Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference "Dialogue 2023"

June 14–16, 2023

Named Entity-Oriented Sentiment Analysis with text2text Generation Approach

Ivan Moloshnikov
Moloshnikov_IA@nrcki.ru

Maxim Skorokhodov
Skorokhodov_MV@nrcki.ru

Aleksandr Naumov
Naumov-AV@nrcki.ru

NRC "Kurchatov Institute"
Moscow, Russia

Roman Rybka
Rybka_RB@nrcki.ru
NRC "Kurchatov Institute"
Russian Technological University "MIREA"
Moscow, Russia

Alexander Sboev
Sboev_AG@nrcki.ru
National Research Nuclear University "MEPhI"
NRC "Kurchatov Institute"
Moscow, Russia

Abstract

This paper describes methods for sentiment analysis targeted toward named entities in Russian news texts. These methods are proposed as a solution for the Dialogue Evaluation 2023 competition in the RuSentNE shared task. This article presents two types of neural network models for multi-class classification. The first model is a recurrent neural network model with an attention mechanism and word vector representation extracted from language models. The second model is a neural network model for text2text generation. High accuracy is demonstrated by the generative model fine-tuned on the competition dataset and CABSAR open dataset. The proposed solution achieves 59.33 over two sentiment classes and 68.71 for three-class classification by f1-macro.

Keywords: multi-class classification, sentiment analysis, text2text generation, neural networks
DOI: 10.28995/2075-7182-2023-22-361-370

Анализ тональности по отношению к именованным сущностям с использованием подхода text2text generation

Иван Молошников
Moloshnikov_IA@nrcki.ru

Максим Скороходов
Skorokhodov_MV@nrcki.ru

Роман Рыбка
Rybka_RB@nrcki.ru
РТУ "МИРЭА"

Александр Наумов
Naumov-AV@nrcki.ru

Александр Сбоев
Sboev_AG@nrcki.ru
НИЯУ "МИФИ"

НИЦ "Курчатовский институт"
Москва, Россия

Аннотация

В данной статье представлено описание решения задачи анализа тональности по отношению к заданным именованным сущностям в новостных текстах, выполненного в рамках на Dialogue Evaluation в 2023 году (RuSentNE). В статье исследуется две типа нейросетевых моделей для решения задачи мультиклассовой классификации: рекурентная нейросетевая модель с вниманием и векторным представлением слов, полученных из языковых моделей, а также нейросетевая модель для генерации текста в заданном формате. Лучшие результаты показала генеративная модель с подобранными гиперпараметрами и дополнительной настройкой на данных соревнования и доступного открытого корпуса CABSAR. Предложенное решение достигает точности по метрике F1-macro: 59.33 для двух классов тональности и 68.71 для трех классов.

Ключевые слова: мультиклассовая классификация, тональный анализ, text2text генерация, нейронные сети

1 Introduction

Sentiment analysis in relation to an entity in a news text is an important direction in the field of opinion mining and Natural Language Processing (NLP). The demand for effective approaches to targeted sentiment analysis grows with the increasing amount of news data (Brauwers and Frasincar, 2022; Zhang et al., 2022).

In recent years, solutions of this problem have transitioned from traditional machine learning methods (such as support vector machine or decision trees) to modern neural network models based on the Transformer architecture (Vaswani et al., 2017), in particular, large language models (LLM) like BERT (Devlin et al., 2018) or GPT (Radford et al., 2018).

Several methods for targeted sentiment analysis task were proposed based on these approaches. For example (Sun et al., 2019), by constructing an auxiliary sentence from the target, this task can be converted to a sentence-pair classification task. The authors of that paper (Sun et al., 2019) used a pre-trained BERT model fine-tuned on Sentihood and SemEval2014 Task 4 datasets. This method achieved the accuracy of 0.933. Another work (Ma et al., 2017) uses the Interactive Attention Network (IAN) with attention mechanism between a target (words that belong to the named entity) and its context. Put together with a recurrent neural network based on Long Short Term Memory (LSTM) layers, that network improved the accuracy by 5.6% compared to the ordinary LSTM on SemEval 2014 Task 4 dataset (Laptop part) (Kirange et al., 2014). An approach (Zhang and Lu, 2019) that used a pre-trained BERT model with point-wise feed-forward networks (PFFN) and Multi-Head Attention (MHA) increased the accuracy further by 4.25%, up to 76.35%, on the same dataset.

