Predicting Sentiment of Polish Language Short Texts

Aleksander Wawer
Institute of Computer Science
Polish Academy of Sciences
Jana Kazimierza 5
01-248 Warszawa, Poland
axw@ipipan.waw.pl

Julita Sobiczewska
University of Warsaw
Krakowskie Przedmieście 26/28
00-927 Warszawa, Poland
j.sobiczewska96@gmail.com

Abstract

The goal of this paper is to use all available Polish language data sets to seek the best possible performance in supervised sentiment analysis of short texts. We use text collections with labeled sentiment such as tweets, movie reviews and a sentiment treebank, in three comparison modes. In the first, we examine the performance of models trained and tested on the same text collection using standard cross-validation (in-domain). In the second we train models on all available data except the given test collection, which we use for testing (one vs rest cross-domain). In the third, we train a model on one data set and apply it to another one (one vs one cross-domain). We compare wide range of methods including machine learning on bag-of-words representation, bidirectional recurrent neural networks as well as the most recent pre-trained architectures ELMO and BERT. We formulate conclusions as to cross-domain and in-domain performance of each method. Unsurprisingly, BERT turned out to be a strong performer, especially in the cross-domain setting. What is surprising however, is solid performance of the relatively simple multinomial Naive Bayes classifier, which performed equally well as BERT on several data sets.

1 Introduction

Automated sentiment analysis usually involves training machine learning or deep learning models in supervised fashion. Typically, studies involve one data type and report high accuracy. For example, the seminal machine learning studies on IMDB movie reviews of (Pang et al., 2002) indicated accuracy over 80%. Recently, the accuracy of a deep learning system reported on this data set exceeded 95% (Howard and Ruder, 2018). Similarly, sentence-level sentiment predictions exceeded 95% accuracy on binary version of the popular Stanford Sentiment Treebank (abbreviated as SST-2) (Liu et al., 2019).

However, multiple studies indicate that the models trained on such data sets are in fact very far from being applicable universally. Machine and deep learning models tend to fit to specific type of texts and language. When applied to different language types in terms of both style and vocabulary, the performance drops sharply. This issue is often explored under the name of domain dependency or data set dependency. Several methods have been proposed to address it, such as for example (Selmer et al., 2013). In another stream of related studies, authors considered the task of cross-domain sentiment analysis: adaptation of a model to target domain or text type that is different from the domain or text type that the model was trained on. Examples include (Peng et al., 2018).

The goal of this article is to evaluate and compare supervised techniques of sentiment analysis of short texts using all available resources in the Polish language. We utilize three generations of machine learning:

- machine learning models using bag-of-words vector representations, algorithms such as Naive Bayes and Support Vector Machines,
- recurrent deep neural networks based on LSTM with and without pre-trained word embeddings,
- finally the most recent generation of deep neural networks, pre-trained on language modeling tasks (BERT and ELMO).

Proceedings of Recent Advances in Natural Language Processing, pages 1321–1327,
Varna, Bulgaria, Sep 2–4, 2019.
https://doi.org/10.26615/978-954-452-056-4_151

The questions that our paper addresses are: (1) to what extent is each of the method usable for training a universal sentiment analysis model (how well it performs in cross-domain setting) and (2) how good it is for predicting sentiment within the same text type it has been trained on (how well it performs in in-domain setting).

The motivation for a universal classifier might not be obvious at first, as clearly the best performance is achieved by in-domain classification (Twitter may need tweet classifier, Facebook needs posts classifier, IMDB needs review classifier, and so on). However, universal character of sentiment classification models allows for better performance in cases when source data distribution is different than target data distribution. This happens on every day basis, for instance Twitter posts change their topic over time and optimum performance requires new train sets to match the newest data. These issues are well-known to machine learning community and explored under topics of domain adaptation, dataset shift and semantic drift. Ability to apply the model universally to various data types is a great advantage from the practical point of view.

Our intention is to focus on supervised learning. Specifically, this means working on collections of short texts such as single sentences, tweets or short reviews with labeled sentiments. The goal of our task is to classify sentiment of the whole written utterance (as for example, a sentence, review or tweet) into three classes: as positive, neutral or negative.

When computing sentiment we rely only only on learning from provided text examples and do not use any resource which might help in sentiment predictions such as a sentiment dictionary.

The paper is organized as follows: Section 2 contains a description of sentiment datasets, Section 3 describes the methods used for sentiment prediction. Section 4 describes the results obtained in our experiments. Finally, Section 5 contains conclusions and a discussion of possible future work.

