Journal of Intelligent Information Systems (2021) 57:601–626
https://doi.org/10.1007/s10844-021-00664-7

Leveraging contextual embeddings and self-attention neural networks with bi-attention for sentiment analysis

Magdalena Biesialska[1] · **Katarzyna Biesialska**[1] · **Henryk Rybinski**[2]

Received: 4 February 2021 / Revised: 12 July 2021 / Accepted: 26 July 2021 /
Published online: 11 December 2021

Abstract
People express their opinions and views in different and often ambiguous ways, hence the meaning of their words is often not explicitly stated and frequently depends on the context. Therefore, it is difficult for machines to process and understand the information conveyed in human languages. This work addresses the problem of sentiment analysis (SA). We propose a simple yet comprehensive method which uses contextual embeddings and a self-attention mechanism to detect and classify sentiment. We perform experiments on reviews from different domains, as well as on languages from three different language families, including morphologically rich Polish and German. We show that our approach is on a par with state-of-the-art models or even outperforms them in several cases. Our work also demonstrates the superiority of models leveraging contextual embeddings. In sum, in this paper we make a step towards building a universal, multilingual sentiment classifier.

Keywords Sentiment classification · Word embeddings · Transformer

Magdalena Biesialska and Katarzyna Biesialska have contributed equally to this work, which was mostly done at the Warsaw University of Technology.

✉ Katarzyna Biesialska
katarzyna.biesialska@upc.edu

✉ Henryk Rybinski
h.rybinski@ii.pw.edu.pl

Magdalena Biesialska
magdalena.biesialska@upc.edu

[1] Universitat Politècnica de Catalunya, Barcelona, Spain

[2] Warsaw University of Technology, Warsaw, Poland

1 Introduction

All areas of human life are affected by people's opinions and views, and thus the adage "opinions create preferences" turns out to be very true in the Internet era. The user generated content – often based on someone's personal experience or preferences rather than on facts – constantly grows in volume thanks to the increasing popularity of online review sites, social media, discussion groups, or blogs available on the Internet. Nowadays, people actively use information technologies to understand the opinions of others on a global scale, rather than limit themselves to seek out advice or recommendations only in their circle of family and friends, or rely only on television commercials and other media advertisements before purchasing a product. With the sheer amount of reviews and other opinions over the web, there is a need for automating the process of extracting relevant information. Sentiment analysis (SA) is a fascinating and a very practical task, not only from the research perspective but also business-wise. Companies strive to understand their customers – by analyzing how company's products or services are perceived, firms are able to improve their offer to better respond to customers' needs, and in result boost their sales. Therefore, research in sentiment analysis and its fast development in recent years is strongly correlated with the rapid growth of social media.

Natural language can be very subtle in its meaning. A single word can hardly convey the whole meaning of a statement – in natural languages the intended meaning is often implicit and depends on the context. Hence, understanding and representing the meaning of language is a really complex task, as language is highly symbolic, discrete and atomic in its nature. People share not only fact-based information, but also experience-based information, which is more subjective and emotional in nature. Complex linguistic phenomena – such as: sarcasm, humor, or bias, among others – can be manifested in many different ways; however, they are usually quite clear for humans and can be recognized by them without great difficulties or hardships. As opposed to machines, which often do not distinguish literal from figurative meaning, and thus can misinterpret statements.

This paper is an extended version of our conference paper (Biesialska et al., 2020). It contains elaborated description of the proposed method as well as in-depth quantitative and qualitative analysis. In addition, the paper is complemented with more thorough discussion and additional findings. We propose a novel approach to SA that builds on recent advances in deep neural networks and distributed word representations, i.e. self-attention and bi-attention mechanisms as well as contextual embeddings. Therefore, our model has a clear advantage over more traditional approaches as it requires no manual preprocessing or feature selection. Moreover, it achieves similar performance, both in accuracy and speed, as compared to the current state-of-the-art approaches. Our contribution can be summarized as follows:

- a sentiment classifier model achieving very good results comparable to state-of-the-art
- a novel architecture based on the transformer encoder with relative position representations
- unlike existing models, this work proposes a model relying solely on a self-attention mechanism and bi-attention

The paper is organized as follows: Section 2 introduces fundamental notions and concepts referring to the proposed method; Section 3 describes our approach; Section 4 discusses experimental setup and characteristics of particular datasets; Section 5 provides in-depth analysis of the results. Finally, Section 6 concludes this paper and outlines the future work.

2 Background and related work

2.1 The problem of sentiment analysis

Sentiment classification has been one of the most active research areas in natural language processing (NLP) and has become one of the most popular downstream tasks to evaluate performance of neural network (NN) based models. The task itself encompasses several different opinion related tasks, hence it tackles many challenging NLP problems (see e.g. Liu, 2012; Mohammad, 2016) such as detecting sentiment at various levels of text granularities, of the writer, reader or other entities mentioned or not explicitly mentioned in the text, distinguishing objective from subjective statements, detecting sarcasm, resolving anaphora, handling negation, word similarity, Named Entity Recognition (NER), Word Sense Disambiguation, to mention the most popular. The availability of standard benchmarks, such as well-known datasets of movie reviews (e.g. Pang & Lee, 2004; Maas et al., 2011), has significantly stimulated research in the area, so that opinion mining is nowadays one of the most popular NLP research topics.

2.2 Sentiment analysis approaches

The first fully-formed techniques for SA emerged around two decades ago, and continued to be prevalent for several years, until deep learning methods entered the stage. In this work, we focus on sentence-level polarity classification. The most straight-forward method, developed by Turney (2002), is based on the number of positive and negative words in a piece of text. Specifically, the text is assumed to have positive polarity if it contains more positive terms than negative ones. Of course, the term-counting method is often insufficient; therefore, an improved method was proposed by Kennedy and Inkpen (2006), which combines counting positive and negative terms with a machine learning (ML) approach (i.e. Support Vector Machine). A basic approach to deal with negation (e.g. Das & Chen, 2001; Pang et al., 2002; Potts, 2011) requires adding a "*NOT_*" prefix or a "*_NEG*" suffix to every term between a negator word and the first punctuation mark that appears after the negation word.

Various studies (e.g. Turney & Pantel, 2010) have shown that one can determine the polarity of an unknown word by calculating co-occurrence statistics of it. For instance, the Pointwise Mutual Information (PMI) measure can be employed to compute the co-occurrence of a word with another word. Moreover, there are classical solutions to the SA problem, which are based on lexicons. Traditional lexicon-based SA approach leverages word-lists that are pre-annotated with positive and negative sentiment. Therefore, for many years lexicon-based approaches have been utilized when there was insufficient amount of labeled data to train a classifier in a fully supervised way.

In general, ML algorithms are popular in determining sentiment polarity. The first ML model applied to SA has been implemented by Pang et al. (2002). Over the years, various variants of NN architectures have been introduced. An extensive discussion on several NN-based approaches for sentiment classification can be found in Wadawadagi and Pagi (2020). Notably, recursive neural networks, such as recurrent neural networks (RNN) (Socher et al., 2013; Tai et al., 2015; Kumar et al., 2016), or convolutional neural networks (CNN) (Kalchbrenner et al., 2014; Kim, 2014) have become the most prevalent choices.

Dynamic Memory Networks (DMN) proposed by Kumar et al. (2016), although intended to be used primarily for question answering, is a versatile hierarchical recurrent sequence model achieving state-of-the-art results in sentiment analysis. While a deep recurrent belief network is proposed by Chaturvedi et al. (2016).

Paulus et al. (2014) proposed Global Belief-Recursive Neural Network (GB-RNN) for granular sentiment analysis. In Chen et al. (2016) Adversarial Deep Averaging Network (ADAN) is presented; it leverages adversarial training for cross-lingual sentiment classification.

2.3 Vector representations of words

The recent success of ML algorithms is largely contingent on data representation (Maas et al., 2011; Bengio et al., 2013; Zhang & LeCun, 2015). More specifically, such vector representations are primarily used as features.

One of the principal concepts in linguistics states that related words can be used in similar ways (Firth, 1957). Clearly, words may have different meaning in different contexts. Nevertheless, until recently it has been a dominant approach, e.g. word2vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014), to learn representations such that each and every word has to capture all its possible meanings. However, recently a new set of methods to learn dynamic representations of words has emerged (McCann et al., 2017; Howard & Ruder, 2018; Peters et al., 2018; Radford et al., 2018; Devlin et al., 2019). These approaches allow each word representation to capture what a word means in a particular context. While every word token has its own vector, the vector can depend on a variable-length sequence of nearby words (i.e. the context). Consequently, a context vector is obtained by feeding a neural network with these context word vectors, and subsequently encoding them into a single fixed-length vector.

ULMFiT (Howard & Ruder, 2018) was the very first method to induce contextual word representations by harnessing the power of language modeling. The authors proposed to learn contextual embeddings by pre-training the language model, and then performing task-specific fine-tuning. The ULMFiT architecture is based on a vanilla 3-layer Long Short-Term Memory neural network without any attention mechanism. The authors proposed discriminative fine-tuning and slanted triangular learning rates for fine-tuning the model. Specifically, instead of using the same learning rate for all layers of the model, discriminative fine-tuning enables to tune each layer with different learning rates. Finally, a classifier fine-tuning on downstream task domain data is performed.

The other contextual embedding model introduced recently by Peters et al. (2018), is called ELMo (Embeddings from Language Models). Similarly to ULMFiT, this model uses tokens at the word level. However, ELMo additionally benefits from the subword-level information due to character convolutions. ELMo contextual embeddings are "deep" as they are a function of all hidden states. Concretely, context-sensitive features are extracted from a left-to-right and a right-to-left 2-layer bidirectional LSTM language models. Thus, the contextual representation of each word is the concatenation of the left-to-right and right-to-left representations as well as the initial embedding (see Fig. 1).

As the domain of vector representations of words is evolving rapidly, soon after ULMFiT and ELMo other architectures were introduced. Comparing to ULMFiT and ELMo, newer models such as e.g. OpenAI GPT (Radford et al., 2018) and BERT (Devlin et al., 2019) use more sophisticated neural network architectures that rely on self-attention. OpenAI GPT and BERT models operate at the subword-level, which is in our case not suitable, because we would not be able to compare such embeddings with other models inducing word vector representations (i.e. word2vec, GloVe).