Generative models based on text generation (text2text) like GPT, BART (Lewis et al., 2019), or T5 (Raffel et al., 2020) can be used for the targeted sentiment analysis as well. A paper (Mishev et al., 2020) presented the BART language model with a dense layer for classification. This model was fine-tuned on SemEval 2017 Task 5 dataset (Cortis et al., 2017), achieving the f1-score of 0.95. Another work (Zhang et al., 2021) proposes an adaptation of a pre-trained T5 model. The authors induce the T5 model to generate text with sentiment elements for named entities. This approach demonstrates the f1-score of 69.42 on SemEval 2016 data (restaurant part) (Pontiki et al., 2016). These works show the efficiency of text2text models for the targeted sentiment analysis task and highlight the potential of using pre-trained generative models.

For the Russian language, solving entity-oriented sentiment analysis task is complicated by the limited amount of available datasets. Previous SentiRuEval competitions in 2015 and 2016 (Loukachevitch et al., 2015; Loukachevitch and Rubtsova, 2016) provided several open datasets. Labeled sentiment entities in the common case are for which sentiment was labeled were defined as words and expressions that denote some important characteristic of an entity (like ‘kitchen’ or ‘interior’ in SentiRuEval2015-reviews) or predefined company names (for tweets in SentiRuEval2015-tweets and SentiRuEval2016-tweets). Besides the competition datasets, an open corpus CABSAR has recently been introduced (Naumov et al., 2020). This corpus contains Russian-language sentences for three different domains: news, tweets, and posts from social networks. Each sentence includes labeling for named entities (Person and Location) and sentiment, labeled for each entity by three classes (positive, negative, and neutral). Sentiment labeling was performed by crowdsourcing.

The RuSentNE-2023 dataset (Golubev et al., 2023) significantly expands the available sets of labeled examples in the Russian language for the entity-oriented sentiment analysis task. Therefore, the purpose of this work is an investigation of two neural network methods for this task using the RuSentNE-2023 dataset:

1. the first method is based on a multi-class classification task. Here we have chosen a well-known neural network architecture based on a recurrent neural network model, which has demonstrated high efficiency in similar tasks. Word vector representations are obtained from large language models known to be efficient in various classification tasks (see section 3.1);
2. the second method is based on the text generation (text2text) approach. The T5 model for the Russian language is used. In this case, several variants of labeling data for output text sequences are tried (see section 3.2).

Dataset	Num. of samples	Avg. length (in chars)	NE sentiment class		
			Positive	Neutral	Negative
RuSentNE (train part)	6637	151.2	856(12.9%)	4774(71.9%)	1007(15.2%)
CABSAR	6705	129.5	2289(34.1%)	3068(45.8%)	1348(20.1%)

Table 1: Number of examples for each sentiment class for the datasets used.

Named entity tag name	RuSentNE-2023						CABSAR		
	Train-subpart			Valid-subpart					
	Pos.	Neg.	All	Pos.	Neg.	All	Pos.	Neg.	All
PERSON	339	290	1546	82	73	388	1962	1078	5070
ORGANIZATION	146	210	1168	40	51	319	327	270	1635
COUNTRY	109	168	1022	33	44	252	-	-	-
PROFESSION	68	108	1352	11	23	314	-	-	-
NATIONALITY	23	29	221	5	11	55	-	-	-

Table 2: Number of examples for each sentiment class by NER tags for the datasets used.

Our main contributions are:

1. two neural network methods are compared for the entity-oriented sentiment analysis task in Russian news texts;
2. the efficiency of merging several open-source datasets is evaluated for each method;
3. the dependence of the accuracy on applying methods for reducing computations during network fine-tuning is studied.

This paper is organized as follows: Section 2 describes the task and characteristics of the datasets used. Section 3 presents methods used for the entity-oriented sentiment analysis task, including neural network architectures, word vector representations, and pre-trained models. Section 4 demonstrates metrics for model validation, experiment results, and hyper-parameters of the final models.

