



# Neuro-Symbolic Models for Sentiment Analysis

Jan Kocoń<sup>(✉)</sup>, Joanna Baran, Marcin Gruza, Arkadiusz Janz, Michał Kajstura, Przemysław Kazienko, Wojciech Korczyński, Piotr Miłkowski, Maciej Piasecki, and Joanna Szolomicka

Department of Artificial Intelligence, Wrocław University  
of Science and Technology, Wrocław, Poland  
`jan.kocoon@pwr.edu.pl`

**Abstract.** We propose and test multiple neuro-symbolic methods for sentiment analysis. They combine deep neural networks – transformers and recurrent neural networks – with external knowledge bases. We show that for simple models, adding information from knowledge bases significantly improves the quality of sentiment prediction in most cases. For medium-sized sets, we obtain significant improvements over state-of-the-art transformer-based models using our proposed methods: Tailored KEPLER and Token Extension. We show that the cases with the improvement belong to the hard-to-learn set.

**Keywords:** Neuro-symbolic sentiment analysis · plWordNet · Knowledge base · Transformers · KEPLER · HerBERT · BiLSTM · PolEmo 2.0

## 1 Introduction

Sentiment analysis is an NLP task performed in industrial or marketing solutions. It aims to determine how customers (authors of textual opinions) react to given products or services. In the classical *symbolic approach*, a text is evaluated using external knowledge bases, e.g., sentiment dictionaries [3, 4]. Then, words from the text are linked to positive, negative, or neutral polarization derived from such dictionaries. The final sentiment is an aggregation over all words. State-of-the-art sentiment analysis methods are mainly based on transformers. Such language models contain millions of parameters but also require large computational resources. Hence, their simplified methods, e.g., BiLSTM [15, 19], are often used in practice. We refer to both of these approaches as our *baselines*.

This work was funded by the National Science Centre, Poland, project no. 2021/41/B/ST6/04471 (PK) and 2019/33/B/HS2/02814 (MP); the Polish Ministry of Education and Science, CLARIN-PL; the European Regional Development Fund as a part of the 2014–2020 Smart Growth Operational Programme, CLARIN – Common Language Resources and Technology Infrastructure, project number POIR.04.02.00-00C002/19; the statutory funds of the Department of Artificial Intelligence, Wrocław University of Science and Technology.

In this paper, we present and validate neuro-symbolic solutions to sentiment analysis that combine both approaches: deep neural networks and symbolic inference. These methods use vector representations of text from deep language models and external knowledge bases in the form of, e.g., lexicons (sentiment), knowledge graphs (WordNet), and lexico-syntactic patterns (sentiment modification rules). Our main contributions are: (1) design or adaptation of multiple neuro-symbolic methods, (2) comparing our approaches against methods without knowledge bases; (3) prove that for simpler models, the knowledge base significantly improves the prediction quality; (4) showing specific cases of medium-sized sets for which knowledge base information significantly improves the prediction quality for the current transformer-based SOTA models; (5) evidence that neuro-symbolic approaches improve reasoning mainly for hard-to-learn cases.

## 2 Related Work

Sentiment analysis (SA) is a standard classification task aiming to decide whether the presented text has a positive, negative or neutral polarity. Some works treat SA as a multi-class prediction problem when data are focused on ranking system (e.g. 5-star). In the past, standard machine learning methods were applied to SA such as decision tree, SVM, Naive Bayes or random forest. However, in recent years we are observing the growing popularity of deep-learning (DL) models which proved to be very successful.

**Standard Deep-Learning Approach.** Different types of DL architectures were exploited in sentiment classification. We can mention here CNN, LSTM, RNN, GRU, Bi-LSTM and their variations with attention mechanism [11]. Most of them were trained in a supervised setting. However, despite the promising results achieved by these models, vulnerabilities have been observed such as poor knowledge propagation of cross-domain sentiment analysis in online systems [2], mainly due to lack of enough manual annotated datasets for all domains.

**Neuro-Symbolic Approach.** Many lexicon resources for various languages have been developed. Princeton WordNet (PWN) is a major one for English but similar knowledge bases were created for other languages too. Some contain emotive annotations for specific word meanings assigned by people (e.g. SentiWordNet). In addition, NLP tools were created to analyse data in a manner similar to human understanding (POS – part-of-speech tagger, WSD – word sense disambiguation). Given the complexity of the SA task, which combines natural language processing, psychology, and cognitive science, using such external knowledge processed according to human logic could improve results of standard DL methods. Moreover, it can imply more explainable predictions. Some works have been done in that field. [8] incorporated graph-based ontology ConceptNet into sentiment analysis enriching the text semantics. Apart from knowledge graph, [25] added a WSD process into social media posts processing. A context-aware sentiment attention mechanism acquiring the sentiment polarity of each word with its POS tag from SentiWordNet was studied in [13]. The pre-training

process very rarely respects sentiment-related knowledge. If so, the problem of proper representation of sentiment words and aspect-sentiment pairs needs to be solved. To address it Sentiment Knowledge Enhanced Pre-training (SKEP) [24] has been proposed. It uses sentiment masking and constructs three sentiment knowledge prediction objectives to embed this information at the word- and aspect-level into a pre-trained representation.