In conclusion, we leverage the ELMo model to obtain contextual embeddings. More specifically, by means of ELMo we are able to feed our classifier model with context-aware

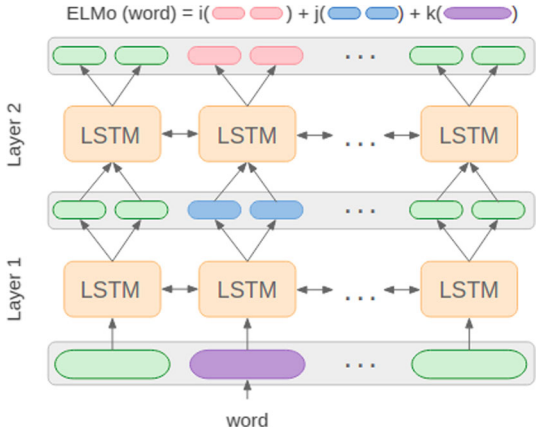


Fig. 1 The architecture of ELMo

embeddings of an input sequence. Hence, in this setting we do not perform any fine-tuning of ELMo on a downstream task.

2.4 Self-attention and bi-attention deep neural networks

Self-attention (or intra-attention) is an attention mechanism that computes a representation of a sequence by relating different positions of a single sequence. For the first time the attention mechanism was introduced by Bahdanau et al. (2014), and since then it has been applied successfully to different computer vision applications (e.g. Mnih et al., 2014; Stollenga et al., 2014), as well as NLP tasks (e.g. machine translation). The mechanism is often used as an extra source of information added on top of the CNN or LSTM model to enhance the extraction of sentence embedding (dos Santos et al., 2016; Lin et al., 2017). However, as noted by Lin et al. (2017), this scenario is not applicable to sentiment classification, since the model only receives a single sentence on input, hence there exist no such extra information. Nevertheless, some papers have appeared recently (e.g. Ambartsoumian and Popowich, 2018; Letarte et al., 2018).

3 Proposed approach

In this section, we present our own model, called Transformer-based Sentiment Analysis (TSA), which is depicted in Fig. 2. The model is based on the recently introduced transformer architecture (Vaswani et al., 2017). Unlike RNN or CNN based models, the transformer is able to learn dependencies between distant positions. Therefore, in this paper we show that attention-based models are suitable for other NLP tasks, such as learning distributed representations and sentiment analysis, and thus are able to improve the overall accuracy.

The architecture of the TSA model is hierarchical; steps to train it can be summarized as follows:

– at the very beginning there is a simple text pre-processing method that performs text clean-up and splits text into tokens;

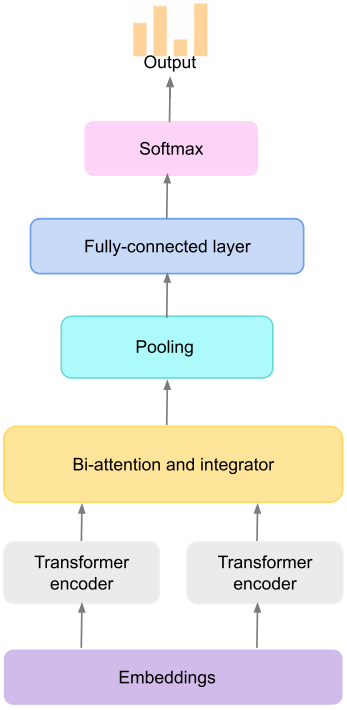


Fig. 2 An overview of the TSA model architecture

- we use contextual word representations to represent text as real-valued vectors;
- after embedding the text into real-valued vectors, the transformer network maps the input sequence into hidden states using self-attention. TSA is composed of two parallel encoders;
- next, a bi-attention mechanism is utilized to estimate the interdependency between representations;
- a single layer LSTM together with self-attentive pooling compute the pooled representations;
- a joint representation for the inputs is later passed to a fully-connected neural network;
- finally, a softmax layer is used to determine sentiment of the text.

In Fig. 3, we present a detailed visualization of the transformer encoder, which is a sole building block of each encoder in the proposed TSA model.

3.1 Embeddings and positional representations for the transformer encoder

Non-recurrent models, such as deep self-attention NN, do not necessarily process the input sequence in a sequential manner. Hence, there is no way they can record the position of each word in a sequence, which is an inherent limitation of every such model. Therefore, in the case of the transformer, the need has been addressed in the following manner: the transformer takes into account the order of the words in the input sequence by encoding their position information in extra vectors dubbed positional encodings (PE). There are many different approaches to embed position information, such as learned or fixed PEs, or recently

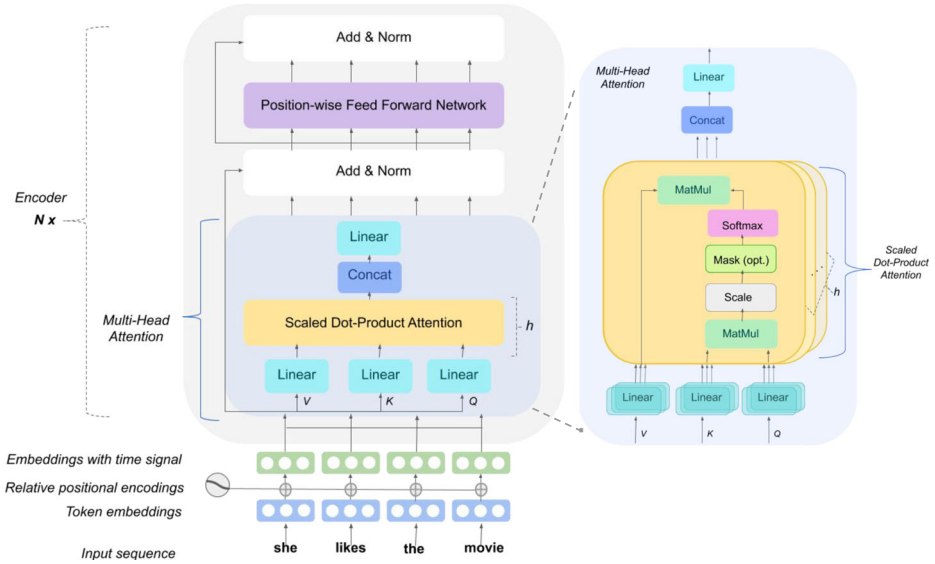


Fig. 3 The architecture of the transformer encoder

introduced relative position representations (RPR) (Shaw et al., 2018). Whereas (Vaswani et al., 2017) used sine and cosine functions of different frequencies to obtain embeddings with time signal, here we explore the effectiveness of applying a modified approach based on incorporating positional information into the model by using RPR instead of PE.

The input sequence, which is a combination of word embeddings and positional encodings providing time signal, is passed through N identical encoder layers. Each encoder layer has two sub-layers: multi-head self-attention (see Section 3.2) and a position-wise feed-forward neural network (FFN). The fully connected FFN consists of two linear layers with a ReLU (Nair, 2010) activation function between them:

$$FFN(x) = \max(0, xW_1 + b_1)\, W_2 + b_2 \tag{1}$$

where W_1, W_2, b_1, b_2 are learnable weights and biases respectively.

Around each sub-layer a residual connection is used as well as layer normalization is applied, as shown in Fig. 3.

3.2 Self-attention and multi-head attention

Multi-head attention allows each position in the encoder to access all positions in the previous layer of the encoder immediately, and in the first layer all positions in the input sequence. Multi-head attention employs h parallel self-attention layers (called heads), each with a triple of different query (Q), key (K) and value (V) matrices. Hence, three separate linear layers (as in Fig. 3) use their own weights to produce corresponding parameter matrices that are unique for each layer and attention head. In a nutshell, the attention mechanism in the transformer architecture relies on a scaled dot-product attention, which is a function of a query and a set of key-value pairs.

The computation of self-attention is performed in the following order. First, a multiplication of a query and transposed key is scaled by a factor of $1/\sqrt{d_z}$, as below:

$$m_{ij} = \frac{Q_i K_j^T}{\sqrt{d_z}} \tag{2}$$

Next, attention weight coefficient α_{ij} is produced using a softmax function over the scaled inner product:

$$\alpha_{ij} = \frac{e^{m_{ij}}}{\sum_{k=1}^{n} e^{m_{ik}}} \tag{3}$$

Finally, the weighted sum of value vectors is calculated as follows:

$$z_i = \sum_{j=1}^{n} \alpha_{ij} V_j \tag{4}$$

3.3 Bi-attention and pooling

The datasets used for training and evaluation of our models contain sequences of different length. However, in order to compute the sentiment score, they need to be of fixed size. Instead of trimming longer sentences or padding those that are shorter than the longest sentence in the dataset with trailing zeroes, we use masking and self-attentive pooling. Our approach is inspired by the BCN model proposed by McCann et al. (2017). Thanks to this mechanism, we are able to fit sequences with irregular sentence lengths into the final fixed-size vector.

The feature matrices X and Y produced by two parallel encoders, as shown in Fig. 2. X and Y are fed into the bi-attention module (see Fig. 4). Specifically, an affinity matrix of the encoder outputs $A = XY^\top$ is computed, on which we perform column-wise normalization to extract attention weights:

$$A^X = \text{softmax}(A) \text{ and } A^Y = \text{softmax}\left(A^\top\right) \tag{5}$$

Next, each representation is conditioned on the other through attention context summaries:

$$C^X = A^{X\top} X \text{ and } C^Y = A^{Y\top} Y \tag{6}$$

Once bi-attention obtains the conditional information, a concatenation of the following is performed: i) the original encoder outputs, ii) differences between the original representations and context summaries, iii) the Hadamard products (i.e. products of element-wise multiplication, denoted by the $\odot$ operator) of the original representations and context summaries. Next, we integrate this conditional information into text representations using two bi-LSTMs:

$$X^{|Y} = \text{bi-LSTM}\left(\left[X; X - C^Y; X \odot C^Y\right]\right) \tag{7}$$

$$Y^{|X} = \text{bi-LSTM}\left(\left[Y; Y - C^X; Y \odot C^X\right]\right) \tag{8}$$

Outputs of the bi-LSTMs are concatenated, and pooling is performed by means of max, min, mean, and self-attention. As proposed by McCann et al. (2017), we use the self-attentive pooling to compute weights and the weighted sum of each sequence. The concatenated pooled representations give a final representation, which is then passed through a fully-connected layer and softmax to provide a classification on output (as depicted at the top of Fig. 2).

