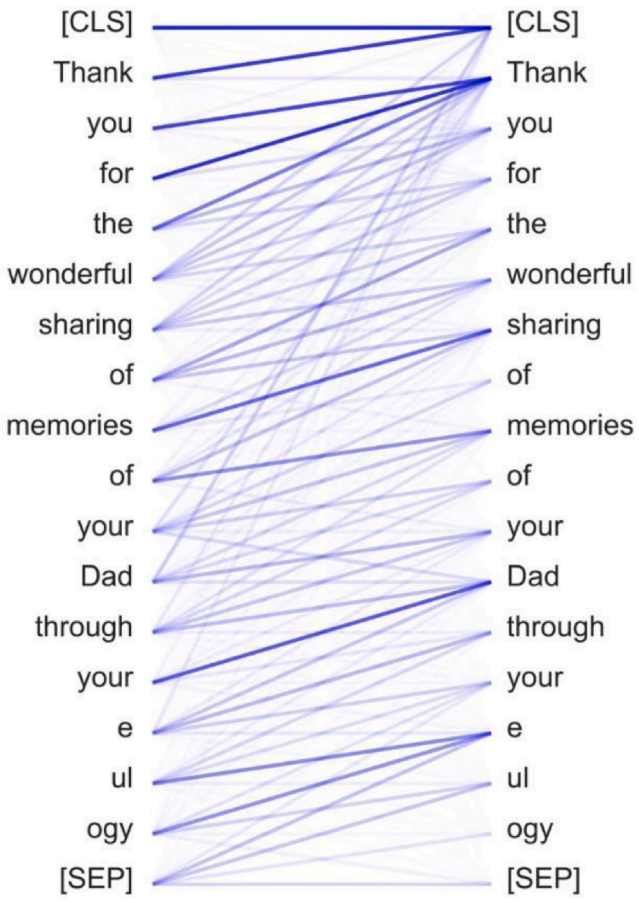

Fig. 4. Visualization of connections between tokens, as represented by attention mechanisms in the pre-trained RoBERTa base model on an example tweet from the sentiment@USNavy data set. Layer 0, Head 0 was selected for visualization. The figure was generated with BertViz repository cloned from github (BertViz, 2020).

techniques will enable some type of model-wise measure of the accuracy of connections between tweet tokens and the outputted [CLS] token.

3.4. Second side experiment – the influence of topic-related information

To investigate the impact of topic-related information extracted through unsupervised ML methods on sentiment prediction based on a selected recent transformer model, we have carried out a second side experiment, which included working with LDA models. As mentioned in section 2.11, we began with a grid search of the adopted LDA model parameters, which resulted in several parameter combinations with similar coherence scores, as presented in Table 7. Furthermore, we decided to use topics inferred for each data instance by LDA models, which assumed five and eleven topics, because adopting five topics mimics the number of sentiment classes and eleven because this variant yielded the highest coherence score.

In the next step, the topics discovered by LDA models for each data instance were fed as features to our ML sentiment classification pipeline. To enable quality comparison with the selected transformer model, we analyzed three feature combinations: 1) LDA topics as only features, 2) LDA topics together with features extracted by fine-tuned RoBERTa large model, and 3) only features from fine-tuned RoBERTa large model. The results of sentiment classification performance in these setups are presented in Table 8, which allow the following conclusions: 1) when the extracted LDA topics are analyzed alone by the ML classifier, they allow low prediction quality, and 2) comparing the use of features provided solely by the selected RoBERTa model and together with LDA topics doesn't allow to decide which model performed better. Therefore, our brief investigation allows hypothesizing that the use of LDA topics as features together with independent variables derived from textual data by a transformer model does not always improve prediction

Table 7
Selected results from the grid search of parameters for the LDA model.

Number of topics	Alpha	Eta (Beta)	Coherence score
5	asymmetric	0,91	0,554
8	asymmetric	0,91	0,543
11	asymmetric	0,91	0,578

Table 8
Comparison of sentiment classification performance of selected transformer model together with topics discovered by LDA as features computed for the sentiment@USNavy data set. The performance of 'Pure RoBERTa large fine tuned' variant was insignificantly (p = 0.47) different compared to the best model highlighted in bold, whereas the 'LDA topics only' model achieved significantly (p = 0,0) less satisfactory performance.