### 3 Datasets

#### 3.1 plWordNet and plWordNet Emo

plWordNet is a very large lexico-semantic network for Polish constructed on the basis of the corpus-based wordnet development method, according to which *lexical units*<sup>1</sup> (henceforth LUs) are the basic building blocks of the wordnet [7]. LUs of very similar meaning are grouped into synsets (sets of synonyms) – each LU belongs to only one synset. The most recent version describes  $\approx 295\text{k}$  LUs for  $\approx 194\text{k}$  lemma of four PoS (part of speech) grouped into  $\approx 228\text{k}$  synsets<sup>2</sup> [1].

Emotive annotation was performed on the level of LUs and LU use examples [27]. Context-independent emotive characterisation of an LU was obtained by comparing its authentic use in text corpora. The main distinction is between *neutrality* vs *polarity* of LUs. Polarised LUs are assigned the *intensity* of the sentiment polarisation, *basic emotions* and *fundamental human values*. The latter two help to determine the sentiment polarity and its intensity expressed in the 5 grade scale: *strong* or *weak* vs *negative* and *positive*, plus the ambiguous tag. Annotator decisions are supported by text examples that must be included in the annotations. Due to the compatibility with other wordnet-based annotations, eight basic emotions<sup>3</sup> recognised by Plutchik [20] were used. One LU can be assigned more than one emotion and, as a result, complex emotions are represented by using the same eight-element set. The 12 fundamental human values<sup>4</sup> postulated by Puzynina [21] link the emotive state of the speaker to the *evaluative attitude*. The annotations were done by two annotators each (a linguist and a psychologist) according to the 2+1 scheme.

#### 3.2 PolEmo

PolEmo 2.0 dataset [12,15] is a sentiment analysis task benchmark dataset. It consists of more than 8,000 consumer reviews, containing more than 57,000 sentences. Texts come from four domains: hotels, medicine, products, and school. Each review was annotated with sentiment in a 2+1 scheme at the text level and the sentence level. In this work, only text level examples were used. There are the following sentiment classes: positive, negative, neutral, and ambivalent.

<sup>1</sup> Triples: lemma, Part of Speech (PoS) and sense identifier.

<sup>2</sup> <http://plwordnet.pwr.edu.pl>.

<sup>3</sup> *Joy, fear, surprise, sadness, disgust, anger, trust and anticipation.*

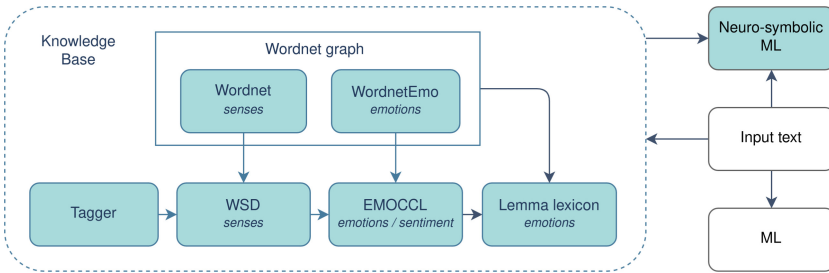
<sup>4</sup> *Utility, truth, knowledge, beauty, happiness, futility, harm, ignorance, error, ugliness.*

The obtained Positive Specific Agreement (PSA) [9] was 90% at the text level and 87% at the sentence level. PolEmo 2.0<sup>5</sup> is available under an MIT open license.

### 3.3 Preprocessing

All texts from PolEmo were tokenized, lemmatized, and tagged using CLARIN PoS tagger<sup>6</sup>. Word sense disambiguation [10] (WSD<sup>7</sup>) was performed to identify the appropriate LU belonging to that token. Next plWordNet Emo was used to annotate words with sentiment, basic emotions and fundamental human values (*valuations*). Additionally, we also propagated sentiment and emotion annotations from wordnet to words that originally did not have this annotation in the plWordNet Emo. It required training a regressor based on fastText model [6] using emotive dimensions from plWordNet Emo aggregated per lemma (emotions propagated). Data annotation statistics are presented in Table 1.

The example pipeline for combining text with a knowledge base is shown in Fig. 1. It tokenizes text and matches words with their correct meanings in Wordnet. Furthermore, information on sentiment and emotions from Wordnet annotations (WordnetEmo) is added to the text at the *word sense* level using the EMOCCCL tool. The emotional Wordnet annotation is aggregated at the *word lemmas level* and added to the text (lemma lexicon).



**Fig. 1.** Baseline approach (ML) vs. neuro-symbolic approach (neuro-symbolic ML). The blue colour on the diagram indicates neuro-symbolic part of the method. (Color figure online)

<sup>5</sup> <https://clarin-pl.eu/dspace/handle/11321/710>.

<sup>6</sup> <http://ws.clarin-pl.eu/tager>.

