Multilingual Sentiment Analysis and Toxicity Detection for Text Messages in Russian

Darya Bogoradnikova, Olesia Makhnytkina, Anton Matveev, Anastasia Zakharova, Artem Akulov
ITMO University
Saint Petersburg 197101, Russian Federation
dabogoradnikova, makhnytkina, aymatveev, aazakharova23@itmo.ru, 287550@niuitmo.ru

***Abstract*—In this paper, we discuss an approach to sentiment analysis and emotion identification for user comments. The solution is threefold: 1) topic detection, 2) sentiment evaluation, 3) toxicity detection and toxic spans localization. The lack of significantly large training data for the Russian language is handled by utilizing multilingual word embeddings, the adversarial domain adaptation model, and data augmentation. We present an overview of various preprocessing pipelines for topic modeling and highlight the LDA-Mallet model which demonstrates the best performance. For sentiment analysis and toxicity detection, we examine the efficacy of a support vector machine and a deep neural network with a multilingual language model and adversarial domain adaptation that allows us to train algorithms with datasets in the English language. All methods are tested with a dataset of user comments to various online-courses and adjusted to provide support for the development of a virtual dialogue assistant for conducting virtual exams.**

I. Introduction

As online education systems become more widespread and experience their user bases grow, new problems and solutions arise, including solutions based on human-machine communication. One of the key aspects of human-machine communication is assessment of an emotional state. That includes evaluation of the emotional background of dialogues and user comments. In this paper, we describe a system for assessing the sentiment and emotionality of texts, which allows identifying a person's attitude to various objects and processes, including detection of toxic messages, as an important component of a virtual dialogue assistant for conducting remote examination [1]. The emotion coloring analysis for texts has a broad meaning which can include emotion analysis as well as mood analysis, while models for emotions and moods can be either discrete (categorical) or continuous. Categorical models most often employ the six main emotional states identified by Ekman [2]: anger, disgust, fear, happiness, sadness, and surprise , or their derivatives, but still in a discrete manner. Researchers are often only interested in valence shifts, in this case, the classification is only performed in one dimension: positive to negative. This case is called sentiment analysis. Commonly, this classification contains two, three, or five levels. In the first case, texts are classified as either "positive" or "negative", in the second case, there is also "neutral", and finally, "positive" and "negative" are subdivided by intensity.

Statements containing obscene language, a threat, a sharply negative attitude towards a person or a situation, insults, or other statements that can offend or humiliate a reader are quite often called toxic [3], although there is currently no precise definition of this term. If such statements are of a pronounced racist, sexist, religious, or other nature that can form a negative attitude towards a particular social group, then such messages are more often referred to as hate speech [4].

Of course, on many Internet platforms, there is manual moderation, which is designed to prevent toxic content from entering the space, but such measures do not always bring the desired result. Due to the volumes of incoming messages, moderators may miss a toxic passage or not cope with their work quickly enough, which will affect the pace of publications of the proposed content, which may also affect the user's attitude to the resource. For this reason, for several years now, research has been going on related to the automation of the process of detecting and classifying toxicity in text messages, among which we can point out contests from Google and Jigsaw on the Kaggle platform, which not only drew attention to this problem but also gave an active impetus to new research due to the provision of public access to several professionally marked rather large datasets. They also released the Perspective API software, which can determine the toxicity of texts in English, Spanish, French, German, Portuguese, Italian, and Russian. With the advent of new types and architectures of neural networks, the task of multilingual classification of toxic messages has also emerged. The presentation of free access to the Perspective API system stimulates researchers to create more accurate models, even if for the same language, however, the lack of appropriate training data is still an issue. Usually, such systems classify a message as a whole, without regard to which segment turns it toxic. In other words, those systems do not look for segments that infuse a message with emotional coloring; sometimes, of course, it is not even possible since toxicity can arise not from a particular word or a phrase, but from an overall construction of a message. When it is possible, however, to identify such segments, that can be utilized to provide support for human-moderators who regularly have to deal with lengthy messages and would prefer to see references to particular parts of a message instead of just an overall toxicity score. A system able to locate toxic spans of a message would significantly help to alleviate the issue.

Analyzing toxicity and sentiment, it is also important to identify the object: which topics find positive and negative responses from users. The number of topics, generally, varies

across discussions, thus, unsupervised machine learning techniques are better fit for object detection. In this paper, we discuss approaches to the specified kinds of problems: sentiment and toxicity analysis, toxic spans detection, and topic identification. Furthermore, we discuss data augmentation and various combinations of methods for preprocessing data and word embedding. Solving these problems is essential for the evaluation of the emotional background of dialogues and user comments.

This paper is organized in the following way: in the "Related works" section we review previous research on the discussed topics; in the "Data" section we outline all datasets gathered for this research; in the "Methods" section we present models and methods we experiment with in this research; in the "Evaluation" section we present the metrics we employ for evaluating models, and in the "Results" section we present the results of our evaluation.

II. Related works

Initially, the assessment of the emotional coloring of a text was made based on the methods of sentiment analysis. So, the classification of comments or messages on social networks into negative, neutral, or positive remains key in this field [5], [6], since it allows to determine the user's attitude to any object or phenomenon. The sentiment analysis problem is well-researched in general as well as for the Russian language in particular. Here, we mainly focus on the detection of toxic messages and segments.