2 Task and Datasets

The RuSentNE-2023 task (Golubev et al., 2023) is sentiment analysis in relation to named entities in a news text in the Russian language. Named entities of the following types are predetermined in the text: PERSON, ORGANIZATION, PROFESSION, COUNTRY, and NATIONALITY. The purpose of this task is to classify each of the given named entities into three sentiment classes: positive, negative, or neutral. The RuSentNE dataset contains train, development, and final-test parts. Each part includes sentences with labeled entities and their types. The train part has sentiment labels for named entities. The development part allows one to check the performance metric on the interface of the competition website [1].

Table 1 shows some statistics on the train part. Analysis of this table shows the following:

1. there is an imbalance of examples for different sentiment classes: entities of neutral class are predominant;
2. entities with different sentiment classes can be in the same sentence.

Since labels of the development part are only available online, to optimize the hyperparameters of the models, the train part of data is divided into 80% and 20% while preserving the representativeness of examples of sentiment classes. The first subpart (train-subpart, 5309 examples) is used to train models, and the second subpart (valid-subpart, 1325 examples) is used to estimate the efficiency of the models' hyperparameters. Table 2 shows the total number of examples for each sentiment class by NER tags for

[1] RuSentNE on CodaLab: `https://codalab.lisn.upsaclay.fr/competitions/9538`

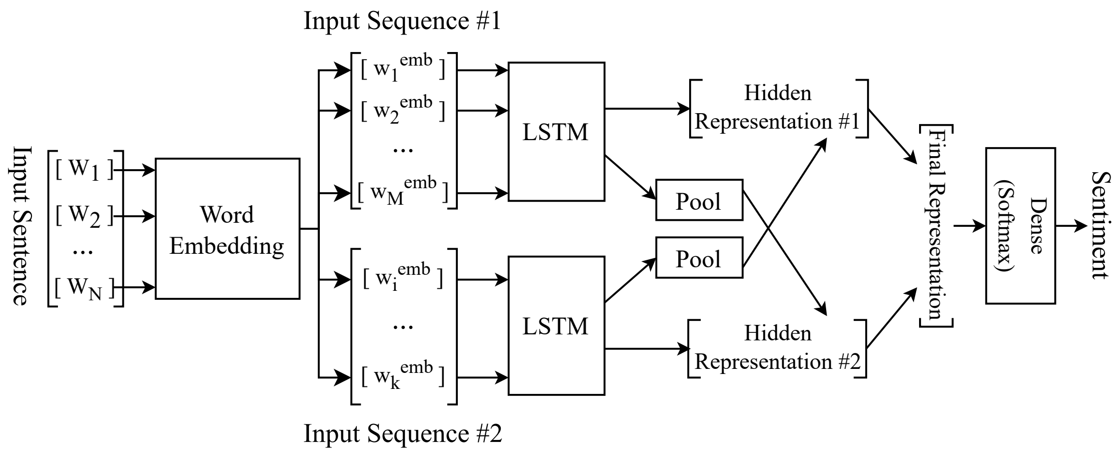

Figure 1: IAN model architecture.

train- and valid- subparts.

CABSAR (Naumov et al., 2020) is the closest corpus available in the Russian language to the dataset presented in the RuSentNE-2023 competition(Golubev et al., 2023). Therefore, this corpus was used to increase the number of train subpart examples. This corpus contains 6705 sentences in the Russian language from several sources: 2105 from LiveJournal blogs, 2603 from Lenta.ru news, and 1997 from Twitter. Named entity sentiment is labeled in these sentences by crowdsourcing. Table 1 and Table 2 show the number of samples and named entities for each sentiment class.

3 Methods

3.1 Multiclass Classification

This approach is based on a deep neural network with attention (Interactive Attention Network - IAN) (Ma et al., 2017). The authors of CABSAR (Naumov et al., 2020) used it to obtain baseline accuracy for the entity-oriented sentiment analysis task. Therefore, it was chosen to evaluate the accuracy of the RuSentNE-2023 task as a multiclass classification approach.