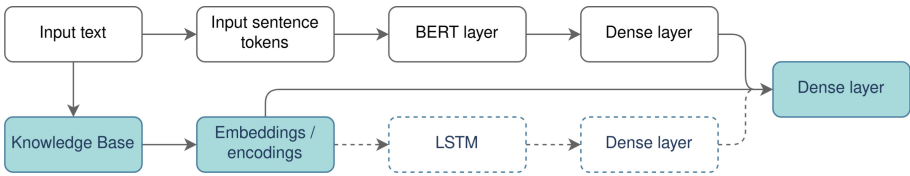
<sup>7</sup> <http://ws.clarin-pl.eu/wsd>.

**Table 1.** Token annotation coverage in preprocessed PolEmo2.0

Feature	Train	Dev	Test
Sentiment (all)	28.3%	28.7%	28.4%
Sentiment (pos/neg)	9.2%	9.4%	9.4%
Basic emotions	8.3%	8.5%	8.5%
Valuations	8.6%	8.8%	8.7%
Emotion propagated	99.9%	99.9%	99.9%

## 4 Neuro-Symbolic Models

### 4.1 HB: HurtBERT Model

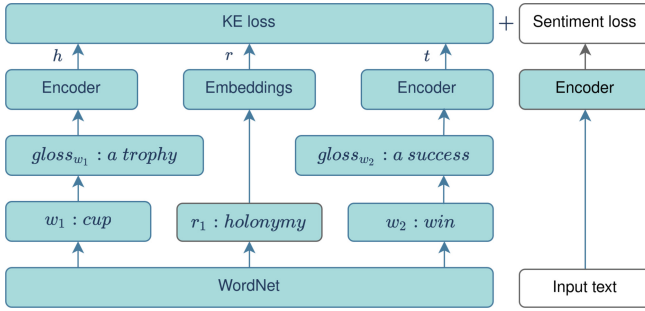


**Fig. 2.** HB: HurtBERT model.

HurtBERT [16] (Fig. 2) was proposed for the abusive language detection task. Apart from the standard transformer-based text representation, it incorporates knowledge from a lexicon [5]. Additional features are processed by a separate branch and then are concatenated with a text representation before the classification layer. Lexical information can be utilized in two ways: (1) HB-enc: HB-encoding using a simple frequency count for the lexicon categories; (2) HB-emb: HB-embedding obtained with a LSTM network. The second method is more expressive, as it takes token order into account. As the number of categories in plWordNet differs from the ones used in the original paper, we modified the dimensionality of sentiment embedding layer accordingly.

### 4.2 TK: Tailored KEPLER Model

Tailored KEPLER model (Fig. 3) is an adaptation of KEPLER [26] which incorporates information from a knowledge graph (KG) into a pretrained language model (PLM) like BERT during fine-tuning. It is different to the original KEPLER model where extra KE knowledge is used during pretraining stage (unsupervised masked language modeling). Our Kepler approach is tailored to single task, it utilizes extra knowledge during fine-tuning. To harness knowledge from KG, its entities representation is obtained by encoding their text



**Fig. 3.** Tailored KEPLER model. The same encoder is used to obtain embeddings for KE loss and for the downstream task.

descriptions with PLM. Thus, PLM can be additionally learned with Knowledge Embedding (KE) objective along with a task objective.

We used pWordNet as KG from which we extract relations between LUs and between synsets. The relation facts are described by a triplet  $(h, r, t)$  where  $h$ ,  $t$  represent the head and the tail entities;  $r$  is a relation type from set  $\mathcal{R}$ . After discarding some types of relations (e.g., hyperonymy is symmetric to hyponymy), 48 types of relations remained.

We get the embeddings for heads and tails by encoding the corresponding LUs descriptions with PLM. The relation types are encoded by a randomly initialized, learnable embedding table. As KE loss, the loss from [22] is used. It adopts negative sampling [18] and tries to minimize TransE distance for the entities being in the relation and to maximize it for negative samples triplets.

To fine-tune the pretrained model, we applied multitask loss  $\mathcal{L} = \mathcal{L}_{KE} + \mathcal{L}_{NLP}$  where  $\mathcal{L}_{NLP}$  is loss for a downstream NLP task. We used only those triplets which LUs are present in the downstream task training set and we clipped the number of steps in each epoch to the size of the downstream task training set.

### 4.3 TE: Token Extension Model

The benefits of additional knowledge bases are best seen in simple language models [17]. For this reason, fastText model for Polish language [14] and BiLSTM model [15] working on the basis of embeddings per token derived from it has been taken into consideration (Fig. 4). This approach allows to use the knowledge base at the level of each token. Thus, we propose 3 variants: (1) baseline - which uses token embedding only, (2) TE-original – where additional knowledge (as a vector) from the wordnet is concatenated to the token embedding, and (3) TE-propagated – using propagated data (Sect. 3.3) on all words in text.