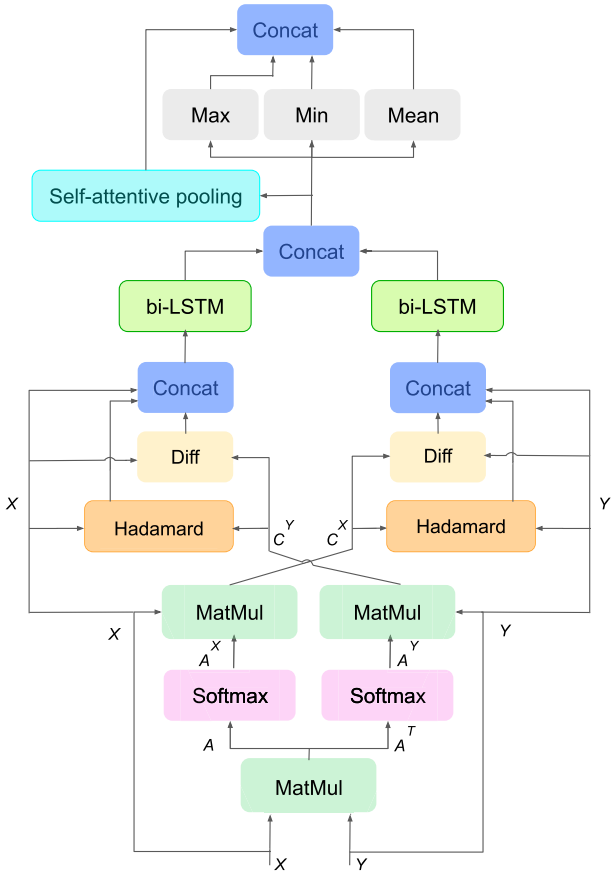


Fig. 4 Bi-attention and pooling

4 Experiments

4.1 Datasets

We evaluate the performance of our model on four annotated datasets containing information from various domains and covering three languages. The selected datasets are originally divided into training, dev and test sets (used to train, validate and evaluate models, respectively). In Table 1 we provide an overview of the benchmark corpora along with their main characteristics.

Stanford Sentiment Treebank (SST) This dataset (Socher et al., 2013) is a collection of movie reviews.[1] SST is annotated for two sentiment classifications – the binary one (SST-2) and fine-grained (SST-5). In SST-2 the reviews are divided into two groups: *positive* and *negative*, while in SST-5 one can distinguish 5 different review types: *very positive*,

[1] `rottentomatoes.com`

Table 1 An overview of the selected sentiment analysis datasets

Dataset	# Classes	Train	Dev	Test	Domain	Language
SST-2	2	6,920	872	1,821	movies	English
SST-5	5	8,544	1,101	2,210	movies	English
PolEmo 2.0-IN	3	5,783	723	722	medical, hotels	Polish
GermEval	3	19,432	2,369	2,566	travel, transport	German

positive, *neutral*, *negative*, *very negative*. The dataset consists of 11,855 single sentences and is widely used in the NLP community. As shown in Fig. 5, a similar distribution of reviews with respect to their length is preserved for all three dataset splits, with reviews between 13 and 26 tokens being the most frequent ones.

PolEmo 2.0 This dataset (Kocoń et al., 2019) comprises around 8,200 online reviews related to education, products, medicine and hotel domains. The vast majority of PolEmo 2.0 reviews (around 85%) come from the medicine and hotel domains. The authors of the dataset proposed different variants of the dataset to allow for in-domain and out-of-domain evaluation. Thus, we followed the approach (and the naming convention) of Rybak et al. (2020) and evaluated our models on the in-domain dataset (PolEmo 2.0-IN) comprising medicine-related and hotel reviews. Reviews covered in PolEmo 2.0 contain often more than one sentence. Human evaluators helped in constructing the PolEmo 2.0 dataset. They were instructed to choose from the following 6 sentiment labels while annotating the dataset: *strong positive (SP)*, *weak positive (WP)*, *neutral (0)*, *weak negative (WN)*, *strong negative (SN)*, and *ambiguous (AMB)*. As discrepancies between annotations existed – most of the mistakes happened for *(WP/WN/AMB)* tags, the authors decided to eliminate separate tags for *weakly positive* and *weakly negative* reviews, and merge those tags into one *(AMB)* tag. Furthermore, Kocoń et al. (2019) indicated that the majority of the errors were related to the identification of *neutral (0)* reviews. Hence, we decided to combine *(0)* and *(AMB)* tags together. As a result, we use three classes in this paper: *positive*, *negative*, and *neutral/ambiguous*. The majority of reviews in the PolEmo 2.0-IN dataset are below 250 tokens and the distribution for all three data splits is similar, as shown in Fig. 6.

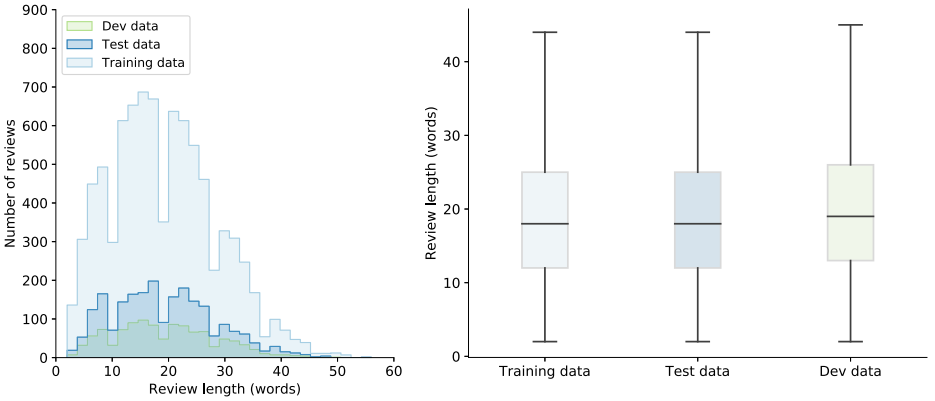


Fig. 5 Distribution of the SST dataset reviews for train/dev/test splits

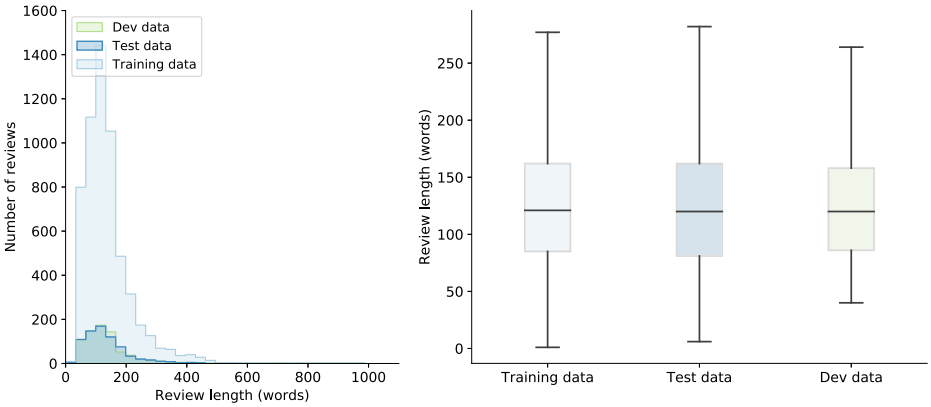


Fig. 6 Distribution of the PolEmo 2.0-IN dataset reviews for train/dev/test splits

GermEval This dataset (Wojatzki et al., 2017) contains customer reviews of the railway operator (Deutsche Bahn), published on social media and various web pages. Customers expressed their feedback regarding the service of the railway company (e.g. travel experience, timetables, etc.) by rating it as *positive*, *negative*, or *neutral*.

The GermEval dataset is the largest corpus used in our study with a similar distribution of reviews with respect to their length for all three data splits (see Fig. 7). Reviews in this dataset are often provided in the form of short posts published on Twitter (i.e. tweets), containing hashtags or twitter handles.

4.2 Experimental setup

We performed experiments using two models, namely ELMo+GloVe+BCN and ELMo+TSA. The first one, introduced by Peters et al. (2018), improved previous state-of-the-art results obtained by McCann et al. (2017), therefore we chose it to be our baseline. Thanks to this, we were able to reproduce experiments for SST datasets from Peters et al. (2018) and compare our model (ELMo+TSA) with other state-of-the-art approaches.

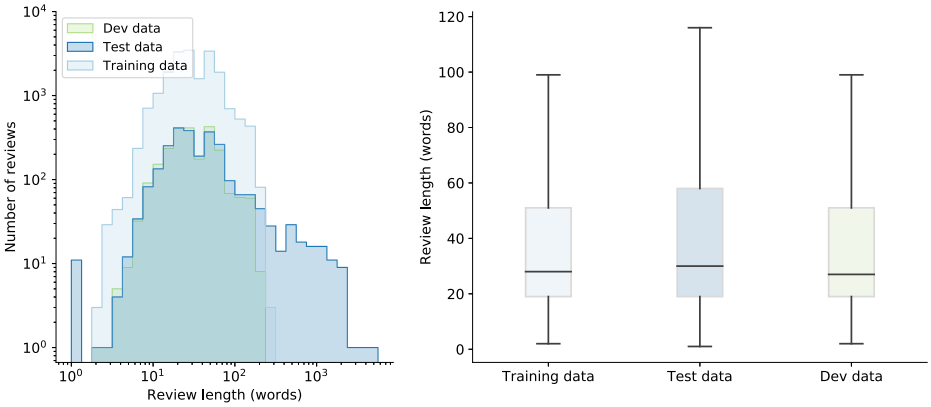


Fig. 7 Distribution of the GermEval dataset reviews for train/dev/test splits

Pre-processing of input datasets in the case of the baseline and our model is kept to a minimum, as we perform only tokenization when required. Furthermore, even though some datasets, such as SST or GermEval, provide additional information (i.e. phrase, word or aspect-level annotations), for each review we only extract text of the review and its corresponding rating.