This method analyzes the input text sentence and splits it into two input sequences: for context (Input Sequence #1 in Figure 1) and target (Input Sequence #2 in Figure 1). The first input sequence is all the words of the sentence that contain a named entity, and the second input sequence is the words that belong to the same named entity for which sentiment is to be predicted. Word vectors obtained from these sequences are fed to a recurrent neural network based on LSTM layers with attention mechanism (see Figure 1).

The original IAN model used the GloVe (Pennington et al., 2014) as a word embedding model. The authors (Naumov et al., 2020) obtained a 0.7 f1-macro score on CABSAR using the ELMo language model (Peters et al., 2018) as a word embedding.

The following language models for word embedding are studied:

- ELMo (Peters et al., 2018) – word vector representations are formed based on Bidirectional LSTM layers. For the Russian language, a model trained on the Wikipedia text corpus is used from the DeepPavlov library [2].
- RuBERT (Kuratov and Arkhipov, 2019) is a model based on the Transformer architecture, obtained from Multilingual BERT pre-trained on 104 languages (Devlin et al., 2018). Then, that Multilingual BERT was trained on Wikipedia text corpus in the Russian language. The RuBERT used in this

[2]ELMo on Russian Wikipedia: http://docs.deeppavlov.ai/en/master/features/pretrained_vectors.html#elmo

Type of Input	Text of Sentence
A.	В роли **Тони Сопрано** Гандольфини удалось впервые создать образ гангстера с человеческим лицом. In the role of **Tony Soprano**, Gandolfini managed to create the image of a gangster with a human face for the first time.
B.	В роли **[Тони Сопрано]** Гандольфини удалось впервые создать образ гангстера с человеческим лицом In the role of **[Tony Soprano]**, Gandolfini managed to create the image of a gangster with a human face for the first time.
C.	**Тони Сопрано**. В роли **[PERSON]** Гандольфини удалось впервые создать образ гангстера с человеческим лицом **Tony Soprano**. In the role of **[PERSON]**, Gandolfini managed to create the image of a gangster with a human face for the first time.

Table 3: Options of data input for the text generation model.

paper is the large version of RuBERT is taken from the Huggingface[3] library.

- XLM-Roberta (Conneau et al., 2019) is a model based on the Transformer architecture, trained on 2.5 TB of data from CommonCrawl. The CommonCrawl data contains text in 100 languages, of which the Russian language is one of the most representative.

3.2 Text2text Generation

This approach is based on a generative neural network model with the Transformer architecture – T5 (Raffel et al., 2020). This model generates a new text from an input text. It consists of Encoder and Decoder blocks. The Encoder block accepts input text sequences as their word vector representation. The Decoder block generates new output text sequences.

Different options of input information for the text generation model were tested (see examples in Table 3):

A: source sentence text without any changes.

B: named entity in source sentence text is highlighted by square brackets, e.g. 'Tony Soprano' is replaced by '[Tony Soprano]'.

C: named entity type is replaced in source sentence text, e.g. 'Tony Soprano' is replaced by '[PERSON]'. The named entity text is added at the start of the source text.

Output text is one of the possible sentiments for the analyzed named entity: negative, neutral or positive.

Two T5-based models are considered within this approach: ruT5-base [4] and ruT5-large [5]. These models are an adaptation of T5-base and T5-large models for the Russian language. Wikipedia, books, news, and CommonCrawl texts were used to train them. The model dictionary size is 32101 tokens. The number of parameters is 220 million for the "base" model and 737 million for the "large" model.