### 4.4 ST(P): Special Tokens (with Positioning) Model

In transformer with **S**pecial **T**okens (ST) model (Fig. 5) we added special BERT tokens corresponding to emotions and sentiments. They are put after a word

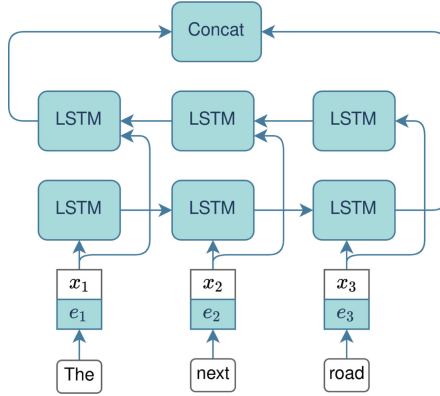


Fig. 4. TE: Token Extension model.

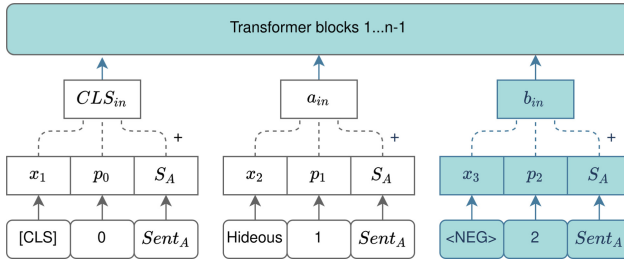


Fig. 5. ST: Special Tokens model.

which lemma is annotated with emotion or sentiment in plWordNet. It is a way to harness emotive knowledge from plWordNet to Transformer. Exemplary input can be in a form of: *She was still weeping [SAD], despite the happy [JOY] end of the movie.* Since emotion tokens are marked as special tokens, they will not be broken down into word pieces by tokenizer and their embedding vectors will be initialized randomly. Since adding new tokens to the text breaks its sequentiality, we test additional version of the model (STP: Special Tokens with Positioning) in which we adjust the emotion token position indexes so that they are equal to the lemma token position indexes they correspond to (e.g. *Happy*<sub>idx=1</sub> [JOY]<sub>idx=1</sub> and *amazed*<sub>idx=3</sub> [SURPRISED]<sub>idx=3</sub> *girl*<sub>idx=4</sub>.) With this adjustment, the emotional tokens will have the same positional embeddings as their corresponding lemmas.

#### 4.5 STN: Special Tokens from Numeric Data Model

STN method is an extension of ST method (same model as in Fig. 5) designed for the cases when a lemma is annotated by many annotators. Lemma intensity of emotion  $e$  can be expressed as fraction  $\alpha_e \in (0, 1)$  of annotation with emotion  $e$ . Since not all LUs are annotated, a regression model is used to propagate

these values to other lemmas. A special token for emotion  $e$  is put after a word if its  $\alpha_e > T$ . In another variant, we add all found special tokens (without replacement) in a text at its end. Threshold  $T$  can be either the same for all emotions or individual value  $T_e$  assigned to each emotion  $e$  as a quantile of all  $\alpha_e$  values for lemmas in the train set. For STN method, the special token embeddings for each emotion are initialized with an average of the embeddings of all subword tokens obtained after tokenization of the emotion name. Adjusting positional embedding proposed for ST method is not applied for STN.

### 4.6 SE: Sentiment Embeddings Model

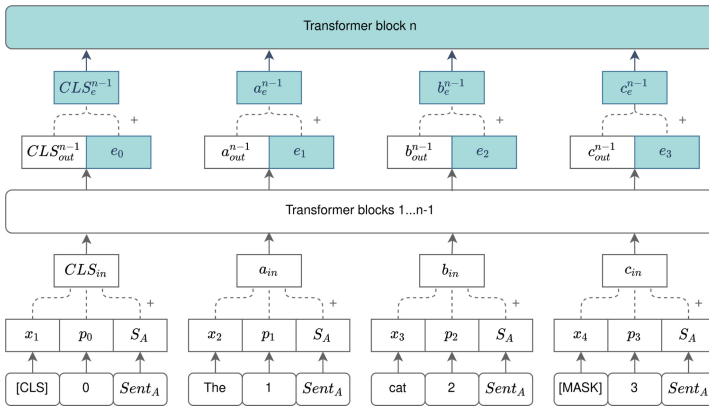


Fig. 6. SE: Sentiment Embeddings model.

Both HurtBERT-embedding and HurtBERT-encoding aggregate additional information at text level, which can limit the interaction between the text and features obtained from plWordNet. To incorporate token-level lexical annotations into a transformer, we add trainable sentiment embeddings as hidden representations before the last transformer layer (Fig. 6). If the word consists of multiple BPE parts, we add the embedding to all subword tokens. Augmented representations are then passed to a classifier to compute the probability of each sentiment class. The classifier consists of a dense layer followed by a softmax activation function. During the pretraining phase of HerBERT, there is no additional lexical information. Adding the sentiment token in the second to last layer of the transformer could corrupt the token representations. We consider two variants: (1) SE: the last transformer block’s weights are left unchanged and (2) SE-reset: the last transformer block’s weights are randomly initialized. Random reinitialization of the last BERT layer is a common practice [28] and can make it easier for the model to utilize additional features.