Both models, TSA and the baseline, are implemented in the Python programming language, using `PyTorch`[2] and `AllenNLP`[3] frameworks. Concretely, the bi-attentive classification network (BCN), which is part of the TSA and the baseline, was adapted from McCann et al. (2017) using the `AllenNLP` library. Moreover, we use pre-trained word-embeddings, such as ELMo (Peters et al., 2018), GloVe (Pennington et al., 2014). In particular, we use the following ELMo models: Original,[4] Polish (Janz & Miłkowski, 2019) and German (May, 2019). In the ELMO+GloVe+BCN model we use the following 300-dimension GloVe embeddings: English,[5] Polish (Dadas, 2019) and German[6].

In order to simplify our approach when training the sentiment classifier model, we establish a very similar setting to the vanilla transformer. We use the same optimizer - Adam (Kingma & Ba, 2015) with its parameters set as follows: $\beta_1 = 0.9$, $\beta_2 = 0.98$, and $\epsilon = 10^{-9}$. We incorporate four types of regularization during training: dropout probability $P_{drop} = 0.1$, embedding dropout probability $P_{emb} = 0.5$, residual dropout probability $P_{res} = 0.2$, and attention dropout probability $P_{attn} = 0.1$. We use 2 encoder layers. In addition, we employ label smoothing of value $\epsilon_{ls} = 0.1$. In terms of the RPR parameters, we set clipping distance to $k = 10$.

To visualize the TSA's performance beyond quantitative metrics, we also performed qualitative analysis of the outputs. To this end, we employed the `LIME` algorithm[7] from the `ELI5` library.[8]

5 Results and analysis

In this section, we present quantitative and qualitative analysis of the results. For the former analysis, we measured the accuracy of our model (ELMo+TSA) and compared it with the baseline (ELMo+GloVe+BCN) as well as with corresponding models found in the literature. Moreover, to enrich our quantitative analysis, we provide confusion matrices for respective datasets as well as visualizations of the distribution of correctly and incorrectly predicted reviews with respect to their length. In order to analyze our results more holistically, we complement quantitative analysis with a qualitative evaluation. Specifically, the main goal behind the conducted qualitative analysis is to show how different models deal with complex linguistic phenomena such as negation, irony, or oxymorons, among others. We discuss the quality of predictions using representative review examples.

[2] https://pytorch.org
[3] https://allennlp.org
[4] https://allennlp.org/elmo
[5] http://nlp.stanford.edu/data/glove.840B.300d.zip
[6] https://wikipedia2vec.github.io/wikipedia2vec/pretrained
[7] https://github.com/marcotcr/lime
[8] https://github.com/eli5-org/eli5

Table 2 Results of TSA compared to the baseline and state-of-the-art systems evaluated on the English dataset (SST-2)

Work	Model	Accuracy [%]
Socher et al. (2013)	RNTN	85.4
Kalchbrenner et al. (2014)	DCNN	86.8
Kim (2014)	CNN	88.1
Tai et al. (2015)	Constituency Tree-LSTM	88.0
Kumar et al. (2016)	DMN	88.6
McCann et al. (2017)	CoVe+BCN	90.3
Ambartsoumian and Popowich (2018)	SSAN+RPR	84.2
Our baseline	ELMo+GloVe+BCN	**91.4**
Our model	ELMo+TSA	89.3

5.1 SST-2 dataset

In Table 2 we summarize the experimental results achieved for the SST-2 dataset by our model and other state-of-the-art systems reported in the literature. The best results were achieved by our baseline and the CoVe+BCN model proposed by McCann et al. (2017).

Similar to TSA, SSAN+RPR (Ambartsoumian & Popowich, 2018) also uses the transformer encoder for the classifier. As one can see in Table 2, TSA achieved better results than SSAN+RPR. One of the reasons why we achieve higher score for the SST-2 dataset might be that the authors of SSAN+RPR used word2vec embeddings (Mikolov et al., 2013), whereas we employ ELMo contextual embeddings (Peters et al., 2018). Moreover, in our TSA model, we use not only self-attention (as in SSAN+RPR) but also a bi-attention mechanism, which presumably also provides performance gains over the standard architectures.

SST-2 provides roughly the same number of *positive* and *negative* reviews. Hence, in principle, the ability of the model to learn to classify reviews is not hindered by non-equally represented sentiment classes for reviews. The distribution of reviews for each of two classes is slightly better balanced and closer to ground truth in the case of baseline predictions.

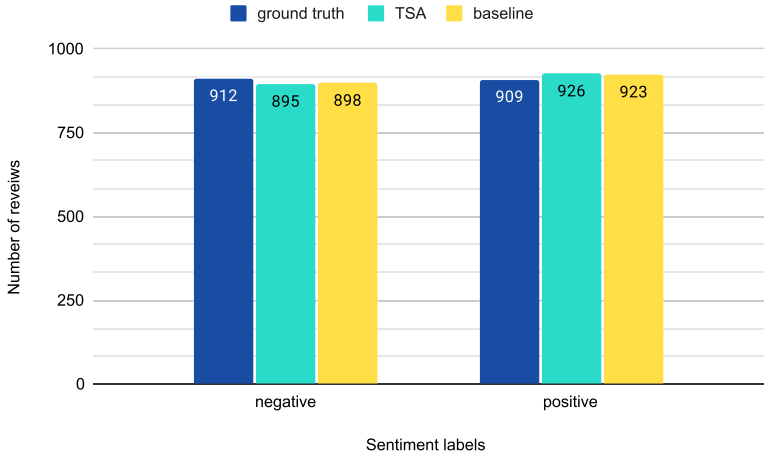
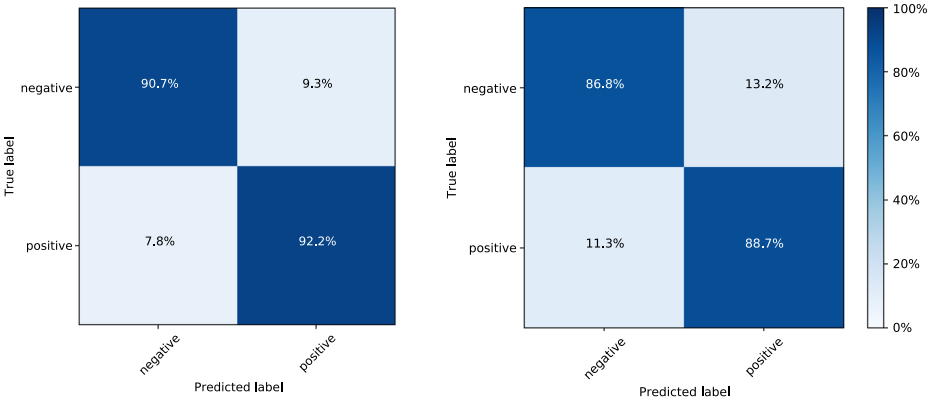


Fig. 8 Review polarity distribution for SST-2

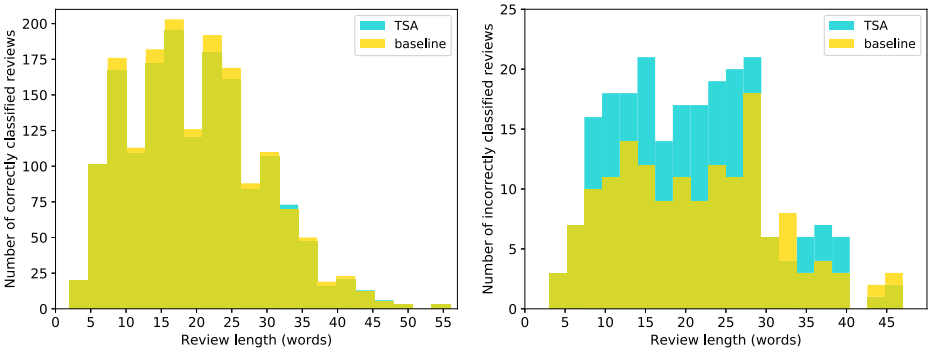


(a) baseline (b) TSA

Fig. 9 Confusion matrices for SST-2

We can observe (see Fig. 9) that the baseline model captures better true *negative* values (90.7%) than TSA (86.8%). Overall, TSA classified more reviews as *positive* than *negative* (as depicted by the comparison with ground truth in Fig. 8). As shown in Fig. 9, the number of misclassified *positive* labels is significantly greater for TSA (11.3%) than for the baseline model (7.8%). The misclassification of *negative* reviews was quite high for both models, 13.2% for TSA and 9.3% for the baseline, respectively.

In Fig. 10, we present the distribution of correctly and incorrectly predicted labels for the SST-2 reviews with respect to their length. The green area in the plot represents an overlap between TSA and the baseline. For instance, short reviews with less than 8 tokens were classified (and misclassified) exactly with the same accuracy by the two models. For mid-range reviews in terms of length, the baseline model showed its superiority. It is noteworthy that reviews having between 13 and 17 tokens occurred most frequently in the dataset, followed by reviews in the 21-26 token range (see Fig. 5). Our algorithm (TSA) performed slightly better for the longest reviews in the reviews (i.e. containing more than 40 tokens), as shown in Fig. 10a. Hence, we conclude, that TSA yielded very similar results or performed



(a) Correctly predicted (b) Incorrectly predicted

Fig. 10 Distribution of review predictions for SST-2 w.r.t. review length

Table 3 Example reviews with corresponding sentiment weights for SST-2

Model	Review	Predicted label	Ground truth label
baseline	A masterful film from a master filmmaker , unique in its deceptive grimness , compelling in its fatalist worldview .	positive	positive
TSA	A masterful film from a master filmmaker , unique in its deceptive grimness , compelling in its fatalist worldview .	positive	
baseline	The story loses its bite in a last-minute happy ending that 's even less plausible than the rest of the picture .	negative	negative
TSA	The story loses its bite in a last-minute happy ending that 's even less plausible than the rest of the picture .	negative	
baseline	In its ragged , cheap and unassuming way , the movie works .	negative	positive
TSA	In its ragged , cheap and unassuming way , the movie works .	negative	
baseline	Son of the Bride may be a good half-hour too long but comes replete with a flattering sense of mystery and quietness .	positive	negative
TSA	Son of the Bride may be a good half-hour too long but comes replete with a flattering sense of mystery and quietness .	positive	
baseline	Everytime you think Undercover Brother has run out of steam , it finds a new way to surprise and amuse .	positive	positive
TSA	Everytime you think Undercover Brother has run out of steam , it finds a new way to surprise and amuse .	positive	
baseline	Absorbing and disturbing -- perhaps more disturbing than originally intended -- but a little clarity would have gone a long way .	positive	negative
TSA	Absorbing and disturbing -- perhaps more disturbing than originally intended -- but a little clarity would have gone a long way .	negative	

better than the baseline for less represented review lengths, i.e. those reviews located at both ends of the length spectrum.