4 Experiments

4.1 Metrics

As mentioned in the evaluation criteria of the RuSentNE-2023 competition, the main performance metric is the macro F1_pn-score, and the macro F1-score will be considered auxiliary. For macro-averaging,

[3]RuBERT-large: https://huggingface.co/ai-forever/RuBERT-large
[4]RuT5-base: `https://huggingface.co/ai-forever/ruT5-base`
[5]RuT5-large: `https://huggingface.co/ai-forever/ruT5-large`

№	Embbedding name	Add. data	Hyper. optim.	MLM tune	Valid-subpart		Final-test-part	
					F1-macro	F1_pn-score	F1-macro	F1_pn-score
1	ELMo	-	-	-	62.57	53.14	54.53	42.36
2	XLM-R-large	-	-	-	54.47	45.86	50.47	38.52
3	RuBERT-large	-	-	-	61.38	52.00	55.37	43.76
4	ELMo	+	-	-	61.10	50.32	54.96	44.14
5	XLM-R-large	+	-	-	50.49	42.38	51.12	41.97
6	RuBERT-large	+	-	-	60.68	51.53	55.94	46.20
7	XLM-R-large	-	+	-	58.07	47.36	54.66	42.16
8	RuBERT-large	-	+	-	65.82	56.29	57.09	44.67
9	XLM-R-large	+	+	-	64.63	54.44	57.45	45.36
10	RuBERT-large	+	+	-	66.79	56.80	59.46	48.17
11	XLM-R-large	-	+	+	64.04	54.22	56.78	44.24
12	RuBERT-large	-	+	+	67.76	58.68	56.17	43.85
13	XLM-R-large	+	+	+	64.69	54.99	56.05	46.16
14	RuBERT-large	+	+	+	65.88	56.21	54.80	44.77
-	RuSentNE-2023	-	-	-	-	-	56.71	40.92

Table 4: Results of the IAN model.

the F1-score calculation is averaged for each class separately. F1_pn-score is calculated by averaging the F1-score of two sentiment classes: negative and positive, excluding the neutral class.

4.2 Interaction Attention Network

The following experiments was performed with the IAN model:

- comparison of language models as word embeddings as part of the IAN model. In this case, hyperparameters were used from (Naumov et al., 2020). Only competition data are used for training;
- analysis of the impact of expanding the training samples by using additional data (CABSAR corpus);
- running hyperparameters optimization experiments with the RayTune library (Liaw et al., 2018) and selecting the more effective combination. The OpTuna framework (Akiba et al., 2019) was used as a search algorithm. The following hyperparameters were optimized: the size of the LSTM layer (hidden_dim), learning rate, batch size, etc.;
- pre-training of the language model used in IAN on the Masked-Language Modeling (MLM) task with 5000 steps and batch_size=64 on the train-subpart of the RuSentNE-2023 dataset. The model checkpoint was saved every 1000 steps, and the best one on the valid-subpart was selected.

The results of these experiments are shown in Table 4. Analysis of the results shows that the IAN model with word embeddings from the RuBERT-large model, using additional data, and with the hyperparameters optimization (exp. №10) has the best results among other methods: 48.17% and 59.46% by F1_pn-score and F1-score respectively on the final-test part of the data. It is better than the RuSentNE-2023 baseline by 7.25% and 2.75% respectively. In addition, there is an increase in scores in all experiments on the final-test part with using additional CABSAR data. Note that after hyperparameter optimization, IAN model with embeddings from RuBERT-large showed better results than with embeddings from XLM-R-large, although it has less parameters.

The best results of these models were obtained with the hyperparameters presented in Table 6.

4.3 ruT5 Model

Experiments with this model included: selecting the more effective option for input data representation, and evaluating the accuracy when using additional samples (CABSAR corpus) in the training part of the

Model Name	Extra Data	F1-macro	F1_pn-score
ruT5-base (type A.)	-	47.47	40.6
ruT5-base (type B.)	-	66.11	56.94
ruT5-base (type C.)	-	64.03	54.77
ruT5-large (type C.)	-	67.27	58.9
ruT5-base (type B.)	CABSAR	67.57	57.96
ruT5-base (type C.)	CABSAR	67.78	58.48
ruT5-large (type C.)	**CABSAR**	**68.71**	**59.33**
RuSentNE-2023	-	56.71	40.92

Table 5: Results of the text generation approach based on ruT5 model on the final-test part.

	ruT5-large	IAN-elmo	IAN-RuBERT-large	IAN-XLM-R-large
input text length	164	-	-	-
output text length	4	-	-	-
learning rate	10^{-5}	10^{-2}	$3.7 * 10^{-4}$	$2.2 * 10^{-5}$
batch size	64	4	64	128
LSTM hidden_dim	-	32	256	256
dropout	-	0.3		
train epochs	50	300 with early stopping		
optimizer	Adam			

Table 6: Hyperparameters for the best models.

data. Table 5 shows the results of experiments on the final-test part with ruT5 models.