2 Data Sets

The experiments described in this paper are based on several resources with manually labeled sentiment. The texts in our experiments are "short": tweets do not exceed 140 characters, Treebank sentences are arbitrarily long in terms of tokens but syntactically and semantically correct, reviews contain from 1 to 3 sentences.

2.1 Polish Sentiment Treebank (TW)

The first resource is Polish language dependency treebank with sentiment annotations ("Treebank Wydźwięku" abbreviated as TW). It is available to download [1]. Similar to Stanford's Sentiment Treebank (SST) (Socher et al., 2013), Treebank Wydźwięku (TW) was intended to study compositional phenomena in sentiment analysis. There are several notable differences between the two:

- Dependency trees (TW) instead of constituency binary parse tress (SST),
- 3-class sentiment (TW) instead of 5-class (SST),
- Open-domain (TW) instead of one domain of movie reviews (SST).

In TW, overall sentiment of each sentence corresponds to the sentiment of its root. Sentences in this data set often contain mixed sentiment, even opposite polarities: one part may be positive and the other negative. This makes the task of predicting overall sentiment more difficult.

2.1.1 TW Version 1.0

Initial version of the TW treebank contained sentences from Składnica Treebank[2] (part referred to as **sklad**) and sentences from two types of product reviews: perfumes and clothes (part called **rev**).

The first release of the treebank was published as a part of Task 2: Sentiment analysis in PolEval 2017 campaign on evaluation of natural language processing tools for Polish. In this competition, submitted tools competed against one another within certain tasks selected by organisers, using provided data. Solutions were evaluated according to common procedures.

The intended use of the treebank was as follows: given a set of syntactic dependency trees, the goal was to provide the correct sentiment for each subtree (phrase). Phrases correspond to sub-trees of dependency parse tree. Annotations assign sentiment values to whole phrases (and in some cases, sentences), regardless of their type. The PolEval

[1] http://zil.ipipan.waw.pl/TreebankWydzwieku

[2] http://zil.ipipan.waw.pl/Sk%C5%82adnica

part of the treebank and related evaluation script may be freely downloaded[3].

Three systems participated in the tasks, all of them based on TreeLSTM (Tai et al., 2015). The description of systems and evaluation methodology can be found in (Wawer and Ogrodniczuk, 2017). Due to different nature of these tasks (computing sentiment of each sub-phrase and each sentence vs sentiment of sentences only) the results of these systems are not directly comparable to results reported in our paper.

2.1.2 TW Version 2.0

In August 2018, a new batch of sentences has been added to TW. It contains following new parts:

- Test (evaluation) sentences from PolEval 2017 sentiment task (part called **polevaltest**),
- 2 x 500 sentences collected from various web sources, mostly difficult, mixed sentiments and negative ones (parts called **neg** and **jun18**).

To our best knowledge, our paper is the first to describe and use the new version of this resource.

2.2 Twitter Data

This data set contains one thousand Polish language tweets, gathered and manually labeled as to their sentiment during the TrendMiner project[4]. Many tweets are related to publicly discussed events, some of them originate from politicians, journalists and public figures. Many of them are tweets of simple Twitter users, including teenagers. Overall, the dataset appears to be a fairly representative sample of communication occurring on Twitter in the Polish language. Due to Twitter's policy it can not be made publicly downloadable.

2.3 Movie Reviews

This data set consists of one thousand manually collected movie reviews from Polish website: http://www.filmweb.pl. Most of them are very short texts (1-2 sentences), the rest is a collection of reviews containing up to 5 sentences. Each review has a corresponding numeric score (number of stars) from 1 to 10, assigned by the review's author. Each category (number of stars) has exactly one hundred reviews.

All scores were re-scaled into three categories: stars from 1 to 3 were re-scaled into negative category (-1), stars from 4 to 6 into neutral (0), stars 7 to 10 into positive (1).

What should be noted, however, is that the reviews very often contain spelling mistakes, words with omitted Polish diactric marks, often contain slang or sarcasm. All that makes them problematic for automated analysis.

2.4 Label Frequencies

Most of the datasets are not balanced in terms of sentiment label distributions. Twitter data set contains mostly neutral texts (tweets). Filmweb's balance is nearly perfect, as only the positive class has slight advantage in terms of frequency. Also TW's balance is far from perfect distribution between three sentiment classes as most sentences are neutral. Some pieces of TW treebank have been deliberately created to address imbalance issues in the sentiment treebank, such as for example part of the data called neg in TW 2.0, which contains many negative sentences.

Table 1 presents sentiment label distribution in each of the data set.