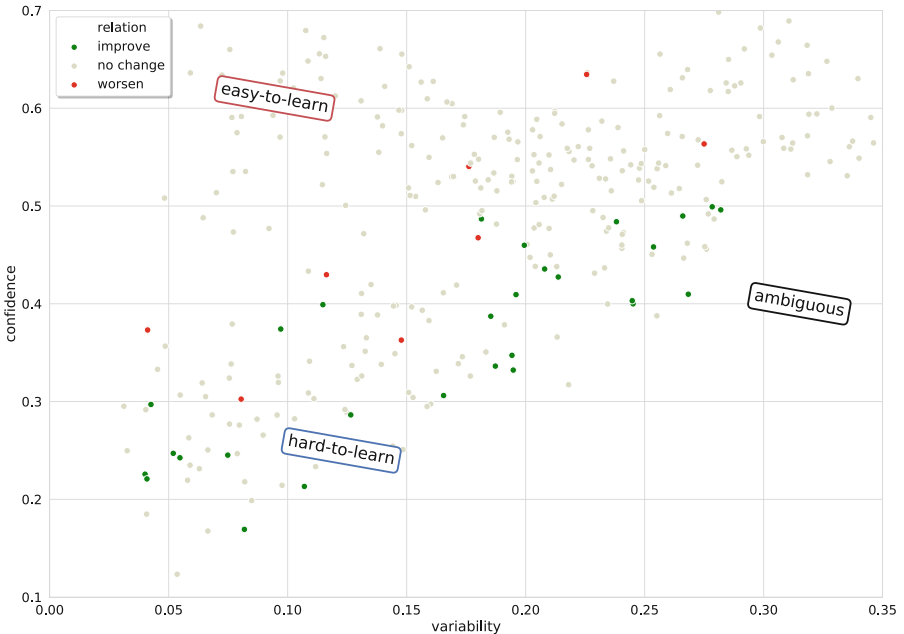
## 5 Experimental Setup and Results

For each experimental setup, we compare a baseline neural model with its neuro-symbolic extension. In each method (excluding TE), we used HerBERT as a SOTA baseline for sentiment analysis performed on PolEmo 2.0 dataset. We test the method on selected undersampled training datasets of different sizes. Both baseline and neuro-symbolic models are trained using the same hyperparameters. Some of the methods are adapted from other papers, so the baselines are not identical in different setups in terms of hyperparameters. For each configuration, the experiments are repeated 10 times.

### 5.1 TK: Tailored KEPLER Model

Fine-tuning is performed for 4 epochs with learning rate  $5e-5$ , batch size 4 and weight decay 0.01. Maximum sequence length is 256 and 32 for *PolEmo texts* and for *entities text* representations, respectively. Results are presented in Fig. 8. The statistical gains are obtained for the smaller training sets what shows that the extra knowledge from KG helps when an amount of data is limited.

For the case where the difference between the baseline and TK was significant, both models were compared using the cartography method [23]. It uses



**Fig. 7.** Datamap [23] for the baseline model in Tailored Kepler (TK) setup. Green colour indicates cases for which correctness  $c_K$  of TK has a higher value than correctness  $c_H$  for the baseline. Gray examples have  $|c_K - c_H| \leq 0.3$  (small or no change). Red instances mean that  $c_K < c_H$ . (Color figure online)

model confidence, variability, and correctness over epochs to find which texts are hard, easy or ambiguous to learn. The correctness specifies the fraction of times the true label is predicted. The confidence is the mean probability of the ground truth between epochs. The variability measures how indecisive the model is. Figure 7 shows datamap for HerBERT. Colours of the points on the diagram indicate if the instance is easier to learn for Tailored KEPLER than the baseline (HerBERT). The diagram shows that adding extra knowledge improves correctness for far more cases than it worsens. Moreover, the examples, which are affected belong to hard-to-learn and ambiguous classes only.

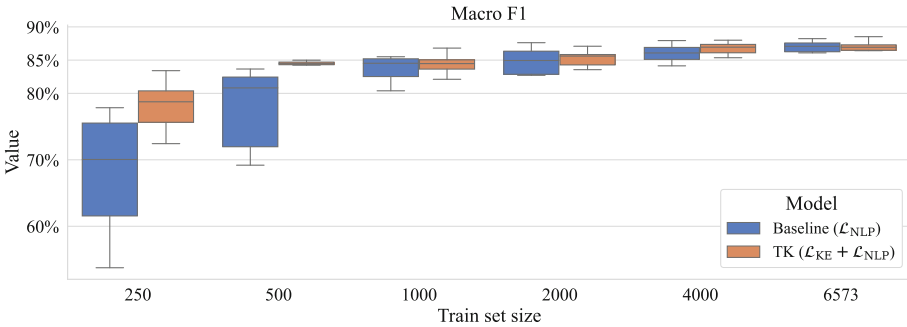


Fig. 8. Results for Tailored KEPLER model.