Speaking about toxic messages, it is worth noting that this term appeared after the launch in December 2017 of the Toxic Comment Classification Challenge from Google and Jigsaw, the main task of which was the identification and classification of toxic online comments. Up to this point, the identification of hateful and offensive statements was performed within the framework of solving the problem of detecting and classifying hate speech [7], work on which continues to this day [4], [8], [9]. One of the main difficulties that can be encountered when training with a teacher is having a well-annotated and sufficiently large dataset. Most of the work related to the classification of toxic messages utilizes kits provided in the framework of competitions from Google and Jigsaw. In 2018, a dataset was presented containing 223549 comments in English, of which 159571 comments refer to the training set [10], [11]. The dataset released in 2019 contains over 1.8 million comments. In 2020, two more datasets were added to those two datasets, one of which contains 8,000 annotated comments in Spanish, Italian and Turkish, the other contains 63,812 unlabeled comments in Turkish, Italian, Russian, French, Portuguese, and Spanish. When solving the problem of detecting toxic messages, the chosen method of obtaining vector representations of words has a significant impact. In several works [4], [12], studies of various ways of obtaining embeddings are conducted: a bag of words or inverse frequency representation (TF-IDF), Word2vec, FastText, GloVe, Bert. Some authors use multiple embeddings simultaneously [13].

Currently, to solve the problem of detecting toxic messages, classical machine learning methods are employed: logistic regression, random forest, support vector machine, decision trees and their modifications, which are compared with neural networks [10], [14] or considered independently [15], [16]. Among the best architectures of deep neural networks for detecting toxic messages are convolutional and recurrent neural networks LSTM, GRU, as well as their various combinations and ensembles [17]. The toxic spans detection problem firsts gained noticeable attention this year within the "SemEval 2021" challenge, specifically "SemEval 2021. Task 5: Toxic Spans Detection". It is hard to find any previous works distinctively focusing on this problem, however, mentions of it can be found in several related articles. In [18], authors review a possibility for semi-automatic moderation where in addition to automatic classification of messages into insulting or neutral a moderator is also presented with the original message with suspicious words highlighted. Among modern methods for general span detection, we can point out SpanBERT [19], a pre-training method that is designed to better represent and predict spans of text. This method expands on BERT by masking random contiguous spans, rather than random individual tokens, and predicting the entire masked span from the observed tokens at its boundary solely using the context in which they appear.

Since the problem is relatively new, there are simply not enough significantly large datasets to train models. To confront the issue, the authors suggest two pathways: 1) multilingual embeddings which allow to train models in one language and test them with another, and 2) collection and labeling of a new dataset and expanding it with augmentation. The first route was employed by the authors for the detection of toxic messages and demonstrated an acceptable performance [20].

In this work, we expand this approach to sentiment analysis and toxic spans detection problems and also conduct experiments with text data augmentation.

A set of simple augmentation techniques is presented in the EDA algorithm [21], which consists of four operations: substitution by synonyms, random insertion, random permutation, and random deletion.

When augmenting textual data, it is important to preserve the meaning of the text, often by replacing words with synonyms using various dictionaries, for example, WordNet [22] or using pre-trained language models, such as BERT, GPT2, Word2Vec, Glove [23]–[25].

An alternative to generating paraphrases is a reverse translation when a sentence is translated into one or several languages, and then re-translated into the original language, resulting in a differently formulated original sentence [26]. Also, it is possible to translate an already existing tagged dataset into the language required for experiments [27]–[29]. Another approach is to "glue" several messages together that are close in context and obtain a new, longer message [30] or add some noise, decipher common abbreviations such as date, units of measurement, and location, or expand abbreviated forms of words [31]. Further, one can use syntax tree transfor-

mations or lexical substitutions based on various augmentation strategies to find semantically related substitutions to create new training instances [32]. There is also another approach based on changing the polarity of communication [33]. In this case, for a message with a positive color, a message with a negative color is generated, which also makes it possible to increase the volume of the training sample.

Detection of the sentiment object most often referred to by users can be accomplished via topic modeling algorithms. In [34] the authors train a dialog model for a dialog systems for customer support in an unsupervised way, avoiding the need for labeled corpora. The authors of [35] investigate how unsupervised context-dependent algorithms for automatic generation of synonyms for keywords can facilitate automatic detection of domain concepts. Topic detection can be viewed as a simultaneous clustering of words and documents by semantic closeness. Commonly, the clustering is done in a fuzzy manner, when a document can be assigned to multiple topics. There are various methods for producing topic models. Some of the more widely used among them are Latent Dirichlet Allocation (LDA) and Latent Semantic Analysis (LSA) [36].

In recent years, a great deal of attention is focused on combining word embeddings with traditional methods [37], [38].

Developing a dialogue assistant requires solving the semantic analysis and toxicity detection problems as well as topic detection. In online courses the number of comments is usually rather small and traditional methods for topic modeling can be successfully employed there, however, the lack of training data in Russian is an obstacle for supervised training of models for semantic analysis and toxic message detection required for training a classifier.

III. Data

In our experiments, we had datasets obtained by different methods. Some of them were collected independently, some were obtained by data augmentation. Let us review them in more detail.

A. Target dataset

The target dataset is a collection of user comments in Russian used for testing both the classifier and topic modeling algorithms. The dataset contains 1703 user reviews in Russian from two online education platforms: Coursera and Stepik. The dataset is annotated by five experts for the following qualities: sentiment and toxicity. Sentiment is discretely categorized as either very positive, positive, neutral, negative, or very negative and respectively labeled with scores from 1 to 5. Toxicity is binary classified for the presence or absence of toxic words or passages in user comments.

The dataset contains 197 comments with a score of 1, 152 comments with a score of 2, 262 comments with a score of 3, 310 comments with a score of 4, and 782 comments with a score of 5.

16 comments are labeled as toxic and 1687 comments are labeled as not toxic.

Here is an example of a non-toxic comment (translated from Russian to English) with a sentiment score in the highest bracket: “I have gotten a clear understanding of what machine learning is as well as some theoretical background. For almost every topic there is a reference to relevant lecture notes, which helps a lot to achieve a better understanding. And most important, there are very interesting and helpful practical tasks.” The longest comment is 457 words long, the shortest comment is 1 word long, and the average length is 32 words.