As a result, the best model is ruT5-large with type "C" representation of input data, trained on the extended train part. Adding the CABSAR corpus, F1-score increases by 1.5%.

Text sequence generation is performed by Beam search with the number of beams equal to 2. For the final model, the hyperparameters were presented in Table 6.

The input text length (number of tokens) is set based on the maximum source sentence length in the competition dataset. The output text length is set based on the maximum number of tokens among the words "негативная" ("negative"), "нейтральная" ("neutral"), "позитивная" ("positive").

Calculations were conducted on the following equipment:

- ruT5-base model: Intel Xeon E5-2650v2 (2.6 GHz), 128 GB RAM, Nvidia Tesla K80;
- ruT5-large model: Intel Xeon E5-2630v4 (2.2 GHz), 64 GB RAM, Nvidia Tesla V100.

Additionally, a comparison of accuracy was performed for ruT5-large models (type C.) trained with and without Parameter-Efficient Fine-Tuning (PEFT)(Sourab Mangrulkar, 2022). In this case, the possibility of saving model accuracy was checked when training on small computing resources:

- Intel Xeon E5-2650v2 (2.6 GHz), 128 GB RAM, Nvidia Tesla K80

LORA(Hu et al., 2021) was used as the PEFT method. This method performs low-rank adaptation. It fixes weights of the pre-trained model and introduces trainable rank decomposition matrices into each level of the Transformer architecture. As a result, accuracy declined by 4% and 2% by F1_pn-score and F1-score respectively. However, it achieved a significant reduction in computing power requirements.

5 Discussion

A comparison of the best model score on the valid-subpart demonstrates a superior performance of the ruT5-large model for 4 of the 5 Named Entity (NE) tags (see Table 7). The accuracy of the sentiment classification for the NATIONALITY NE tag is similar for both models. The best accuracy (F1_pn-score

Named entity tag name	ruT5-large (type C)			IAN-RuBERT-large (exp-№10)		
	F1-micro	F1-macro	F1_pn-score	F1-micro	F1-macro	F1_pn-score
PERSON	73.71	69.98	65.19	70.1	64.57	57.75
ORGANIZATION	78.68	69.76	61.53	75.55	63.16	52.39
COUNTRY	85.32	81.05	76.28	80.56	71.23	62.59
PROFESSION	90.45	65.35	50.61	88.85	61.24	44.97
NATIONALITY	85.45	77.38	70.83	85.45	77.84	71.08

Table 7: Results of the best models by NER tags.

> 70) is achieved for the COUNTRY and NATIONALITY NE tags, and the worst (F1_pn-score 50) for PROFESSION. There are several factors involved in this, the most important of which is the balance of classes in the dataset used. For example, the proportion of the positive and negative sentiment classes to the total number of samples is 28% for COUNTRY and 25% for NATIONALITY NE tags. In contrast, the same proportion for the PROFESSION NE tag is 13%.

In this regard, improved accuracy can be achieved by:

- increasing the number of training data examples for the target task. This is confirmed by experiments with the addition of CABSAR data to the train-part of the RuSentNE-2023 dataset;
- applying more complex generative neural network models and training on larger datasets (e.g. GPT(Radford et al., 2018), T5-XXL(Raffel et al., 2020)).

Both datasets used in this paper extend the number of labeled examples for the joint task of named entities recognition and entity-oriented sentiment analysis for the Russian-language texts. However, they contain labels of mostly simple named entity samples, with a continuous word sequence and non-overlapping entities. The proportion of such complex samples for the RuSentNE-2023 and CABSAR datasets is 62 of 6637 (<1%) and 110 of 6705 (1.6%), respectively. Therefore, developing and researching named entity-oriented sentiment analysis methods for complex named entities is a very promising task.