### 5.2 TE: Token Extension Model

The models were trained for 25 epochs. The model performing best on the validation set was used for testing (maximum F1-macro). The results of the experiments are presented in Fig. 9. The performance of models based on fastText embeddings increases with the size of the training set. On 5 of the 6 dataset sizes tested, the approach using additional data in the original (TE-orig) or propagated form (TE-prop) was better than the baseline. For train sizes over 1,000, using propagated data proved to be the best possible approach.

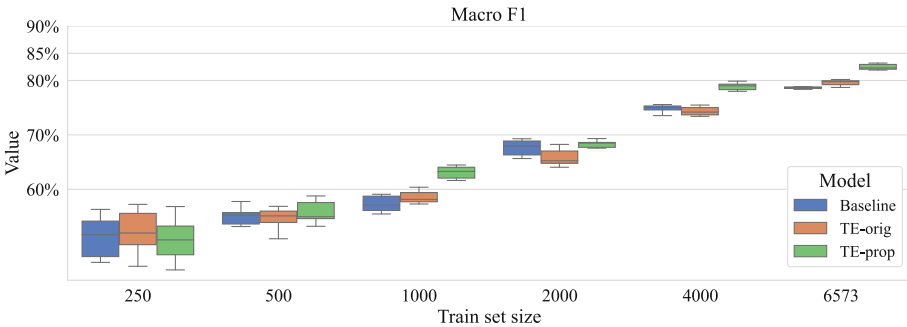
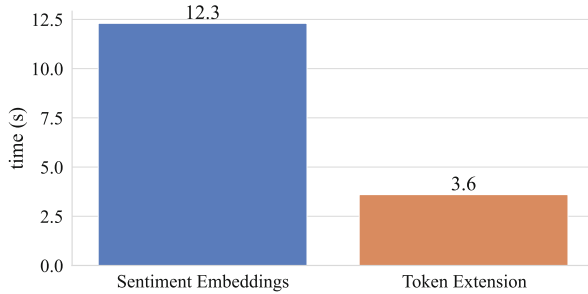


Fig. 9. Results for Token Extension model.

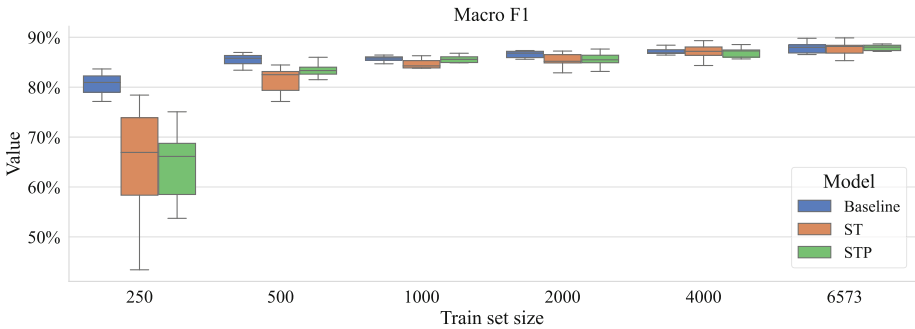
It is important to compare the running time of the TE model with that of the example transformer-based (SE) model in Fig. 10. The performance of the TE model (macro F1: 83%) is significantly worse by about 4 p.p. relative to the SE model (macro F1: 87%). However, the inference time of the TE model for the test set (3.6 s) is almost four times shorter than that of the SE model. <https://www.overleaf.com/project/620b8ae3cda06ae4691ba512>.



**Fig. 10.** Inference time of Sentiment Embeddings (transformer-based) and Token Extension (BiLSTM+fastText) neuro-symbolic models.

### 5.3 ST(P): Special Tokens (with Positioning) Model

The maximum tokenizer text length has been set to 512, so that adding new emotional tokens does not require to truncate the text. The batch size was set to 20. We used the Adam optimizer with the learning rate set to  $2e-5$  during training. The models were trained for 5 epochs and the model with the smallest validation loss was checkpointed and tested. The results are presented in Fig. 11.



**Fig. 11.** Special Token Model (ST) and ST with positional embeddings (STP).

ST and STP models achieve worse results than the baseline for smaller train datasets (250 and 500 samples). For bigger train datasets, there are no significant differences between the models.

### 5.4 STN: Special Tokens from Numeric Data

We consider the following variant with *in-text* and *at-end* special tokens: (1) no propagated data,  $T = 0.5$ ; (2) propagated data,  $T = 0.6$ ; (3) propagated data, individual threshold  $T_e$  equal to 0.75 quantile. In each case, fine-tuning is performed for 4 epochs with learning rate  $5e-5$ , batch size 16, weight decay 0.01, and maximum sequence length 512. Results are presented in Fig. 12.