The dataset was assembled by ITMO University students.

Additionally, there are several datasets used for training and testing.

B. Datasets for sentiment analysis

1) A collection of user comments in English from kaggle.com for sentiment analysis published as “100K Coursera’s Course Review Dataset”, containing 107018 user comments for various courses in English on the online education platform Coursera, which corresponds to the domain of the dataset in Russian. For sentiment analysis, the classes are the same as for the dataset in Russian, a discrete scale from 1 to 5 marks a user’s review of a course: for a 5-star rating, the review is labeled as very positive, positive for 4-star, neutral for 3-star, negative for 2-star, and very negative for 1-star. Since the dataset is not balanced, which can impair the classifier’s performance, we removed samples from some classes to achieve a uniform distribution.
2) A collection of short texts from Rubtsova [39] that contains 114 991 positive and 111 923 negative comments from Twitter in Russian.

C. Datasets for toxicity comment detection

Here is a group of datasets collected from the Russian-language social network VKontakte. These datasets contain comments from 6 user groups, divided into three classes: education, news, and entertainment. All comments were collected between April 1 and 30, 2020. Representative subsamples were selected from each group with an error of 5%. Selected comments were manually annotated by three experts with a Cohen kappa coefficient for their agreement between 40% and 60%, depending on the group. Thus, assessments of the quality of work of automatic algorithms for the classification of texts by toxicity should correspond to these levels.

The “Education” topic is represented by two communities: “Habr” and “Suiauctus”. “Habr” dataset, collected from comments to posts in this group, contains 359 comments, of which 324 (90%) are non-toxic, 35 (10%) - toxic. “Suiauctus” dataset contains 371 comments, of which 339 (91%) are non-toxic, 32 (9%) are toxic.

The “News” topic is represented by two communities: “The fifth channel. News” and “Mash”. “The fifth channel. News” dataset contains 380 comments, of which 305 (80%) are non-toxic, 75 (20%) are toxic. “Mash” dataset contains 381 comments, of which 287 (75%) are non-toxic, 94 (25%) - toxic.

The “Entertainment” theme is represented by two communities: “MARVEL/DC” and “IGM”. “MARVEL/DC” dataset contains 381 comments, of which 325 (85%) are non-toxic, 56 (15%) are toxic.“IG” dataset contains 383 comments, of which 341 (89%) are non-toxic, 42 (11%) are toxic.

Subsequently, the resulting datasets were divided into two main parts: test, which is 30% of the total, and training.

For the test subset, the maximum length of a comment is 219 words and the minimum length is 1 word. The average length is 13 words. For the training subset, the minimum and the average length are the same, while the longest comment is 451 words long.

D. Datasets for toxicity span detection

A dataset published at CodaLab for “SemEval 2021 Task 5: Toxic Spans Detection” challenge is used for building and training a model for toxic spans detection. It contains 7939 messages, 485 of which are not annotated with toxic spans.

In this dataset, the maximum length of a message is 192 words, the minimum length is 1 word, and the average is 36 words. The maximum length of a toxic span is 994 symbols, the minimum is 2 symbols, and the average is 18.6 symbols. To point out, a toxic span is represented by an ordered sequence of positions of symbols assessed to belong to a toxic word or a toxic phase including white spaces within it. The positions are numbered from zero starting from the first symbol of a comment. For example, a comment “Another IDIOT!, with fake data and resources. LOL!” contains a toxic span [8, 9, 10, 11, 12] which belongs to the toxic word “IDIOT”.

For experiments on comments in Russian, we employed datasets extracted from the target dataset and datasets from VKontakte. Only messages labeled as toxic were selected.

6 datasets from VKontakte were merged into a single dataset with 334 comments. The longest toxic span for a message from this dataset is 116 symbols, and the shortest is 3 symbols. The average length of a toxic interval is 12 symbols. Here is an approximately translated from Russian to English example of a message with a toxic span: “you sir are knowledgeable in perversion”. In this case, the toxic span covers only the last word.

IV. METHODS

1) Text pre-processing: There are several features of user’s messages from the internet that are to be considered. Those messages require preprocessing, for the selected datasets we performed removal of references to other users’ names and hyperlinks, substitution of abbreviations, replacement of emoticons with relevant words, removal of punctuation, and word tokenization. For each problem, except for a toxic span detection, tokens were also subsequently lemmatized.

For all datasets used for sentiment analysis and toxic spans detection in user comments, word embeddings are obtained via context-free Multilingual BERT and xlm-roberta which can make cross-linguistic generalizations; the resulting vector space makes it possible to handle cross-linguistic problems. For experiments, we used two pre-trained Multilingual BERT models: bert-base-multilingual-uncased (12-layer, 768-hidden, 12-heads, 168M parameters, trained on lower-cased texts in the top 102 languages with the largest Wikipedias), bert-base-multilingual-cased (12-layer, 768-hidden, 12-heads, 179M parameters, trained on cased texts in the top 104 languages with the largest Wikipedias), and xlm-roberta (270M parameters with 12-layers, 768-hidden-state, 3072 feed-forward hidden-state, 8-heads, trained on 2.5 TB of newly created clean CommonCrawl data in 100 languages).

For toxic message detection, we utilize multilingual embeddings described before. The choice of the embedding method and classification method was based on the results of past work [20]. In this work, we looked at word embedding methods such as Word2Vec, FastText, GloVe, BERT. As methods for classifying toxic comments, we used Naïve Bayes, Random Forest, Logistic regression, Support Vector Machine, Majority vote, and Recurrent Neural Networks. The best result was shown by the combination of word2vec+SVM, which we will use.