6 Conclusion

This research shows the advantage of using a text generation approach for the entity-oriented sentiment analysis task. According to the results, the best accuracy was shown by the ruT5-large model with training on an extended dataset and a special input text representation. It was uploaded to the competition leaderboard as our final submission and showed a result of 59.33, which is 19% better than the baseline method in terms of the RuSentNE-2023 competition (Golubev et al., 2023). This result took the 5th place in the final rating leaderboard.

Our experiments with the multi-class classification model show that this method can be used for the target task. When using additional training data, a large language model for extracting word embeddings, and a hyperparameter optimization method, results were obtained that exceeded the baseline by 8%.

Further research will be focused on the improvement of input and output text data representation methods in generative neural network models, including for targeted sentiment analysis task.

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. // *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Gianni Brauwers and Flavius Frasincar. 2022. A survey on aspect-based sentiment classification. *ACM Computing Surveys*, 55(4):1–37.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Keith Cortis, André Freitas, Tobias Daudert, Manuela Huerlimann, Manel Zarrouk, Siegfried Handschuh, and Brian Davis. 2017. Semeval-2017 task 5: Fine-grained sentiment analysis on financial microblogs and news. // *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, P 519–535.

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

Anton Golubev, Nicolay Rusnachenko, and Natalia Loukachevitch. 2023. RuSentNE-2023: Evaluating entity-oriented sentiment analysis on russian news texts. // *Computational Linguistics and Intellectual Technologies: papers from the Annual conference "Dialogue"*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

DK Kirange, Ratnadeep R Deshmukh, and MDK Kirange. 2014. Aspect based sentiment analysis semeval-2014 task 4. *Asian Journal of Computer Science and Information Technology (AJCSIT) Vol*, 4.

Yuri Kuratov and Mikhail Arkhipov. 2019. Adaptation of deep bidirectional multilingual transformers for russian language. *arXiv preprint arXiv:1905.07213*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Richard Liaw, Eric Liang, Robert Nishihara, Philipp Moritz, Joseph E Gonzalez, and Ion Stoica. 2018. Tune: A research platform for distributed model selection and training. *arXiv preprint arXiv:1807.05118*.

NV Loukachevitch and Yu V Rubtsova. 2016. Sentirueval-2016: overcoming time gap and data sparsity in tweet sentiment analysis. // *Computational Linguistics and Intellectual Technologies*, P 416–426.

Natalia Loukachevitch, Pavel Blinov, Evgeny Kotelnikov, Yulia Rubtsova, Vladimir Ivanov, and Elena Tutubalina. 2015. Sentirueval: testing object-oriented sentiment analysis systems in russian. // *Proceedings of International Conference Dialog*, volume 2, P 3–13.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. *arXiv preprint arXiv:1709.00893*.

Kostadin Mishev, Ana Gjorgjevikj, Irena Vodenska, Lubomir T Chitkushev, and Dimitar Trajanov. 2020. Evaluation of sentiment analysis in finance: from lexicons to transformers. *IEEE access*, 8:131662–131682.

Aleksandr Naumov, R Rybka, A Sboev, A Selivanov, and A Gryaznov. 2020. Neural-network method for determining text author's sentiment to an aspect specified by the named entity. // *CEUR Workshop Proceedings*, P 134–143.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. // *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, P 1532–1543.

ME Peters, M Neumann, M Iyyer, M Gardner, C Clark, K Lee, and L Zettlemoyer. 2018. Deep contextualized word representations. arxiv 2018. *arXiv preprint arXiv:1802.05365*, 12.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammed AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. // *ProWorkshop on Semantic Evaluation (SemEval-2016)*, P 19–30. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Lysandre Debut Younes Belkada Sayak Paul Sourab Mangrulkar, Sylvain Gugger. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence. *arXiv preprint arXiv:1903.09588*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Qiuyue Zhang and Ran Lu. 2019. A multi-attention network for aspect-level sentiment analysis. *Future Internet*, 11(7):157.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2021. Towards generative aspect-based sentiment analysis. *// Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, P 504–510.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2022. A survey on aspect-based sentiment analysis: tasks, methods, and challenges. *IEEE Transactions on Knowledge and Data Engineering*.