Feature set	MMAE	MAE	EMD	MCC	F1 Macro
LDA topics only	1,223***	0,706	0,592	0,156	0,199
Pure RoBERTa large fine tuned	0,528	0,275	0,063	0,615	0,571
LDA+RoBERTa large fine tuned	**0,527**	0,276	0,062	0,614	0,569

performance. However, we believe one has to restrain from drawing any more robust conclusions in this regard due to a number of factors: 1) only one small data set was analyzed, 2) only one arbitrary selected transformer model was used, 3) selection of LDA model parameters is always problematic and depends on the researcher, 4) there are numerous ways of the possible use of LDA model output as features, and in our study, we only used the most probable topic as feature. An example of another possible approach can be to use for each data instance the probabilities of belonging to all predefined topic groups. The ML classifier could benefit from more features provided by the same LDA model in such a case.

4. Study limitations

One limitation of this study is the previously mentioned possible bias resulting from a lack of FE parameter optimization. For older trainable FE methods, such as TF, the extent to which the lack of parameter optimization may influence performance has been studied in previous research, and an example of the extent to which a similar phenomenon affects recent transformer models was demonstrated in our side experiment.

Another potential source of bias is the selection of a single ML classifier for all FEs. It is possible that the selected GB classifier was more beneficial for some FEs than for others, and various ML classifiers could be compared.

Finally, the use of only simplified Twitter text preprocessing could hinder the performance of some FEs more than others. Again, this influence has been well studied for established LMs, such as n-gram (Singh and Kumari, 2016) while the influence on recent transformer models is unknown.

5. Conclusions

This study introduced a new data set for Twitter sentiment analysis tasks, composed of tweets directed to a single Twitter account. We demonstrated the utility of this data set by performing experiments with selected LMs. Future research should be conducted to mitigate possible bias in obtained results that origins from the unknown impact of several performance-influencing factors mentioned in the "study limitations" section. We also deployed various models, including a selection of recent transformer models, on the SemEval-2017 data set to demonstrate their performance. We found that even without elaborate optimization, additional training data, or text preparation, transformer models would achieve high ranks in the competition.

This study also addressed the question of whether prediction quality obtained with recent transformer models can be increased by using tweet topic information discovered by unsupervised ML methods? Our investigation brought us to a belief that this question remains open and should be a topic for more thorough research in the future. Finally, the results of our study show that the use of state-of-the-art XAI tools in Twitter sentiment analysis is possible but mostly limited to LMs based on human-understandable features. Research on XAI for transformer models is underway; however, the use and interpretation of available solutions is subject to improvements.

Funding

This study was supported in part by a research grant from the Office of Naval Research (N000141812559) and was performed at the University of Central Florida, Orlando, Florida.

CRediT authorship contribution statement

Krzysztof Fiok: Conceptualization, Methodology, Writing – original draft, Software. **Waldemar Karwowski:** Writing - review & editing, Supervision, Funding acquisition. **Edgar Gutierrez:** Writing - review & editing, Investigation. **Maciej Wilamowski:** Writing - review & editing, Investigation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Adadi, A., & Berrada, M. (2018). Peeking inside the black-box: A survey on Explainable Artificial Intelligence (XAI). *IEEE Accessed, 6*, 52138–52160.

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. J. (2011). Sentiment analysis of twitter data. In *Proceedings of the Workshop on Language in Social Media (LSM 2011)* (pp. 30–38).

Akbik, A., Bergmann, T., Blythe, D., Rasul, K., Schweter, S., & Flair, V. R. (2019). An easy-to-use framework for state-of-the-art nlp. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)* (pp. 54–59).

Alharbi, A. S. M., & de Doncker, E. (2019). Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information. *Cognitive Systems Research, 54*, 50–61.

Arras, L., Montavon, G., Müller, K. R., & Samek, W. (2017). Explaining recurrent neural network predictions in sentiment analysis. arXiv preprint arXiv:1706.07206.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., … Chatila, R. (2019). Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. arXiv, arXiv-1910.

Beel, J., Langer, S., Genzmehr, M., Gipp, B., Breitinger, C., & Nürnberger, A. (2013). October). Research paper recommender system evaluation: A quantitative literature survey. In *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation* (pp. 15–22).

Bertviz. (2020). Master branch commit 590c957799c3c09a4e1306b43d9ec10785e53745 from <https://github.com/jessevig/bertviz> (Accessed June 15, 2020).