To obtain embeddings, we used a pre-trained Russian-language word2vec model [40], made publicly available by the RusVectors resource. To use this model, additional information about the part-of-speech markup of the analyzed texts is required. The markup must conform to the Universal PoS Tagging format. If this information is not present, the system would try to automatically determine the part-of-speech.

2) Data augmentation: The training samples underwent the following augmentation procedure:

Paraphrasing. By definition, paraphrasing is an alternative external representation in the same language that expresses the same semantic content as the original form. Paraphrasing can occur at several levels. For example, words that have the same meaning — synonyms — can also be viewed as lexical paraphrasing. There is paraphrasing at the level of a group of words or phrases (phrase paraphrasing), as well as at the level of complete sentence (re-phrasing sentences) [31]. Using the paraphrasing algorithm, 1535 new messages were obtained.

EDA Algorithm (Easy Data Augmentation) - simple methods of data augmentation to improve performance when performing text classification tasks [21]. EDA consists of four simple but effective operations:

1) Synonym Replacement (SR): n words from a sentence that are not stop words are randomly selected and each of these words is replaced by one of its synonyms, chosen at random.
2) Random Insertion (RI): finds a random synonym for a random word in a sentence that is not a stop word. This synonym is embedded in a random position in the sentence. The action is repeated n times.
3) Random Swap (RS): two words in a sentence are randomly selected and swapped. The procedure is repeated n times.
4) Random Deletion (RD): random deletion of each word in a sentence with probability p.

Translate dataset. This dataset (www.kaggle.com/c/jigsaw-multilingual-toxic-comment-classification/discussion/150334)

was made publicly available by one of the participants in the "Jigsaw Multilingual Toxic Comment Classification" competition. This dataset was obtained by translating the first 110 thousand messages of the training set into 6 languages present in the test set, including Russian, using the Yandex API.

Pseudo-Labeling dataset. Pseudo-Labeling is used in various subject areas [41], [42] with a significant lack of well-labeled data. In practice, this method is applied when there is a good model that can provide reasonably accurate results. In this work, pseudo-labels were obtained for the Russian-language part of the test set of the Jigsaw Multilingual Toxic Comment Classification competition. Google's Perspective API was used as a classification model.

3) Sentiment analysis: For user comments sentiment analysis we used a support vector machine (SVM), a recurrent neural network LSTM, and a modified adversarial domain adaptation model (ADA) similar to the one from our earlier work [20].

4) Detection of toxic comments: For the detection of toxic messages, we used a support vector machine (SVM), which showed results comparable to classifiers based on recurrent neural networks, which is explained by relatively small data sets. SVM is one of the most popular classification and regression tools. It is based on the simple idea of finding a hyperplane that optimally separates samples. This algorithm is flexible enough and can be modified per a specific task.

5) Detection of toxic spans: For the detection of toxic spans, we employed two models based on the Transformer network.

The first model is BERT [43], which stands for Bidirectional Encoder Representations from Transformers. It is designed to pre-train deep bidirectional representations from an unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications.

Pretraining multilingual language models at scale leads to significant performance gains for a wide range of cross-lingual transfer tasks. The second model, XML-RoBERTa [44], is a transformer-based masked language model. It is trained on one hundred languages using more than two terabytes of filtered CommonCrawl data.

For these models, we used AdamW optimizer with $3 * 10^{-5}$ learning rate and batch size 8. Cross-entropy is used as a loss-function.

6) Topic modeling: For topic modeling, we consider stochastic Latent Semantic Analysis (pLSA) and Latent Dirichlet Allocation (LDA). The topic model is built with implementations of LDA from Gensim and Mallet libraries, adapted for the Russian language by the authors of the method for topic modeling via lda2vec word embeddings [45]. In lda2vec, the pivot word vector and a document vector are added to obtain a context vector. This context vector is then used to predict context words. Similar to LDA, a document vector is decomposed into a document weight vector and a topic matrix. The document weight vector represents the percentage of the different topics, whereas the topic matrix consists of the different topic vectors. A context vector is thus constructed by combining the different topic vectors that occur in a document. The accuracy of topic models is evaluated by coherence and perplexity coefficients.

V. Evaluation

1) Sentiment evaluation and toxic detection: We utilized several metrics to evaluate the performance of models for semantic analysis and toxic messages classification. Let us describe each of them.

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition (1):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

where TP is True Positives, TN True Negatives, FP is False Positives and FN is False Negatives.

However, accuracy is often not employed in cases with unbalanced datasets.

For a more meaningful evaluation in cases with unbalanced datasets, it is more appropriate to utilize "weighted" Precision, Recall, and F1-score. In this case, "weighted" means that each value is calculated as an average between those values for each class scaled with the number of legitimate instances for each class.

Before moving on to the equations for each metric, first, let us list some supplemental equations. For the calculations of precision and recall, we use the following conditional probability equations:

$$P(y_l, \hat{y}_l) = \frac{\mid y_l \cap \hat{y}_l \mid}{\mid y_l \mid} \tag{2}$$

$$R(y_l, \hat{y}_l) = \frac{\mid y_l \cap \hat{y}_l \mid}{\mid \hat{y}_l \mid} \tag{3}$$

$$F1(y_l, \hat{y}_l) = 2 \times \frac{P(y_l, \hat{y}_l) \times R(y_l, \hat{y}_l)}{P(y_l, \hat{y}_l) + R(y_l, \hat{y}_l)} \tag{4}$$

where y_l is the subset of the prediction set with label l, $\hat{y}_l$ is the subsets of the actual set with label l.