Table 3 shows the results of the qualitative analysis of the example SST-2 reviews. We observe that the TSA model performs quite similar to the baseline. The majority of reviews were assigned the same classification labels. For instance, in the first review, which both TSA and the baseline classified correctly, the former managed to better capture the sentiment polarity of particular words than the latter one. The third review was one of the most challenging, as it consists of a group of negative words, yet the overall meaning is positive. Both models misclassified the review, however, TSA correctly identified that the word "unassuming" carries a slightly positive meaning. Importantly, TSA was also able to properly classify the sixth review, while the baseline stumbled. This may account to the fact that TSA better dealt with a determiner before a noun, in this case "more disturbing", which is important for sentiment classification as the word "more" acts here as an opinion intensifier. Overall, the baseline model as well as TSA were quite accurate in determining the polarity of adjectives. However, they grappled more with nouns and verbs.

5.2 SST-5 dataset

The results for the SST-5 dataset are summarized in Table 4. The best results were achieved by the model proposed by McCann et al. (2017), which bears close resemblance in terms of the architecture to our baseline (which is the second best). Again, for the two models that use self-attention, namely TSA and SSAN+RPR (Ambartsoumian & Popowich, 2018), TSA achieves better results, similarly as for the SST-2 dataset.

A relatively low number of *very negative* reviews in the SST-5 dataset (see Fig. 11) can partially explain why our model was not able to learn to properly classify such reviews, and classified them as *negative* reviews instead. In the case of TSA, 73.1% of *very negative* reviews were labeled as *negative*, while just 16.5% of them were classified correctly, as shown in Fig. 12b. Similarly, but to a lesser degree, *positive* reviews were over-represented in TSA in comparison with *very positive* ones. Again, the ratio of *very positive* reviews labeled as *positive* was greater (47.1%) than a ratio of *very positive* reviews classified correctly (44.4%). Interestingly, *neutral* reviews were more often classified either as *negative* (45.5%) or *positive* (30.1%) than *neutral* (19.0%). We can observe that our model identified more accurately *negative* values (75.5%) than the baseline model (63.0%). In the case of true *positive* labels, both the baseline and TSA achieved similar performance, 66.3% and 63.7% respectively. All in all, the highest accuracy in predicting correct sentiment polarity was reached for *negative* (75.5%) followed by *positive* (63.7%) labels. Importantly, reviews with

Table 4 Results of TSA compared to the baseline and state-of-the-art systems evaluated on the English dataset (SST-5)

Work	Model	Accuracy [%]
Socher et al. (2013)	RNTN	45.7
Kalchbrenner et al. (2014)	DCNN	48.5
Kim (2014)	CNN	48.0
Tai et al. (2015)	Constituency Tree-LSTM	51.0
Kumar et al. (2016)	DMN	52.1
McCann et al. (2017)	CoVe+BCN	**53.7**
Ambartsoumian and Popowich (2018)	SSAN+RPR	48.1
Our baseline	ELMo+GloVe+BCN	53.5
Our model	ELMo+TSA	50.6

these two labels appear most frequently in the dataset. Hence, we may conclude, that the TSA model dealt better with less extreme reviews in terms of sentiment polarity. Whereas, the baseline model yielded more balanced results, in particular it classified *very negative* reviews more accurately (40.5%) than TSA.

Figure 13 demonstrates more accurate predictions obtained by the baseline than TSA when classifying reviews of various length. Frequent misses, especially in the case of TSA (as depicted in Fig. 13b), can be explained by a considerable difficulty in precise label prediction for a 5-class sentiment classification task.

However, such quantitative analysis presents the results of the models only from one angle. Table 5 contains a sample of five exemplary reviews selected for qualitative analysis. As such, it does not represent only the ratio of correctly and incorrectly classified reviews from the SST 5 dataset by each model, but provides a more holistic view on the performance of the models. Although qualitative analysis of sample reviews further confirms that TSA was slightly more prone to misclassify reviews contained in the SST-5 dataset, we conclude that the fine-grained scenario proves to be challenging for all models (see Table 4).

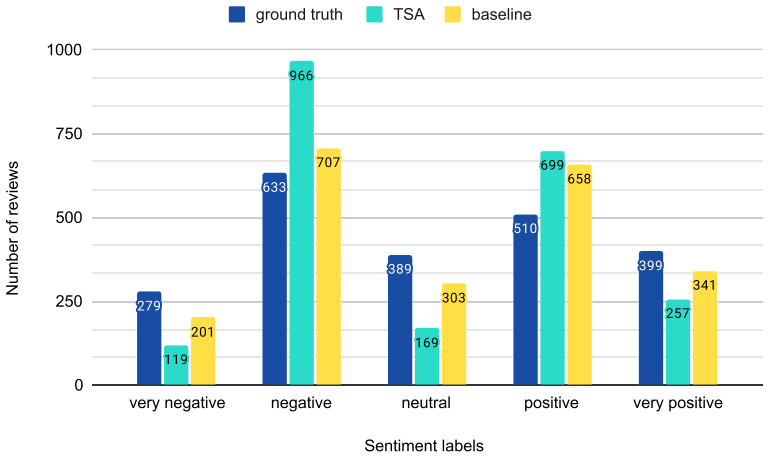
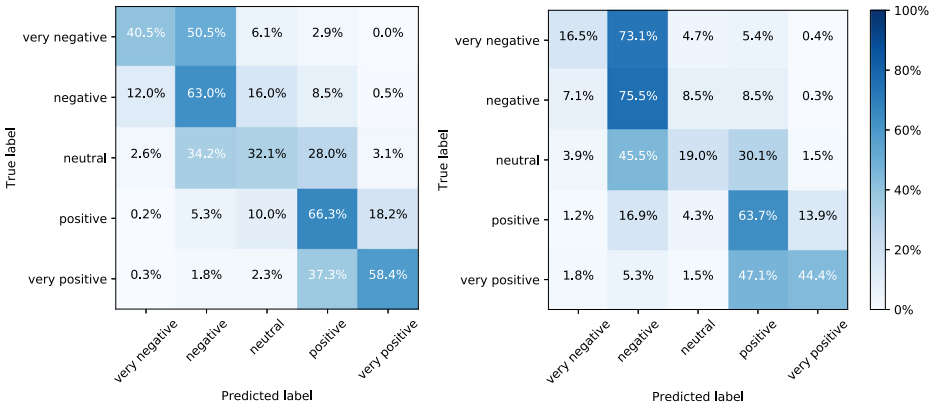


Fig. 11 Review polarity distribution for SST-5

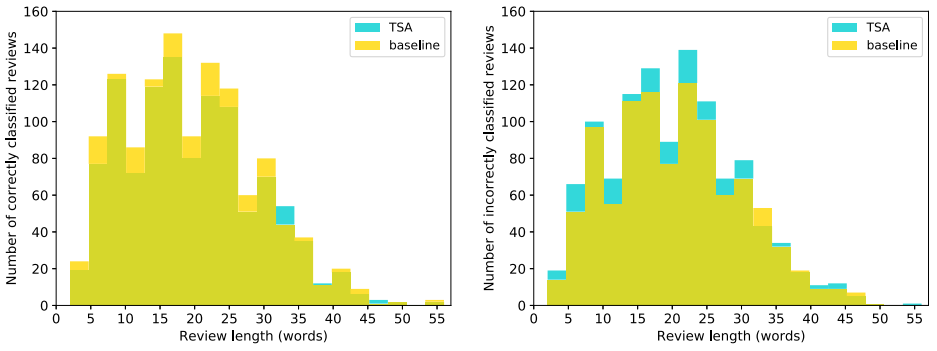


(a) baseline (b) TSA

Fig. 12 Confusion matrices for SST-5

One can observe here that TSA turned out to be more accurate when the degree of positivity or negativity varied due to complex linguistic phenomena, such as negation, irony, or oxymorons.

For instance, in one of the selected reviews, TSA was able to detect sarcasm (review 2), which remained undetectable for the baseline model. In a similar vein, baseline's wrong classification of a *very positive* review as a *very negative* one is another example of its difficulty in distinguishing literal from figurative meaning. In the first review, while TSA correctly detected positive sentiment in almost all words, it assigned a negative meaning to the quantitative pronoun "one" and this resulted in the misclassification of the entire review. In the same vein, the quantitative pronoun "some" in the second review was also considered by TSA a negative word. In reality, both these pronouns should be considered neutral. While the baseline model wrongly classified "one" as a positive word, it managed to treat the pronoun "some" as neutral. We hypothesize that such errors may be specific to the dataset and not necessarily represent systematic biases in the models. The second review is an interesting case, because the phrase "to listen to them reading the phone book"



(a) Correctly predicted (b) Incorrectly predicted

Fig. 13 Distribution of review predictions for SST-5 w.r.t. review length

Table 5 Example reviews with corresponding sentiment weights for SST-5

Model	Review	Predicted label	Ground truth label
baseline	One of the greatest family-oriented , fantasy-adventure movies ever .	very positive	very positive
TSA	One of the greatest family-oriented , fantasy-adventure movies ever .	neutral	
baseline	Some actors have so much charisma that you 'd be happy to listen to them reading the phone book .	positive	positive
TSA	Some actors have so much charisma that you 'd be happy to listen to them reading the phone book .	negative	
baseline	Run , do n't walk , to see this barbed and bracing comedy on the big screen .	very positive	very positive
TSA	Run , do n't walk , to see this barbed and bracing comedy on the big screen .	positive	
baseline	Intriguing and beautiful film , but those of you who read the book are likely to be disappointed .	negative	positive
TSA	Intriguing and beautiful film , but those of you who read the book are likely to be disappointed .	positive	
baseline	The smartest bonehead comedy of the summer .	very negative	very positive
TSA	The smartest bonehead comedy of the summer .	negative	

is a sarcastic expression with a negative meaning, although it contains only neutral words. The TSA model was able to detect mildly negative undertones, while the baseline model claimed it is a positive phrase. In fact, even though this review was assigned a *positive* ground truth label, it may be disputable if it is not a *neutral* or even a *negative* one in some contexts. The fourth review was misclassified by the baseline model, while TSA evaluated the review correctly. Noteworthy, the baseline considered (incorrectly) the linking word "but" to be very negative, increasing negative activation in the sentiment and reversing the positive polarity of the first clause and the whole review in result. While for TSA, the contrasting clause starting with "but" had a more nuanced polarity. Indeed, the baseline in general leaned towards classifying reviews with extreme labels (i.e. *very negative* or *very positive*).