Table 1: Distribution of sentiment classes in each data set

file	-1	0	1	all
jun18-TW	59	240	202	501
neg-TW	252	245	3	500
polevaltest-TW	40	215	95	350
rev-TW.	7	868	90	965
sklad-TW	3	230	2	235
twitter	80	854	66	1000
filmweb	300	300	400	1000

3 Machine and Deep Learning Methods

3.1 Bag-of-Words and Machine Learning

Machine learning methods described in this subsection used bag-of-words text representations. We converted text to word vectors using word-level unigram vectorizer with TF-IDF weights.

We have focused on two machine learning algorithms: Naive Bayes and Support Vector Machines.

[3] http://2017.poleval.pl/index.php/tasks/

[4] https://cordis.europa.eu/project/rcn/100752/factsheet/en

Multinomial Naive Bayes (NB) Classifier is a well-known supervised machine-learning algorithm with an assumption of independence among predictors.

Support Vector Machine (SVM) is also a well-known supervised machine learning algorithm. We used linear kernels and implementation from the liblinear library[5].

3.2 LSTM Neural Network (NN)

We used two approaches to implement the first layer of the neural network.

In the first approach we used untrained, random initialized embedding layer that uses 32 length vectors to represent each word. This method is marked as NN in the results.

In the second approach we changed this layer to pre-trained word2vec 100-dimensional word embeddings for Polish[6]. The embeddings were generated using gensim package (Řehůřek and Sojka, 2010) with skip-gram model architecture for two large text corpora: The National Corpus of Polish[7] and Polish edition of Wikipedia. We marked this approach as NN+E subsequently.

The structure of the neural network was as follows: word embedding layer, LSTM layer with 100 memory units, two Dense layers: the first with 100 neurons, the second with 3 output values, one for each class, with dropout regularization between them. As an output layer we used the softmax activation function.

3.3 ELMO

In one of the approaches we used ELMo method (Peters et al., 2018) to represent texts. ELMo, as it's authors put it, is a deep contextualized word representation that models both (1) characteristics of word use (e.g., syntax and semantics), and (2) how these uses vary across linguistic contexts (i.e., to model polysemy). These word vectors are extracted from states of a deep bidirectional language model, which is pre-trained on a large text corpus. We used the ELMO implementation and models described in (Che et al., 2018). The model was trained on Polish language Wikipedia.

ELMO, although currently often superseded by other alternatives, contributed to state-of-the-art results in multiple natural language processing tasks such as question answering or paraphrase detection.

To obtain ELMO representation of each text, we computed average vector from 3 neural network layers, which resulted in a vector of 512 numbers. In the second step, these vectors were used to classify sentiment of an input text. Here, we experimented with multiple well-known machine learning methods such as Logistic Regression (LR), Random Forest (RF) with 200 trees and Support Vector Machine classifier (SVC) with a linear kernel. Each of these variants is subsequently marked as ELMO-LR, ELMO-RF and ELMO-SVC.

3.4 BERT

BERT is an example of the newest generation of pre-trained neural networks based on transformer architecture (Devlin et al., 2018). BERT acronym stands for Bidirectional Encoder Representations from Transformers, also obtained state-of-the-art results on multiple NLP tasks such as natural language inference and question answering. BERT model comes pre-trained on a very large corpus of unlabeled data, can be subsequently fine-tuned to a task with a limited amount of data such as sentiment analysis.

In our experiments we used smaller version of BERT. It contains 110M parameters and has support for 104 languages, 12 layers, the size of each hidden layer is 768, 12 self-attention heads (bert-base-multilingual-cased). We set maximum sequence length parameter to 128, which is enough to cover several sentences, and tested 3 and 4 training epochs. We tested BERT in a scenario which adds a sequence classification head on top. BERT Transformer is pre-trained, the sequence classification head is only initialized and has to be trained.

4 Results

This section illustrates the results of three experimental modes tested in our paper.

4.1 In-Domain

In the first mode (in-domain), we examine the performance of models trained and tested on the same text collection using standard cross-validation. On one hand, the performance of in-domain is driven up by lexical and structural similarity: training data are likely to be similar to test data in terms of vocabulary and syntactic structures. On the other,