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research, 3*(1), 993–1022.

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics, 5*, 135–146.

Cer, D., Yang, Y., Kong, S. Y., Hua, N., Limtiaco, N., John, R. S., … Sung, Y. H. (2018). Universal sentence encoder. arXiv preprint arXiv:1803.11175.

Cliche, M. (2017). Bb_twtr at semeval-2017 task 4: Twitter sentiment analysis with cnns and lstms. arXiv preprint arXiv:1704.06125.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., … Stoyanov, V. (2019). Unsupervised cross-lingual representation learning at scale. arXiv preprint arXiv:1911.02116.

Crossley, S. A., Kyle, K., & McNamara, D. S. (2017). Sentiment analysis and social cognition engine (SEANCE): An automatic tool for sentiment, social cognition, and social order analysis. *Behavior Research Methods, 49*(3), 803–821. https://doi.org/10.3758/s13428-016-0743-z

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Fiok, K., (2020). Analysis of Twitter sentiment with various Language Models. Github <https://github.com/krzysztoffiok/twitter_sentiment>.

Fiok, K., Karwowski, W., Gutierrez, E., & Ahram, T. (2020). Predicting the volume of response to tweets posted by a single Twitter account. *Symmetry, 12*(6), 1054.

Gambino, G., & Pirrone, R. (2019). Investigating Embeddings for Sentiment Analysis in Italian.

Gensim Python Package. <https://radimrehurek.com/gensim/> <Accessed November 3, 2020).