5.3 PolEmo 2.0-IN dataset

In Table 6, we report experimental results for the PolEmo 2.0-IN dataset. Apart from TSA, another model that uses some variant of the transformer architecture is HerBERT (Rybak et al., 2020), based on BERT (introduced in Devlin et al., 2019), and optimized specifically for Polish. As one can see, the TSA model outperforms both, the baseline and the HerBERT model.

The PolEmo 2.0-IN dataset contains many more *negative* reviews than *positive* ones, as shown in Fig. 14. TSA achieved the best results in classifying *positive* reviews, correct labels were assigned with 91.9% accuracy (see Fig. 15). For *negative* reviews, correct label assignment was achieved in 89.3% cases. *Neutral/ambiguous* labels were more often misclassified, their sentiment polarity was more often confused with *negative* sentiment (9.8%) than the *positive* one (6.7%). However, due to the merge of *(AMB/WN/WP/0)* labels into the *neutral/ambiguous* group, the number of reviews with such a sentiment polarity is slightly

Table 6 Results of TSA compared to the baseline and state-of-the-art systems evaluated on the Polish dataset (PolEmo 2.0-IN)

Work	Model	Accuracy [%]
Rybak et al. (2020)	HerBERT	89.2
Our baseline	ELMo+GloVe+BCN	88.9
Our model	ELMo+TSA	**89.8**

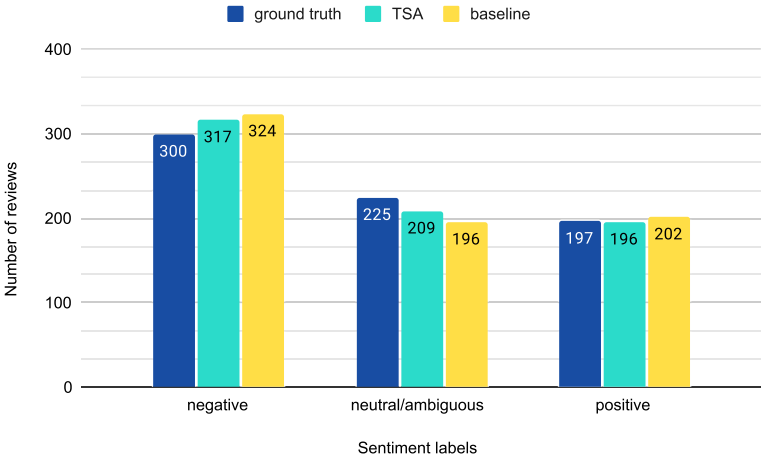
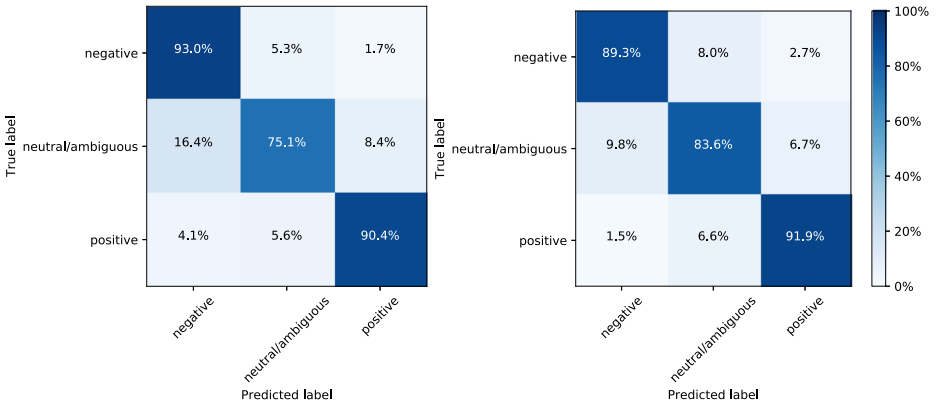


Fig. 14 Review polarity distribution for PolEmo 2.0-IN

higher than the number of *positive* reviews, and smaller than *negative* reviews, so we assume that the training dataset was well-balanced.

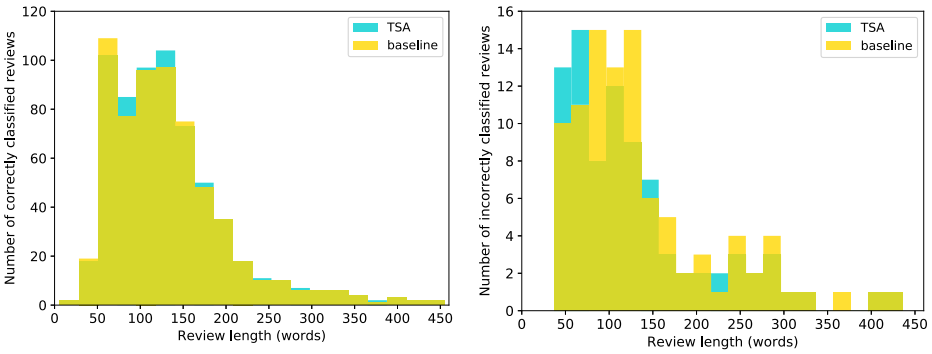
In terms of the correctness of predictions with respect to the review length (see Fig. 16), TSA again showed its slight superiority in classifying the long-tail reviews. In particular, for reviews over 240 tokens, the baseline performed at the similar level or worse than TSA. Also for mid-range review lengths, the baseline misclassified considerable number of reviews, while TSA yielded better results for reviews having 70 to 140 tokens. This is perhaps attributed to the fact that TSA is based on the transformer architecture, which is capable of learning dependencies between distant positions.

Our findings from the previous sections are also confirmed in the qualitative analysis of TSA's performance on the PolEmo 2.0-IN dataset. As one can see in Table 7, the baseline



(a) baseline (b) TSA

Fig. 15 Confusion matrices for PolEmo 2.0-IN



(a) Correctly predicted (b) Incorrectly predicted

Fig. 16 Distribution of review predictions for PolEmo 2.0-IN w.r.t. review length

and TSA models produce similar results. The first review was not classified correctly by both models; however, it is not that surprising given that the third sentence has a positive meaning and constitutes a substantial portion of the whole review. TSA was more successful at detecting negative phrases (e.g. "pomimo umówionych", "wciskając", "niezbyt miła", "próbuje na siłę") as well as positive ones (e.g. "przeprowadziła wszystkie badania", "dużym plusem"). This is in accord with the fact that TSA is better at capturing long-range dependencies. Both models correctly classified the second review. While the phrase "fantastyczny terapeuta" is positive, in this context it is used in a sarcastic way. The baseline and TSA didn't manage to capture that. However, we noticed that TSA was much more successful in detecting negative sentiment in the phrase "najgorzej wydane pieniądze". While the last review is classified as *neutral/ambigous* both models considered it *negative*. This review conveys hardly any positive sentiment, thus not surprisingly the baseline as well as TSA misclassified it. Both models correctly identified the word "niestety" to be negative. In

Table 7 Example reviews with corresponding sentiment weights for PolEmo 2.0-IN

Model	Review	Predicted label	Ground truth label
baseline	Lekarka pomimo umówionych godzin przyjmuje pacjentów bez rezerwacji wciskając ich pomiędzy osoby umówione . Jest niezbyt miła przy tym . Przeprowadziła wszystkie badania co jest dużym plusem (ale w sumie za to się jej płaci ciężkie pieniądze) . Tak jak ktoś wcześniej pisał próbuje na siłę sprzedać okulary ze swojego salonu .	neutral/ ambiguous	negative
TSA	Lekarka pomimo umówionych godzin przyjmuje pacjentów bez rezerwacji wciskając ich pomiędzy osoby umówione Jest niezbyt miła przy tym . Przeprowadziła wszystkie badania co jest dużym plusem (ale w sumie za to się jej płaci ciężkie pieniądze) . Tak jak ktoś wcześniej pisał próbuje na siłę sprzedać okulary ze swojego salonu .	neutral/ ambiguous	
baseline	Fantastyczny terapeuta do terapii małżeńskiej . 1 . Nie zawarła z nami żadnego kontraktu 2 . Wizyta odbyła się w obskurnym mieszkanku w wieżowcu 3 . Nie próbowała w żaden sposób nakierowywać na znalezienie problemów 4 . Sugerowała mojej żonie że mogę mieć zaburzenia psychiczne , które są przyczyną problemu czego nie potwierdziło badanie przez psychiatrę poleconego przez Panią terapeutkę 5 . Nie przyjmuje że ktoś może mieć inny punkt widzenia 6 . Stara się narzucić swoje rozwiązania , które jeszcze pogłębiły konflikt Były to najgorzej wydane pieniądze w moim życiu	negative	negative
TSA	Fantastyczny terapeuta do terapii małżeńskiej . 1 . Nie zawarła z nami żadnego kontraktu 2 . Wizyta odbyła się w obskurnym mieszkanku w wieżowcu 3 . Nie próbowała w żaden sposób nakierowywać na znalezienie problemów 4 . Sugerowała mojej żonie że mogę mieć zaburzenia psychiczne , które są przyczyną problemu czego nie potwierdziło badanie przez psychiatrę poleconego przez Panią terapeutkę 5 . Nie przyjmuje że ktoś może mieć inny punkt widzenia 6 . Stara się narzucić swoje rozwiązania , które jeszcze pogłębiły konflikt Były to najgorzej wydane pieniądze w moim życiu	negative	
baseline	Niestety póki co komentarz neutralny . Pomimo obietnicy przesłania diety trzy dni po wizycie , po 8 dniach zero diety , zero kontaktu , brak odpowiedzi na wiadomości , smsy , telefony . Jest mi bardzo przykro , bo pani Olimpia zrobiła na mnie dobre wrażenie i uważam , że za pozostawioną kwotę wypadało by choćby odezwać się i uspokoić , co do zamiaru wywiązania się z umowy .	negative	neutral/ ambiguous
TSA	Niestety póki co komentarz neutralny . Pomimo obietnicy przesłania diety trzy dni po wizycie , po 8 dniach zero diety , zero kontaktu , brak odpowiedzi na wiadomości , smsy , telefony . Jest mi bardzo przykro , bo pani Olimpia zrobiła na mnie dobre wrażenie i uważam , że za pozostawioną kwotę wypadało by choćby odezwać się i uspokoić , co do zamiaru wywiązania się z umowy .	negative	