[5] https://www.csie.ntu.edu.tw/~cjlin/liblinear/

[6] http://dsmodels.nlp.ipipan.waw.pl/dsmodels

[7] http://nkjp.pl

					Method			
data set	NB	SVC	NN	NN+E	ELMO+LR	ELMO+RF	ELMO+SVC	BERT
jun18-TW	44.3%	44.3%	45.9%	45.9%	40.9%	44.7%	43.5%	**47.9%**
neg-TW	42%	42%	46%	44.8%	47.8%	46.8%	46.8%	**50%**
polevaltest-TW	55.4%	54.2%	**61.1%**	**61.1%**	49.1%	**61.1%**	46.9%	53.4%
rev-TW	88.2%	92.2%	92%	92%	**93%**	92%	90.8%	89.9%
sklad-TW	**100%**	**100%**	96.7%	96.7%	97.9%	97.9%	98%	97.8%
all-TW	**71.5%**	67.9%	71.2%	71.2%	67.3%	70.9%	63.2%	70.4%
twitter	84.6%	**85.4%**	84%	84%	75.5%	80.2%	70.5%	85.3%
filmweb	63.2%	**69.6%**	40%	40%	58.7%	55.2%	55.3%	40%

Table 2: In-domain average accuracy in 5-fold cross-validation

					Method			
data set	NB	SVC	NN	NN+E	ELMO+LR	ELMO+RF	ELMO+SVC	BERT
jun18-TW	**47.9%**	44.9%	42.7%	**47.9%**	43.9%	47.5%	43.5%	**47.9%**
neg-TW	**49%**	41.4%	40.8%	41.6%	40.6%	48%	37.8%	**49%**
polevaltest-TW	**61.4%**	57.4%	54.3%	58.9%	57.1%	60.9%	57.1%	**61.4%**
rev-TW	89.7%	76.7%	69%	82.5%	69.1%	88.4%	65%	**89.9%**
sklad-TW	**97.9%**	83.8%	73.2%	95.7%	74%	**97.9%**	66.4%	97.8%
twitter	85.3%	78.3%	64.1%	84.5%	60.1%	79.8%	52%	**85.4%**
filmweb	30%	37.7%	34.3%	30.2%	36.9%	30%	**39.1%**	30%

Table 3: One vs rest cross-domain accuracy

models do not utilize information contained in other available data sets.

The results of this mode are presented in Table 2. The best results were achieved by BERT (two treebank datasets: jun18 and neg) and SVC classifier (treebank dataset sklad, twitter and filmweb). Except for filmweb data set with over 30% discrepancy between the best and worst methods, the differences between methods were not large, usually did not exceed several percents.

On the whole TW sentiment treebank (marked as all-TW) the surprising winer was Naive Bayes, that managed to outperform other methods including BERT by a small margin.

4.2 One vs Rest Cross-Domain

In the second mode (one vs rest cross-domain) we train models on all available data except the given test collection, which we use as a test set. In this mode, models did not have a chance to learn from data similar to test data (maybe except some parts of TW treebank which may be considered similar). However, can utilize and possibly benefit from information contained in all other data sets available. Table 3 presents the results of one vs rest cross-domain experiments. In this mode, the best performers were BERT (the best results for 5 out 7 data sets) and surprisingly, Naive Bayes algorithm (best performance for 4 out of 7). Recurrent neural networks (NN) did not manage to reach state-of-the-art results, however it is easy to notice strong positive influence of pre-trained word2vec word embeddings (NN+E), with accuracy gains from several to as much as twenty percentage points in the case of twitter. One can also note that the performance of ELMO with Random Forest (ELMO+RF) is significantly better than the two other ELMO variants we tested.

4.3 One vs One Cross-Domain

	all-TW	tweets	filmweb
all-TW	-	70.4%	15.3%
tweets	85.4%	-	30.6%
filmweb	30%	30%	-

Table 4: One vs one cross-domain accuracy of BERT method

	all-TW	tweets	filmweb
all-TW	-	67.9%	32.1%
tweets	79.1%	-	37%
filmweb	38.9%	30.5%	-

Table 5: One vs one cross-domain accuracy of NB method

In the third mode (one vs one cross-domain), we train a model on one data set and apply it to another one, repeating for each combination. In this mode we tested only the highest performing methods, such as BERT and Multinomial Naive Bayes. In this experiment we merged all sub-parts of the sentiment treebank (jun18, neg, polevaltest, rev, sklad) into one data set presented as TW. It consists of 2500 sentences.

Table 4 contains one vs one results for BERT classifier. Rows refer to test data sets and columns to train data sets. As we can see, the accuracy of 85.4% on tweets with models trained on TW is high and identical to SVC of in-domain 5-fold cross-validation and also identical to BERT in one vs rest mode. The performance on filmweb indicates that the models did not start to learn effectively. Training on filmweb did not help the performance on TW treebank (only 15.3%). Since filmweb dataset is reasonable in size and balanced, we can only hypothesize that the language is too different.