The equations for "weighted" metrics:

$$W.Precision = \frac{\Sigma_{l \in L} \mid \hat{y}_l \mid P(y_l, \hat{y}_l)}{\Sigma_{l \in L} \mid \hat{y}_l \mid} \tag{5}$$

$$W.Recall = \frac{\Sigma_{l \in L} \mid \hat{y}_l \mid R(y_l, \hat{y}_l)}{\Sigma_{l \in L} \mid \hat{y}_l \mid} \tag{6}$$

$$F1(y_l, \hat{y}_l) = \frac{\Sigma_{l \in L} \mid \hat{y}_l \mid F1(y_l, \hat{y}_l)}{\Sigma_{l \in L} \mid \hat{y}_l \mid} \tag{7}$$

2) Topic modeling: As an assessment of the efficiency of the algorithms, the following are used - these are coherence and perplexity. A topic is called coherent if the terms most frequent in a given topic, it is no coincidence that they often appear together side by side in the documents. In this work, a measure of coherence was used, which is the log conditional probability (LCP), which estimates the probability of a less frequent word given a more frequent one and is calculated by the formula (8):

$$LCP(t) = \sum_{i=1}^{k-1} \sum_{j=i}^{k} log(\frac{N(w_i, w_j)}{N(w_i)}) \quad (8)$$

where w_i is the i-term in descending order, $N(w)$ – is the number of documents with at least one w, $N(w, w')$ – the number of documents with w, w' close together at least once.

Perplexity is a measure of how well a probabilistic model predicts a topic for a document. It is also used to compare probabilistic models. Low perplexity indicates that the probability distribution is good for predicting samples.

VI. Results

1) Sentiment evaluation: For sentiment analysis, we performed classification into three classes. To tailor 5 classes into 3, classes 1 and 2 with negative comments were combined into one, and similarly classes 4 and 5 with positive comments were combined into one as well. The training dataset was adjusted so that the number of samples in each class is equal. For the described datasets, annotated by sentiment, the results for similar experiments are displayed in Table I. For sentiment analysis, the best results were achieved by multilingual models. All training methods for the Rubtsova datasets resulted in lower performance when tested with the target dataset, even when combined with augmentation. It might be explained by the fact that the dataset in English used for training contains reviews of courses, i.e. is in the same domain as the target dataset, meanwhile, the dataset in Russian contains tweets from discussions of various topics. There is no significantly large dataset in Russian with courses reviews.

TABLE I. Sentiment analysis results

Dataset	Score	SVM	LSTM	ADA
Test ENG	Accuracy	0.6467	0.7000	0.4217
	Weighted Precision	0.6440	0.7033	0.4328
	Weighted Recall	0.6467	0.7000	0.4217
	Weighted-F1	0.6451	0.7010	0.4237
Target dataset	Accuracy	0.6183	0.5379	0.3136
	Weighted Precision	0.5908	0.6774	0.5428
	Weighted Recall	0.6183	0.5379	0.3136
	Weighted-F1	0.5896	0.5573	0.3362

Concluding the results, we can confirm that for the specified problem embeddings, produced by a multilingual model, are sufficient for transferring knowledge from one language to another, while an adversarial domain adaptation model does not perform as well. That might be explained by the difference between volumes of datasets for Russian and English used for training the model, thus the lack of training data from the target domain may result in poor performance. Training with the Russian dataset and testing the model with the target dataset does not show a performance improvement. Summarizing, multilingual embedding models combined with traditional machine learning techniques for small datasets demonstrate the best performance.

2) Toxic detection: For toxic messages detection, we trained classifiers with datasets from Vkontakte and tested them with the target dataset.

All datasets from "VKontakte" were divided into training and validation subsets. All training sets were combined into one "original" set. Table II shows the results of checking the classifier trained on this set, as well as the results of the Perspective API for six validation sets.

TABLE II. Accuracy with the validation sets for the original training dataset and the Perspective API

Classification method	Validation	W.pr.	W.r.	W.F1
SVM + Original train	HABR	0.80	0.88	0.84
	IGM	0.85	0.88	0.85
	MARVEL/DC	0.80	0.84	0.81
	MASH	0.76	0.78	0.72
	SUIAUCTUS	0.85	0.89	0.87
	TV5	0.77	0.81	0.75
Perspective API	HABR	0.93	0. 92	0.89
	IGM	0.92	0.92	0.90
	MARVEL/DC	0.88	0.88	0.84
	MASH	0.84	0.80	0.74
	SUIAUCTUS	0.93	0.92	0.90
	TV5	0.86	0.83	0.78

The Perspective API performs better than our SVM, trained on the original training subset. Next, we were interested in the opportunity to improve the results by increasing the volume of training data for our SVM. We considered the following combinations of training sets:

1) Original + EDA (Orig + EDA)
2) Original + paraphrase (Orig + par)
3) Original + paraphrase + EDA (Orig + par + EDA)
4) Original + paraphrase + EDA + Pseudo Labeling + Translate (Orig + par + EDA + PL + Tr)
5) Original + Pseudo Labeling (Orig + PL)
6) Original + Pseudo Labeling + EDA (Orig + PL + EDA)
7) Original + Pseudo Labeling + paraphrase (Orig + PL + par)
8) Original + Pseudo Labeling + Translate (Orig + PL + Tr)
9) Original + Translate (Orig + Tr)
10) Original + Translate + EDA (Orig + Tr + EDA)
11) Original + Translate + paraphrase (Orig + Tr + par)
12) Pseudo Labeling + EDA (PL + EDA)
13) Pseudo Labeling + paraphrase (PL + par)
14) Pseudo Labeling + paraphrase + EDA (PL + par + EDA)
15) Pseudo Labeling + Translate (PL + Tr)
16) Pseudo Labeling + Translate + EDA (PL + Tr + EDA)

17) Pseudo Labeling + Translate + paraphrase (PL + Tr + par)
18) Translate + EDA (Tr + EDA)
19) Translate + paraphrase (Tr + par)

Table III shows the results of the combinations for which the best results were obtained.