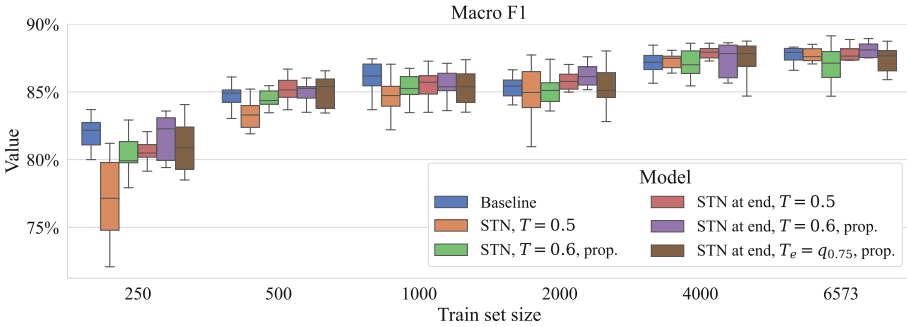


Fig. 12. Results for Special Tokens for Numeric Data model.

The results do not show significant improvements for each STN method. In the case of *in-text* special tokens, the results are usually worse. For *at-end-of-text* special tokens performance is very similar to the baseline.

### 5.5 HB: HurtBERT Model and SE: Sentiment Embeddings Model

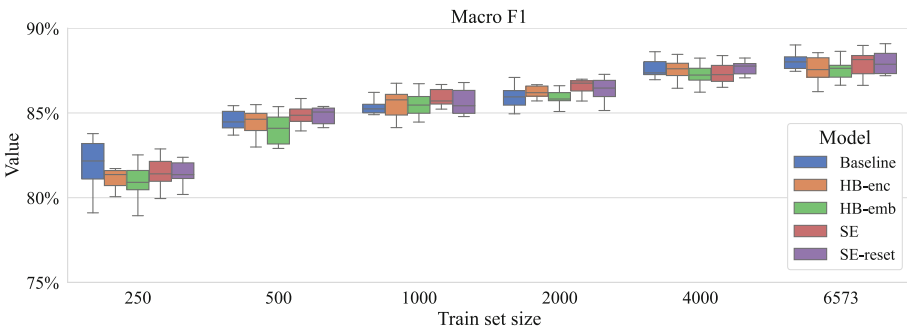


Fig. 13. Results for HurtBERT-encoding, HurtBERT-embeddings, Sentiment Embedding and Sentiment Embedding Reset models.

Models are fine-tuned using AdamW optimizer with learning rate  $1e-5$ , linear warmup schedule, batch size 32, and maximum sequence length 256 for 30 epochs

and the best model is chosen according to a validation F-score. Results are presented in Fig. 13. In lower data regimes (250 and 500 samples), there may not be enough data to learn embeddings of sentiment features, hence the similar performance. For larger datasets, the additional information from a knowledge base is outweighed by a textual information. Our experiments do not show a significant improvement over a baseline, both for HurtBERT and the proposed SE method. Texts in PolEemo dataset are complex and aggregating additional lexical features on the level of a whole text is not sufficient.

## 6 Conclusions

We designed and adapted multiple neuro-symbolic methods. The additional knowledge in most transformer-based neuro-symbolic models does not lead to improvement in most cases. For the smallest variants of datasets (training dataset: 250 texts), it can even make the training process more unstable and degrade the model quality (ST\*, HB\*, SE\*). Adding special tokens inside the text is not beneficial for pretrained BERT models because it damages the natural structure of the text. It is not the case for tokens added at the end of the text, but still no performance gain is observed. It can be caused by the fact that the considered PolEemo dataset has high PSA, so the knowledge encompassed in the pretrained HerBERT model is sufficient to obtain very good results.

However, for small and medium-sized datasets, our Tailored KEPLER neuro-symbolic transformer-based model produced statistically significant gains. It also allowed to obtain better and more stable results. Analysis of these cases shows performance gains for examples belonging to *ambivalent* sentiment class. We examined in which cases additional knowledge improved the quality of inference, Fig. 7. The vast majority of these cases were identified by the baseline model as hard-to-learn.

A key finding of the study is that the knowledge base significantly improves the quality of simple models such as Token Extension, Fig. 10. Compared to transformer-based models, we obtain an almost fourfold reduction in inference time, at the cost of a significant but relatively small decrease in quality (4 pp.). For the TK model, the quality gain due to additional knowledge was significant for most cases. This shows that with very little computational cost, the inference quality can be significantly improved for such models.

## References

1. p1WordNet 4.5 (2021). <http://hdl.handle.net/11321/834>. CLARIN-PL
2. Al-Moslmi, T., Omar, N., Abdullah, S., Albared, M.A.: Approaches to cross-domain sentiment analysis: systematic lit. Review. *IEEE Access* **5**, 16173–16192 (2017)
3. Augustyniak, L., Kajdanowicz, T., Kazienko, P., Kulisiewicz, M., Tuligłowicz, W.: An approach to sentiment analysis of movie reviews: lexicon based vs. classification. In: Polycarpou, M., de Carvalho, A.C.P.L.F., Pan, J.-S., Woźniak, M., Quintian, H., Corchado, E. (eds.) HAIS 2014. LNCS (LNAI), vol. 8480, pp. 168–178. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-07617-1\\_15](https://doi.org/10.1007/978-3-319-07617-1_15)