Table 5 contains one vs one results for Naive Bayes classifier. As before, rows refer to test data sets and columns to train data sets. In some cases Naive Bayes outperformed BERT. It was the case with training on filmweb movie reviews, models trained on this data set performed better than BERT both on tweets and on sentiment treebank TW. Also in the scenario of training on TW and testing on filmweb, Naive Bayes turned out to be better.

5 Conclusions and Future Work

The main point of this paper was to use all available Polish language data sets to seek the best possible performance in supervised sentiment analysis of short texts. We compared three generations of methods: machine learning with bag-of-words representation, recurrent neural networks (with and without pre-trained word embeddings) and finally deep neural networks pre-trained on language modeling task, including the newest transformer architecture BERT.

In sentiment classification, data available at training time is often different from data we intend to analyze in production environments. Ideally, classifiers should be capable of predicting sentiment on multiple types of data, covering various topics and texts of varied length without the need of re-training. In practice, achieving the best possible performance requires training or re-training on data very similar to those we intend to analyze.

To explore the limits of this approach we experimented with cross-domain setting in which we train the model on one text type and apply it to another text type (one vs one cross-domain). We confirmed that this setting poses a problem often leading to substantial performance degradation.

Using several sentiment-labeled data sets as training data may in theory improve classifier's accuracy and robustness. In our paper we investigated possible benefits from training models on data less similar to the test set (cross-domain one-vs-rest mode) and compared this to model training on smaller amounts of highly similar data (in-domain, models trained on the same type of data).

We found that for some data sets (TW treebank sub-sets, twitter) the results are comparable for cross-domain and in-domain setting, while for movie reviews in-domain setting turned out to be almost 30% better. Here, similar training data played a more significant role than using other less similar data sets to learn from. The best-performing in-domain method turned out to be somewhat old SVC with simple bag-of-words representation.

BERT, transformer-based neural network with pre-training, turned out to benefit from large amounts of less similar data, with top performance in one-vs-rest cross-domain setting. Interestingly, multinomial Naive Bayes method turned out to perform on a very similar level with far less model parameters, which may be a viable alternative in more speed oriented environments without GPU processors.

Some of the issues raised in this paper are worth pursuing in further work. The first problem is the amount of pre-training and architecture changes needed to reach acceptable cross-domain performance. Apparently, this problem has still not been solved and more efforts are needed to reach high accuracy.

The second problem worth investigating is the matter of how suitable are sentiment treebanks, designed for experiments on within-sentence compositional sentiment phenomena (for example, studying sentiment propagation in sentence structure or mixed sentiments) for predicting single label of sentence-level sentiment. As a part of this future study, we intend to compare the performance of Tree-LSTM methods such as those reported in PolEval 2017 (Wawer and Ogrodniczuk, 2017) on sentence-level sentiment with methods reported in our work.

References

Wanxiang Che, Yijia Liu, Yuxuan Wang, Bo Zheng, and Ting Liu. 2018. Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Association for Computational Linguistics, Brussels, Belgium, pages 55–64. http://www.aclweb.org/anthology/K18-2005.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR* abs/1810.04805. http://arxiv.org/abs/1810.04805.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Melbourne, Australia, pages 328–339. https://www.aclweb.org/anthology/P18-1031.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *CoRR* abs/1901.11504. http://arxiv.org/abs/1901.11504.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics, pages 79–86.

Minlong Peng, Qi Zhang, Yu-gang Jiang, and Xuanjing Huang. 2018. Cross-domain sentiment classification with target domain specific information. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Melbourne, Australia, pages 2505–2513. https://www.aclweb.org/anthology/P18-1233.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, pages 2227–2237. https://doi.org/10.18653/v1/N18-1202.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. ELRA, Valletta, Malta, pages 45–50. `http://is.muni.cz/publication/884893/en`.

Øyvind Selmer, Mikael Brevik, Björn Gambäck, and Lars Bungum. 2013. NTNU: Domain semi-independent short message sentiment classification. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*. Association for Computational Linguistics, Atlanta, Georgia, USA, pages 430–437. https://www.aclweb.org/anthology/S13-2071.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 1631–1642. http://aclweb.org/anthology/D13-1170.

Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, pages 1556–1566. https://doi.org/10.3115/v1/P15-1150.

Aleksander Wawer and Maciej Ogrodniczuk. 2017. Results of the PolEval 2017 competition: Sentiment Analysis shared task. In Zygmunt Vetulani and Patrick Paroubek, editors, *Proceedings of the 8th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*. Fundacja Uniwersytetu im. Adama Mickiewicza w Poznaniu, Poznań, Poland, pages 406–409.