TABLE III. ACCURACY OF VALIDATION DATASETS FOR VARIOUS TRAINING DATASETS

Validation set	Train set	W.pr.	W.r.	W.F1
HABR	PL + EDA	0.94	0.93	0.92
	PL + Tr + par	0.94	0.93	0.92
	Orig + PL	0.92	0.93	0.91
	PL + par	0.92	0.93	0.91
	PL + par + EDA	0.92	0.93	0.91
IGM	Orig + par	0.85	0. 88	0.85
	Orig + EDA	0.83	0.88	0.84
	Orig + par + EDA	0.83	0.88	0.84
	Orig + Tr	0.82	0.87	0.83
	Orig + Tr + EDA	0.82	0.87	0.83
	Orig + Tr + par	0.82	0.87	0.83
	Tr + EDA	0.82	0.87	0.83
	Tr + par	0.82	0.87	0.83
MARVEL/DC	Orig + Tr + EDA	0.87	0.88	0.87
	Orig + Tr + par	0.87	0.88	0.87
	Orig + Tr	0.86	0.88	0.86
	Tr + EDA	0.86	0.88	0.86
	Tr + par	0.86	0.88	0.86
	Orig + par + + EDA + PL + Tr	0.86	0.88	0.86
	PL + Tr	0.86	0.88	0.86
	PL + Tr + EDA	0.86	0.88	0.86
	PL + Tr + par	0.86	0.88	0.86
MASH	Orig + par + EDA	0.83	0.81	0.77
	Orig + PL + EDA	0.78	0.79	0.76
	PL + par + EDA	0.78	0.79	0.76
	Orig + PL + par	0.78	0.79	0.76
	Orig + par	0.79	0.79	0.75
	Orig + EDA	0.77	0.78	0.75
SUIAUCTUS	Orig + Tr	0.94	0.95	0.94
	Orig + par + + EDA + PL + Tr	0.94	0.95	0.94
	PL + Tr	0.94	0.95	0.94
	PL + Tr + EDA	0.94	0.95	0.94
	PL + Tr + par	0.94	0.95	0.94
	Orig + PL + Tr	0.94	0.95	0.94
	Orig + Tr + EDA	0.94	0.95	0.94
	Orig + Tr + par	0.94	0.95	0.94
	Tr + EDA	0.94	0.95	0.94
TV5	PL + par	0,85	0,86	0,85
	Orig + PL	0.84	0.85	0.83
	Orig + PL + par	0.84	0.85	0.83
	PL + EDA	0.82	0.83	0.81
	PL + par + EDA	0.82	0.83	0.81
	Orig + PL + EDA	0.82	0.83	0.81
	PL + Tr + EDA	0.81	0.82	0.81

The results of the experiments show that our hypothesis appears to be correct. The increase in the volume of the training subset allows us to surpass the results of the Perspective API for almost all datasets, except for "IGM" where our model, in the best-case scenario, achieves a 0.05 worse F1-score than Perspective API. In other cases, at least one of the training set variants allows us to surpass or equal the system from Google. The best improvement was achieved by using the original training subset in combination with Pseudo Labeling or Translate. Combining the original training subset with paraphrasing and/or EDA in almost all cases produces a lower F1-score.

The model with the best performance on the validation dataset was selected for testing with the target dataset. The results are shown in Table IV.

TABLE IV. ACCURACY OF TOXIC DETECTION

Dataset	Accuracy	W.pr.	W.r.	W.F1
Target dataset	0.99	0.98	0.97	0.98

3) Toxic Spans Detection: The model for toxic spans detection was trained on the dataset from "SemEval 2021 Task 5: Toxic Spans Detection" challenge with various multilingual embeddings. For validation, we took the dataset from Vkontakte which we annotated with toxic spans. Due to a rather small total number of toxic messages, all of them were combined in a single dataset. The results are shown in Table V.

TABLE V. ACCURACY OF TOXIC SPANS DETECTION

Dataset	Embedding method	W.F1
Train dataset	Bert-uncased	0.67
	Bert-cased	0.70
	xlm-roberta	0.81
validation dataset	Bert-uncased	0.60
	Bert-cased	0.58
	xlm-roberta	0.67

The best performance for the toxic spans detection on the target dataset was achieved by the previously described in the Methods section model combined with multilingual xlm-roberta. The F1-score reached 0.73, level to the results demonstrated by the best performers from the "SemEval 2021 Task 5: Toxic Spans Detection" challenge.

4) Topic modeling: Last, we conducted experiments for building topic models for the target dataset with users' reviews from online education platforms with traditional methods for topic modeling such as LSA, LDA, the LDA-Mallet implementation, and the modification with word embeddings lda2vec. Since the performance of the model depends on the choice of a method and a pre-processing pipeline, we investigated several combinations of stop-word removal, lemmatization, and common phrase detection for each method. The optimal number of topics is determined by the perplexion coefficient for each preprocessing pipeline. We conducted experiments with the number of topics from 2 to 40.

The results are shown in Table VI.

From the obtained results we can highlight LDA Mallet with the best performing preprocessing pipeline "tokenization + lemmatization + stop-word removal". The keywords, 5 topics, and the number of messages for each topic are listed in Table VII.