4. Augustyniak, L., et al.: Simpler is better? Lexicon-based ensemble sentiment classification beats supervised methods. In: ASONAM 2014, pp. 924–929 (2014)
5. Bassignana, E., Basile, V., Patti, V.: Hurllex: a multilingual lexicon of words to hurt. In: CLiC-it 2018, vol. 2253, pp. 1–6. CEUR-WS (2018)
6. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information (2017)
7. Dziob, A., Piasecki, M., Rudnicka, E.: plWordNet 4.1 - a linguistically motivated, corpus-based bilingual resource. In: The 10th Global Wordnet Conference, pp. 353–362. Global Wordnet Association, July 2019
8. Ghosal, D., Hazarika, D., Roy, A., Majumder, N., Mihalcea, R., Poria, S.: Kingdom: knowledge-guided domain adaptation for sentiment analysis. [arXiv:2005.00791](https://arxiv.org/abs/2005.00791) (2020)
9. Hripscak, G., Rothschild, A.: Agreement, the f-measure, and reliability in information retrieval. *J. Am. ER. Med. Inform. Ass. (JAMIA)* **12**(3), 296–298 (2005)
10. Janz, A., Piasecki, M.: A weakly supervised word sense disambiguation for polish using rich lexical resources. *Poznan Stud. Cont. Ling.* **55**(2), 339–365 (2019)
11. Joseph, J., Vineetha, S., Sobhana, N.: A survey on deep learning based sentiment analysis. *Mater. Today Proc.* **58**, 456–460 (2022)
12. Kanclerz, K., Miłkowski, P., Kocoń, J.: Cross-lingual deep neural transfer learning in sentiment analysis. *Procedia Comput. Sci.* **176**, 128–137 (2020)
13. Ke, P., Ji, H., Liu, S., Zhu, X., Huang, M.: SentiLARE: sentiment-aware language representation learning with linguistic knowledge. [arXiv:1911.02493](https://arxiv.org/abs/1911.02493) (2020)
14. Kocoń, J., Gawor, M.: Evaluating KGR10 Polish word embeddings in the recognition of temporal expressions using BiLSTM-CRF. *Schedae Informaticae* **27** (2018)
15. Kocoń, J., Miłkowski, P., Zaśko-Zielińska, M.: Multi-level sentiment analysis of PolEmo 2.0: extended corpus of multi-domain consumer reviews. In: CoNLL2019, pp. 980–991. ACL, November 2019
16. Koufakou, A., Pamungkas, E.W., Basile, V., Patti, V.: HurtBERT: incorporating lexical features with BERT for the detection of abusive language. In: The 4th Workshop on Online Abuse and Harms, pp. 34–43. ACL, November 2020
17. Ma, Y., Peng, H., Cambria, E.: Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM. In: AAAI 2018, vol. 32 (2018)
18. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS 2013, pp. 3111–3119 (2013)
19. Kocoń, J., Miłkowski, P., Kanclerz, K.: MultiEmo: multilingual, multilevel, multidomain sentiment analysis corpus of consumer reviews. In: Paszynski, M., Kranzlmüller, D., Krzhizhanovskaya, V.V., Dongarra, J.J., Sliot, P.M.A. (eds.) ICCS 2021. LNCS, vol. 12743, pp. 297–312. Springer, Cham (2021). [https://doi.org/10.1007/978-3-030-77964-1\\_24](https://doi.org/10.1007/978-3-030-77964-1_24)
20. Plutchik, R.: EMOTION: A Psychoevolutionary Synthesis. Harper & Row (1980)
21. Puzynina, J.: Język wartości [The language of values]. Polish Scientific Publishers PWN (1992)
22. Sun, Z., Deng, Z.H., Nie, J.Y., Tang, J.: RotatE: knowledge graph embedding by relational rotation in complex space. In: The International Conference on Learning Representations (ICLR) (2019)
23. Swayandipta, S., et al.: Dataset cartography: mapping and diagnosing datasets with training dynamics. In: EMNLP 2020, pp. 9275–9293. ACL, November 2020
24. Tian, H., et al.: SKEP: sentiment knowledge enhanced pre-training for sentiment analysis (2020)

25. Vizcarra, J., Kozaki, K., Torres Ruiz, M., Quintero, R.: Knowledge-based sentiment analysis and visualization on social networks. *NGC* **39**(1), 199–229 (2021)
26. Wang, X., Gao, T., Zhu, Z., Liu, Z., Li, J.Z., Tang, J.: KEPLER: a unified model for knowledge embedding and pre-trained language representation. *Trans. Assoc. Comput. Linguist.* **9**, 176–194 (2021)
27. Zaśko-Zielińska, M., Piasecki, M.: Towards emotive annotation in plWordNet 4.0. In: *The 9th Global Wordnet Conference*, pp. 153–162. Global WordNet Association (2018)
28. Zhang, T., Wu, F., Katiyar, A., Weinberger, K.Q., Artzi, Y.: Revisiting few-sample BERT fine-tuning. [arXiv:2006.05987](https://arxiv.org/abs/2006.05987) (2020)