Table 8 Results of TSA compared to the baseline and state-of-the-art systems evaluated on the German dataset (GermEval)

Work	Model	Accuracy [%]
Wojatzki et al. (2017)	SWN2-RNN	74.9
Our baseline	ELMo+GloVe+BCN	78.2
Our model	ELMo+TSA	**78.9**

the case of "neutralny" only TSA was correct to classify this as a non-negative word, however assigning a positive label was not correct either. Overall, both models performed in a similar manner, but importantly TSA managed to detect a positive sentiment in the phrase "dobre wrażenie".

5.4 GermEval dataset

The experimental results obtained for the GermEval dataset are summarized in Table 8. TSA achieved the best result, followed by the baseline, while the SWN2-RNN model, based on a traditional RNN architecture, demonstrated weaker results than the two.

The largest group in the GermEval dataset comprises *neutral* reviews, followed by *negative* ones (Fig. 17). Not surprisingly, TSA and the baseline achieved the best results in classifying *neutral* reviews, correct labels were assigned with 88.3% and 94.8% accuracy, respectively (see Fig. 18). Importantly, the number of reviews identified as *neutral* by the baseline, exceeded significantly the actual number of *neutral* reviews in the dataset (see Fig. 17). Hence, the ratio of misclassfied *neutral* reviews is also high for the baseline model: *neutral* labels were assigned in 77.1% of cases for *positive* reviews, and 55.1% for negative ones. The TSA model dealt better with *negative* reviews than the baseline, as it managed to classify them with 64.5% accuracy, as opposed to 42.4%. Furthermore, both models struggled with *positive* reviews, however, it is understandable as the number of *positive* reviews in the dataset was very small (only 105 reviews).

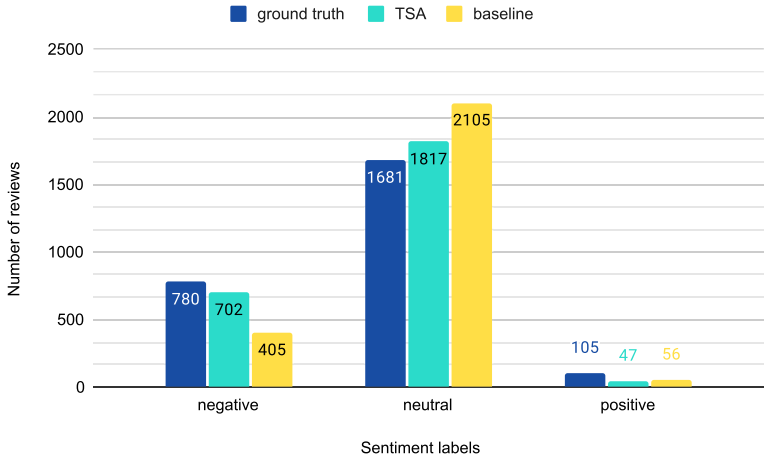
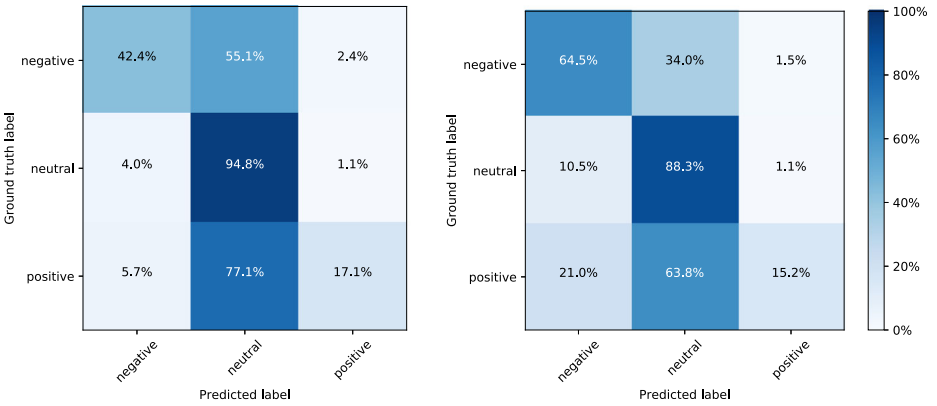


Fig. 17 Review polarity distribution for GermEval

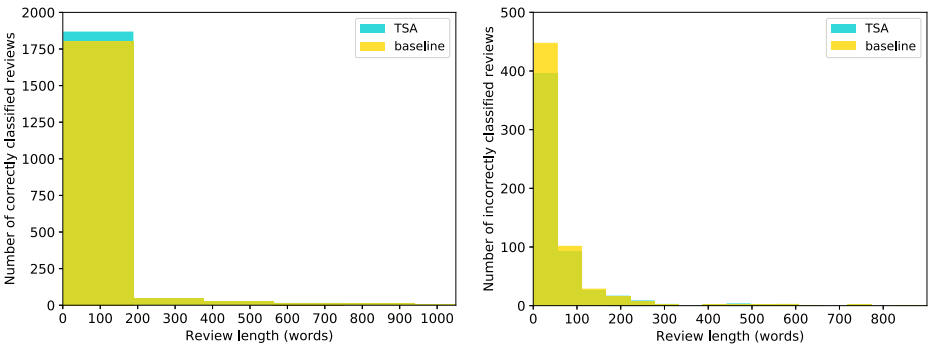


(a) baseline (b) TSA

Fig. 18 Confusion matrices for GermEval

As shown in Fig. 19, TSA was considerably better than the baseline at predicting reviews with less than 150 tokens. Importantly, reviews in this range constituted the majority of the GermEval dataset. For reviews between 200 and 500 tokens we observe slightly more incorrect predictions for TSA than the baseline model, however, due to a limited number of such reviews this interpretation may not be conclusive.

Table 9 presents a qualitative analysis performed on a sample of the GermEval reviews. Both models incorrectly predicted sentiment polarity of the first review. However, TSA seems to be slightly better at capturing actual positive words: “pünktlich” and “gute”. The baseline model on the other hand showed great variety in identifying positive words, being clearly not correct. For instance, it assigned almost equal weights to words “Nachtbusse” and “gute”. In the second review, in fact the only review from the selected ones where TSA predicted different label from the baseline, we can observe a similar pattern. The third review, although positive, contains only one adjective that clearly contributes to the sentiment of the sentence - the word “pünktlich”, whose polarity and weight were again better



(a) Correctly predicted (b) Incorrectly predicted

Fig. 19 Distribution of review predictions for GermEval w.r.t. review length

Table 9 Example reviews with corresponding sentiment weights for GermEval

Model	Review	Predicted label	Ground truth label
baseline	Die Nachtbusse sind gerade pünktlich zu ihrer ersten Runde gestartet . SWB Bus und Bahn wünscht eine gute Nacht .	positive	neutral
TSA	Die Nachtbusse sind gerade pünktlich zu ihrer ersten Runde gestartet . SWB Bus und Bahn wünscht eine gute Nacht .	positive	
baseline	Die Bahn ist so voll , dass es sich doppelt lohnt früher auszusteigen und zu laufen	neutral	negative
TSA	Die Bahn ist so voll , dass es sich doppelt lohnt früher auszusteigen und zu laufen	negative	
baseline	In der # Bahn mal wieder mehr geschafft als am restlichen Tag und dabei noch pünktlich ans Ziel . :-)	neutral	positive
TSA	In der # Bahn mal wieder mehr geschafft als am restlichen Tag und dabei noch pünktlich ans Ziel . :-)	neutral	
baseline	Der Hengst ist weg Aber die Bahn hat ausgeholfen ! Hmm ... Ich möchte nicht darüber reden	neutral	neutral
TSA	Der Hengst ist weg Aber die Bahn hat ausgeholfen ! Hmm ... Ich möchte nicht darüber reden	neutral	
baseline	# erfurt # bombendrohung Einfahrt in den Hbf per Bahn möglich , alles normal soweit	negative	negative
TSA	# erfurt # bombendrohung Einfahrt in den Hbf per Bahn möglich , alles normal soweit	negative	

identified by TSA. Furthermore, none of the two models interpreted non-alphanumeric characters at the end of the review as an emoticon carrying sentiment. The fourth review is a bit ironic, but both models managed to correctly predict its sentiment. Yet, it is difficult to explain why the baseline model interpreted "der Hengst" as strongly positive. Finally, the last review does not reveal much about its sentiment polarity, except the hashtags that precede the main sentence. Hashtags, popular in social media, were treated here as if they were normal sentiment-carrying words. Hence, the word indicating a bomb threat was assigned a very negative sentiment label. Interestingly, in both cases the word "normal" was classified as negative, even though its meaning is rather reassuring.