We obtained 5 different topics related to various aspects of a course. The first topic consists of comments describing

TABLE VI. EVALUATION OF METHODS FOR TOPIC MODELING

Text preprocessing	LDA Coherence/ Perplexity / Optimal topic number	LSA Coherence/ Perplexity / Optimal topic number	LDA Mallet Coherence/ Perplexity / Optimal topic number	LDA2VEC Coherence/ Perplexity / Optimal topic number
Tokenisation	0.48/ -8.24/3	0.30/-7.22/5	0.51/-8.72/3	0.44/-7.92/3
Tokenisation + removal of stop words	0.46/ -9.18/4	0.36/-7.89/4	0.43/-8.58/3	0.47/-8.75/3
Tokenisation + detection of idiomatic phrases	0.46/ -9.18/4	0.36/-7.89/4	0.43/-8.58/3	0.47/-8.75/3
Tokenisation + removal of stop words + detection of idiomatic phrases	0.45/ -9.78/3	0.35/-7.56/4	0.40/-9.75/36	0.40/-8.75/3
Tokenisation + Lemmatization	0.54/-7.18/3	0.30/ -7.21/3	0.54/-6.89/3	0.46/-6.80/3
Tokenisation + Lemmatization + removal of stop words	0.54/-7.18/4	0.32/-7.42/3	0.57/-7.45/4	0.49/-7.44/4
Tokenisation + Lemmatization+ detection of idiomatic phrases	0.53/-7.19/3	0.30/-7.43/3	0.48/-6.82/4	0.45/-6.79/3
Tokenisation + Lemmatization+ detection of idiomatic phrases removal of stop words	0.47/-7.88/3	0.47/-8.20/4	0.55/-7.38/4	0.51/-7.48/3

TABLE VII. DISTRIBUTION OF TOPICS

Keywords	Number of messages
exercise, lecture, example, theory, practice	258
course, knowledge, method, specialization, introduction	399
week, task, programming, video, time	260
learning, machine, maths, algorithm, mathematical	386
useful, big, thank, difficult, like	400

a course structure, the second relates to the general information about the course, the third is about practice, the fourth encapsulates theory, and the fifth contains reviews.

5) Summary: The best performing solutions were implemented in a subsystem for evaluation of emotional background in dialogues and comments for the virtual dialogue assistant. The output of the subsystem also contains summary information about the course topic, as shown in Table VIII.

VII. CONCLUSIONS

Sentiment analysis and toxicity detection is an important aspect of an online-course quality evaluation. It is also important to locate which segments of a course are presented best and which receive negative reviews. Traditional methods for topic modeling such as LSA, LDA, LDA-Mallet implementation, and the modification with word embeddings LDA2VEC are justified by the small number of samples. Experiments show that virtually all methods and preprocessing pipelines separate messages in a modest number of topics, which indicates coherence between users' opinions about elements of a course.

Multilingual embedding models, manual collection and labeling, and augmentation help to handle the lack of training datasets in Russian. The solution to the problem of automatic detection of toxic language with an emphasis on the transfer of knowledge from one language to another proved to provide

TABLE VIII. SENTIMENT AND TOXICITY FOR COURSE TOPICS

keywords	NEG	NETR	POS	TOX	NONTOX
exercise lecture example theory practice	60	49	149	1	257
course knowledge method specialization introduction	33	29	337	4	395
week task programming video time	91	70	99	5	255
learning machine maths algorithm mathematical	121	84	181	3	383
useful big thank difficult like	44	30	326	3	397

improvements to the baseline approach but still showed that multilingual models are far from always be able to classify toxicity equally well for messages in different languages, provided that the model was trained in only one language. In this case, to improve the classification results, it might be reasonable to add messages in the Russian language to the training set, however, there is still the same problem of lack of well-labeled data. Therefore, in the article, we consider the possibility of improving the accuracy of the classification of toxic messages in Russian by increasing the volume of the training dataset. As a baseline, we use the classification

results obtained using the Perspective API from Google, which included the Russian language. The results obtained with the original training set are below the baseline. For this reason, we used various data augmentation methods, such as EDA, paraphrasing, translation of the marked-up English-language dataset into Russian and pseudo labeling of the Russian-language part of the test suite presented by Jigsaw. The experiments conducted with various combinations of training sets not only allowed us to achieve baseline results but also surpass them.

Finally, we achieved a complex solution for evaluating users' opinions about online-courses. This solution can also be employed for other kinds of services since unsupervised machine learning techniques combined with focused datasets and supervised machine learning methods can allow to detect toxic messages and evaluate sentiment in a broad variety of fields.

ACKNOWLEDGMENT

This work was partially financially supported by ITMO University (grant № 620183 "Development of a virtual dialogue assistant for online exams based on Transformer models and natural and mathematical language understanding").

REFERENCES

[1] A. Matveev, O. Makhnytkina, I. Lizunova, T. Vinogradova, A. Chirkovskii, A. Svischev, and N. Mamaev, "A virtual dialogue assistant for conducting remote exams," *Proceedings of the 26th Conference of Open Innovations Association FRUCT*, pp. 284–290, 2020. [Online]. Available: https://doi.org/10.23919/FRUCT48808.2020.9087557

[2] P. Ekman, "An argument for basic emotions," *Cognition and Emotion*, vol. 6(3-4), pp. 284–290, 1992. [Online]. Available: https://doi.org/10.1080/02699939208411068

[3] J. Rišch and R. Krestel, "Toxic comment detection in online discussions," *Deep Learning-Based Approaches for Sentiment Analysis. Algorithms for Intelligent Systems*, pp. 85–109, 2020. [Online]. Available: https://doi.org/10.1007/978-981-15-1216-2_4

[4] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, "Deep learning for hate speech detection in tweets," *WWW '17 Companion: Proceedings of the 26th International Conference on World Wide Web Companion*, pp. 759–760, 2017. [Online]. Available: https://doi.org/10.1145/3041021.3054223

[5] A. A. Kharlamov, A. V. Orekhov, S. S. Bodrunova, and N. S. Lyudkevich, "Social network sentiment analysis and message clustering," *Internet Science. 6th International Conference, INSCI 2019*, pp. 18–31, 2019. [Online]. Available: https://doi.org/10.1007/978-3-030-34770-3_2

[6] A. Dvoynikova, O. Verkholyak, and A. Karpov, "Analytical review of methods for identifying emotions in text data," *CEUR Workshop Proceedings*, vol. 2552, pp. 8–21, 2020.