Ghiassi, M., Skinner, J., & Zimbra, D. (2013). Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network. *Expert Systems with Applications, 40*(16), 6266–6282.
Giachanou, A., & Crestani, F. (2016). Like it or not: A survey of twitter sentiment analysis methods. *ACM Computing Surveys (CSUR), 49*(2), 1–41.
Go, A., Huang, L., & Bhayani, R. (2009). Twitter sentiment analysis. *Entropy, 17*, 252.
González, J. Á., Hurtado, L. F., & Pla, F. (2019). ELiRF-UPV at TASS 2019: Transformer Encoders for Twitter Sentiment Analysis in Spanish.
Hutto, C. J., & Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Eighth international AAAI conference on weblogs and social media*.
Ibrahim, R. (2019). TwitterBERT: Framework for Twitter Sentiment Analysis Based on Pre-trained Language Model Representations. *Emerging Trends in Intelligent Computing and Informatics: Data Science, Intelligent Information Systems and Smart Computing, 1073*, 428.
Karpathy, A. (2015). The unreasonable effectiveness of recurrent neural networks. *Andrej Karpathy Blog, 21*, 23.
Kouloumpis, E., Wilson, T., & Moore, J. (2011). Twitter sentiment analysis: The good the bad and the omg!. *Fifth International AAAI conference on weblogs and social media, Barcelona, Spain*.
Krippendorff, K. (2011). Computing Krippendorff's alpha-reliability.
Kumar, A., & Jaiswal, A. (2020). Systematic literature review of sentiment analysis on Twitter using soft computing techniques. *Concurrency and Computation: Practice and Experience, 32*(1), Article e5107.
Lample, G., & Conneau, A. (2019). Cross-lingual language model pretraining. arXiv preprint arXiv:1901.07291.
Language recognition chart. (2019, August). from <https://en.wikipedia.org/wiki/Wikipedia:Language_recognition_chart> (Accessed May 15, 2020).
Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., … Zettlemoyer, L. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461.
Li, J., Chen, X., Hovy, E., & Jurafsky, D. (2015). Visualizing and understanding neural models in nlp. arXiv preprint arXiv:1506.01066.
Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., … Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.
Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. In *Advances in neural information processing systems* (pp. 4765–4774).
Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., … Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence, 2*(1), 56–67.
Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
Miller, G. A. (1998). *WordNet: An electronic lexical database*. Cambridge, MA: MIT Press.
Mishra, S., & Mishra, S. (2019). 3Idiots at HASOC 2019: Fine-tuning Transformer Neural Networks for Hate Speech Identification in Indo-European Languages.. *Proceedings of the 11th annual meeting of the Forum for Information Retrieval Evaluation (December 2019)*.
Munson, A., Cardie, C., & Caruana, R. (2005, October). Optimizing to arbitrary NLP metrics using ensemble selection. In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing (pp. 539-546). Association for Computational Linguistics.
Pagolu, V. S., Reddy, K. N., Panda, G., & Majhi, B. (2016). Sentiment analysis of Twitter data for predicting stock market movements. In *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)* (pp. 1345–1350). Paralakhemundi, India: IEEE. https://doi.org/10.1109/SCOPES.2016.7955659.
Pak, A., & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. *LREc, 10*(2010), 1320–1326.
Pennebaker, J. W., Francis, M. E., & Booth, R. J. (2001). Linguistic inquiry and word count: LIWC 2001. *Mahway: Lawrence Erlbaum Associates, 71*.
Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532–1543).
Potamias, R. A., Siolas, G., & Stafylopatis, A. G. (2019). A Transformer-based approach to Irony and Sarcasm detection. arXiv preprint arXiv:1911.10401.
XGboost Python Package Introduction. (2020). from <https://xgboost.readthedocs.io/en/latest/python/python_intro.html> (Accessed May 15, 2020).
Ren, Y., Wang, R., & Ji, D. (2016). A topic-enhanced word embedding for Twitter sentiment classification. *Information Sciences, 369*, 188–198.
Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should i trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135–1144).
Rosenthal, S., Farra, N., & Nakov, P. (2019). SemEval-2017 task 4: Sentiment analysis in Twitter. arXiv preprint arXiv:1912.00741.
Saif, H., He, Y., & Alani, H. (2012). Semantic sentiment analysis of twitter. In *International semantic web conference* (pp. 508–524). Berlin, Heidelberg: Springer.
Schwarz, S., Theóphilo, A., & Rocha, A. (2020). EMET: Embeddings from multilingual-encoder transformer for fake news detection. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 2777–2781). IEEE.
Scipy.stats.wasserstein_distance. (2020) from <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.wasserstein_distance.html> (Accessed May 15, 2020).
SemEval-2017 Task 4. (2020). from <https://alt.qcri.org/semeval2017/task4/> (Accessed May 15, 2020).
Severyn, A., & Moschitti, A. (2015). August). Twitter sentiment analysis with deep convolutional neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 959–962).
Si, J., Mukherjee, A., Liu, B., Li, Q., Li, H., & Deng, X. (2013). August). Exploiting topic-based twitter sentiment for stock prediction. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (pp. 24–29).
Singh, T., & Kumari, M. (2016). Role of text pre-processing in twitter sentiment analysis. *Procedia Computer Science, 89*, 549–554.
Sklearn.metrics.mean_absolute_error. (2020). from <https://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_absolute_error.html> (Accessed May 15, 2020).
Song, Y., Wang, J., Liang, Z., Liu, Z., & Jiang, T. (2020). Utilizing BERT intermediate layers for aspect based sentiment analysis and natural language inference. arXiv preprint arXiv:2002.04815.
Sousa, M. G., Sakiyama, K., de Souza Rodrigues, L., Moraes, P. H., Fernandes, E. R., & Matsubara, E. T. (2019). BERT for stock market sentiment analysis. In *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)* (pp. 1597–1601). IEEE.
Transformers. (2020). from <https://huggingface.co/transformers/index.html> (Accessed May 15, 2020).
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998–6008).
Vig, J. (2019). A multiscale visualization of attention in the transformer model. arXiv preprint arXiv:1906.05714.
Wang, H., Can, D., Kazemzadeh, A., Bar, F., & Narayanan, S. (2012). A system for real-time twitter sentiment analysis of 2012 us presidential election cycle. In *Proceedings of the ACL 2012 system demonstrations* (pp. 115–120). Association for Computational Linguistics.
Xiang, B., & Zhou, L. (2014). June). Improving twitter sentiment analysis with topic-based mixture modeling and semi-supervised training. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (pp. 434–439).
Xue, J., Chen, J., Chen, C., Zheng, C., Li, S., & Zhu, T. (2020). Public discourse and sentiment during the COVID 19 pandemic: Using Latent Dirichlet Allocation for topic modeling on Twitter. *PLoS ONE, 15*(9). e0239441.
Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., & Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems* (pp. 5754–5764).
Zhao, S., Fard, M. M., Narasimhan, H., & Gupta, M. (2018). Metric-optimized example weights. arXiv preprint arXiv:1805.10582.