6 Conclusion

In this work, we presented TSA - a hierarchical, multi-layer sentiment classification model based on an architecture of the transformer encoder and a bi-attention mechanism. Hence, unlike many existing models, this work introduces an approach relying primarily on a self-attention mechanism and bi-attention. Our analysis shows that models leveraging contextual embeddings (i.e. TSA, the baseline, SSAN-RPR, CoVe+BCN) demonstrate remarkably better results than the rest of the reported models, which use traditional distributional word vectors (e.g. RNTN, CNN, SWN2-RNN). Moreover, the TSA model proved to be better at predicting sentiment labels for longer reviews than our baseline, which does not leverage self-attention. The ability to handle long-range dependencies by transformer is one of the key advantages of this architecture. We performed experiments for three languages and various domains using four benchmark datasets. Our method ELMo+TSA outperformed state-of-the-art for two languages (Polish and German). We show that our sentiment classifier achieves very good results, comparable to the state of the art, even though it is language-agnostic. Hence, this work is a step towards building a universal, multi-lingual model for sentiment classification. Furthermore, our method addressed the problem of context-dependent sentiment analysis. So far our model has been tested for three languages, each from different language family, including morphologically rich Polish and German. Yet, it is evident that evaluation of TSA using benchmarks also for other languages would be beneficial. It would be particularly interesting to analyze the behavior of our model with respect to low-resource languages, similarly to how we evaluated our approach for the Polish language. Finally, other promising research avenues worth exploring are related to unsupervised cross-lingual sentiment analysis.

Funding Open access funding provided by Warsaw University of Technology.

References

Ambartsoumian, A., & Popowich, F. (2018). Self-attention: A better building block for sentiment analysis neural network classifiers. In *Proc. of the 9th workshop on computational approaches to subjectivity, sentiment and social media analysis* (pp. 130–139). Association for Computational Linguistics, Brussels, Belgium, DOI 10.18653/v1/W18-6219.

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. In *Proc. of ICLR, San Diego, CA, USA*.

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, *35*(8), 1798–1828.

Biesialska, K., Biesialska, M., & Rybinski, H. (2020). Sentiment analysis with contextual embeddings and self-attention. In *Proc. of ISMIS 2020, LNCS, vol 12117. Graz, Springer, Austria, pp 32–41*. https://doi.org/10.1007/978-3-030-59491-6_4.

Chaturvedi, I., Ong, Y. S., Tsang, I. W. H., Welsch, R. E., & Cambria, E. (2016). Learning word dependencies in text by means of a deep recurrent belief network. *Knowledge-Based System*, *108*, 144–154.

Chen, X., Sun, Y., Athiwaratkun, B., Cardie, C., & Weinberger, K.Q. (2016). Adversarial deep averaging networks for cross-lingual sentiment classification. *Transactions of the Association for Computational Linguistics*, *6*, 557–570.

Dadas, S. (2019). A repository of Polish NLP resources. Github. https://github.com/sdadas/polish-nlp-resources/, accessed: 2020-01-20.

Das, S., & Chen, M. (2001). Yahoo! for Amazon: Extracting market sentiment from stock message boards. In *Proc. of the Asia Pacific Finance Ass. Ann. Conf. (APFA) Bangkok, Thailand*, (Vol. 35 p. 43).

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL-HLT, Association for Computational Linguistics, Minneapolis, Minnesota,* pp 4171–4186, https://doi.org/10.18653/v1/N19-1423, https://aclanthology.org/N19-1423.

Firth, J. R. (1957). A synopsis of linguistic theory, 1930-1955. *Studies in Linguistic Analysis*.

Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. In *Proc. of ACL, Association for Computational Linguistics, Melbourne, Australia,* pp 328–339, https://doi.org/10.18653/v1/P18-1031, https://aclanthology.org/P18-1031.

Janz, A., & Miłkowski, P. (2019). ELMo embeddings for Polish. http://hdl.handle.net/11321/690, CLARIN-PL digital repository.

Kalchbrenner, N., Grefenstette, E., & Blunsom, P. (2014). A convolutional neural network for modelling sentences. In *Proc. of ACL, Association for Computational Linguistics, Baltimore, Maryland,* pp 655–665, https://doi.org/10.3115/v1/P14-1062.

Kennedy, A., & Inkpen, D. (2006). Sentiment classification of movie reviews using contextual valence shifters. *Comput. Intell.*, *22*, 110–125.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proc. of EMNLP, Association for Computational Linguistics, Doha, Qatar,* pp 1746–1751, https://doi.org/10.3115/v1/D14-1181, https://aclanthology.org/D14-1181.

Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic gradient descent. In *Proc. of ICLR, San Diego, CA, USA*.

Kocoń, J., Miłkowski, P., & Zaśko-Zielińska, M. (2019). Multi-level sentiment analysis of PolEmo 2.0: Extended corpus of multi-domain consumer reviews. In *Proc. of CoNLL, Association for Computational Linguistics, Hong Kong, China,* pp 980–991, https://doi.org/10.18653/v1/K19-1092, https://aclanthology.org/K19-1092.

Kumar, A., Irsoy, O., Ondruska, P., Iyyer, M., Bradbury, J., Gulrajani, I., Zhong, V., Paulus, R., & Socher, R. (2016). Ask me anything: Dynamic memory networks for natural language processing. In *Proc. of ICML, PMLR, New York, New York, USA, vol 48, p 1378–1387*.

Letarte, G., Paradis, F., Giguère, P., & Laviolette, F. (2018). Importance of self-attention for sentiment analysis. In *Proc. of the EMNLP Workshop BlackboxNLP, Association for Computational Linguistics, Brussels, Belgium,* https://doi.org/10.18653/v1/W18-5429.

Lin, Z., Feng, M., dos Santos, C. N., Yu, M., Xiang, B., Zhou, B., & Bengio, Y. (2017). A structured self-attentive sentence embedding. In *Proc. of ICLR, Toulon, France*.

Liu, B. (2012). *Sentiment analysis and opinion mining. Synthesis lectures on human language technologies*. San Rafael: Morgan & Claypool Publishers.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proc. of ACL, Association for Computational Linguistics, Portland, Oregon, USA,* pp 142–150, https://aclanthology.org/P11-1015.

May, P. (2019). German ELMo model. https://github.com/t-systems-on-site-services-gmbh/german-elmo-model, accessed: 2020-01-20.

McCann, B., Bradbury, J., Xiong, C., & Socher, R. (2017). Learned in translation: Contextualized word vectors. In *Proc. of NeurIPS*, (Vol. 30 pp. 6294–6305).

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proc. of NeurIPS*, (Vol. 26 pp. 3111–3119).

Mnih, V., Heess, N. M. O., Graves, A., & Kavukcuoglu, K. (2014). Recurrent models of visual attention. In *Proc. of NeurIPS*.

Mohammad, S. M. (2016). Sentiment analysis: Detecting valence, emotions, and other affectual states from text. In *Emotion measurement, Elsevier* (pp. 201–237).

Nair, V. (2010). Rectified linear units improve restricted boltzmann machines. In *Proc. of ICML, Omnipress, Haifa, Israel* (pp. 807–814).

Pang, B., & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proc. of ACL, Barcelona, Spain,* pp 271–278, https://doi.org/10.3115/10.3115/1218955.1218990, https://aclanthology.org/P04-1035.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques. In *Proc. of EMNLP, Association for Computational Linguistics,* pp 79–86, https://doi.org/10.3115/1118693.1118704, https://aclanthology.org/W02-1011.

Paulus, R., Socher, R., & Manning, C.D. (2014). Global belief recursive neural networks. In *Proc. of NeurIPS*, (Vol. 27 pp. 2888–2896).

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In *Proc. of EMNLP, Association for Computational Linguistics, Doha, Qatar,* pp 1532–1543, https://doi.org/10.3115/v1/D14-1162, https://aclanthology.org/D14-1162.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proc. of NAACL-HLT, Association for Computational Linguistics, New Orleans, Louisiana,* pp 2227–2237, https://doi.org/10.18653/v1/N18-1202, https://aclanthology.org/N18-1202.

Potts, C. (2011). Sentiment analysis tutorial. In *Sentiment Analysis Symp., Nov. 8-9, 2011*.

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving Language Understanding by Generative Pre-Training. Preprint. https://www.cs.ubc.ca/amuham01/LING530/papers/radford2018improving.pdf.

Rybak, P., Mroczkowski, R., Tracz, J., & Gawlik, I. (2020). KLEJ: Comprehensive benchmark for Polish language understanding. In *Proc. of ACL, Association for Computational Linguistics, Online,* pp 1191–1201, https://doi.org/10.18653/v1/2020.acl-main.111.

dos Santos, C. N., Tan, M., Xiang, B., & Zhou, B. (2016). Attentive pooling networks. arXiv:1602.03609.

Shaw, P., Uszkoreit, J., & Vaswani, A. (2018). Self-attention with relative position representations. In *Proc. of NAACL-HLT, Association for Computational Linguistics, New Orleans, Louisiana,* pp 464–468, https://doi.org/10.18653/v1/N18-2074.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proc. of EMNLP, Association for Computational Linguistics, Seattle, Washington, USA,* pp 1631–1642, https://aclanthology.org/D13-1170.

Stollenga, M. F., Masci, J., Gomez, F. J., & Schmidhuber, J. (2014). Deep networks with internal selective attention through feedback connections. In *Proc. of NeurIPS* (pp. 3545–3553).

Tai, K. S., Socher, R., & Manning, C.D. (2015). Improved semantic representations from tree-structured long short-term memory networks. In *Proc. of ACL-IJCNLP, Association for Computational Linguistics, Beijing, China,* pp 1556–1566, https://doi.org/10.3115/v1/P15-1150.

Turney, P. (2002). Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proc. of ACL, Association for Computational Linguistics, Philadelphia, Pennsylvania, USA,* pp 417–424, https://doi.org/10.3115/1073083.1073153.

Turney, P. D., & Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. *J Artif Intell Res*, *37*, 141–188.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In *Proc. of NeurIPS* (pp. 5998–6008).

Wadawadagi, R. S., & Pagi, V. (2020). Sentiment analysis with deep neural networks: comparative study and performance assessment. *Artificial Intelligence Review*, *53*(8), 6155–6195.

Wojatzki, M., Ruppert, E., Holschneider, S., Zesch, T., & Biemann, C. (2017). GermEval 2017: Shared task on aspect-based sentiment in social media customer feedback. In *Proc. of the GermEval* (pp. 1–12).

Zhang, X., & LeCun, Y. (2015). Text understanding from scratch. arXiv:150201710.