[7] N. Djuric, J. Zhou, R. Morris, M. Grbovic, V. Radosavljevic, and N. Bhamidipati, "Hate speech detection with comment embeddings," *WWW '15 Companion: Proceedings of the 24th International Conference on World Wide Web*, pp. 29–30, 2015. [Online]. Available: https://doi.org/10.1145/2740908.2742760

[8] S. Zimmerman, U. Kruschwitz, and C. Fox, "Improving hate speech detection with deep learning ensembles," *Language Resources and Evaluation Conference (LREC) 2018 At: Miyazaki, Japan*, 2018. [Online]. Available: https://www.aclweb.org/anthology/L18-1404.pdf

[9] S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, and O. Frieder, "Hate speech detection: Challenges and solutions," *PLoS ONE 14(8): e0221152*, 2019. [Online]. Available: https://doi.org/10.1371/journal.pone.0221152

[10] A. Elnaggar, B. Waltl, I. Glaser, J. Landthaler, E. Scepankova, and F. Matthes, "Stop illegal comments: A multi-task deep learning approach," *AICCC '18: Proceedings of the 2018 Artificial Intelligence and Cloud Computing Conference*, pp. 41–47, 2018. [Online]. Available: https://doi.org/10.1145/3299819.3299845

[11] S. V. Georgakopoulos, S. K. Tasoulis, A. G. Vrahatis, and V. P. Plagianakos, "Convolutional neural networks for toxic comment classification," *Proceedings of the 10th Hellenic Conference on Artificial Intelligence*, 2018. [Online]. Available: https://doi.org/10.1145/3200947.3208069

[12] A. D'Sa, I. Illina, and D. Fohr, "Towards non-toxic landscapes: Automatic toxic comment detection using dnn," *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pp. 21–25, 2020. [Online]. Available: https://www.aclweb.org/anthology/2020.trac-1.4

[13] R. Saia, A. Corriga, R. Mulas, D. Recupero, and S. Carta, "A supervised multiclass multi-label word embeddings approach for toxic comment classification," *11th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (KDIR-2019)*, vol. 1, pp. 105–112, 2019. [Online]. Available: https://www.scitepress.org/Link.aspx?doi=10.5220/0008110901050112

[14] M. Saif, A. Medvedev, M. Medvedev, and T. Atanasova, "Classification of online toxic comments using the logistic regression and neural networks models," *AIP Conference Proceedings*, vol. 2048, 2018. [Online]. Available: https://doi.org/10.1063/1.5082126

[15] S. Shtovba, O. Shtovba, O. Yahymovych, and M. Petrychko, "Impact of the syntactic dependencies in the sentences on the quality of the identification of the toxic comments in the social networks," *Scientific Works of VNTU*, vol. 4, 2019. [Online]. Available: https://doi.org/10.31649/2307-5392-2019-4-35-42

[16] O. Hosam, "Toxic comments identification in arabic social media," *International Journal of Computer Information Systems and Industrial Management Applications. ISSN 2150-7988*, vol. 11, pp. 219–226, 2019. [Online]. Available: http://www.mirlabs.org/ijcisim/regular_papers_2019/IJCISIM_21.pdf

[17] G. Haralabopoulos, I. Anagnostopoulos, and D. McAuley, "Ensemble deep learning for multilabel binary classification of user-generated content," *Algorithms*, vol. 13(4), 2020. [Online]. Available: https://doi.org/10.3390/a13040083

[18] J. Pavlopoulos, P. Malakasiotis, and I. Androutsopoulos, "Deeper attention to abusive user content moderation," *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 1125–1135, 2017. [Online]. Available: https://www.aclweb.org/anthology/D17-1117

[19] M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, and O. Levy, "Spanbert: Improving pre-training by representing and predicting spans," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 64–77, 2020. [Online]. Available: https://www.aclweb.org/anthology/2020.tacl-1.5

[20] O. Makhnytkina, A. Matveev, D. Bogoradnikova, I. Lizunova, A. Maltseva, and N. Shilkina, "Detection of toxic language in short text messages," *A. Karpov, & R. Potapova (Eds.), Speech and Computer: 22nd International Conference, SPECOM 2020, Proceedings. Lecture Notes in Computer Science, vol 12335. Springer, Cham.*, pp. 315–325, 2020. [Online]. Available: https://doi.org/10.1007/978-3-030-60276-5_31

[21] J. Wei and K. Zou, "Eda: Easy data augmentation techniques for boosting performance on text classification tasks," *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019. [Online]. Available: https://www.aclweb.org/anthology/D19-1670

[22] P. K. B. Giridhara, C. Mishra, R. K. M. Venkataramana, S. S. Bukhari, and A. Dengel, "A study of various text augmentation techniques for relation classification in free text," *Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods*, 2019. [Online]. Available: https://doi.org/10.5220/0007311003600367

[23] X. Wu, S. Lv, L. Zang, J. Han, and S. Hu, "Conditional bert contextual augmentation," *International Conference on Computational Science*, pp. 84–95, 2019. [Online]. Available: https://doi.org/10.1007/978-3-030-22747-0_7

[24] K. Varun, C. Ashutosh, and C. Eunah, "Data augmentation using pre-trained transformer models," *Proceedings of the 2nd Workshop on Life-*

long Learning for Spoken Language Systems, pp. 18–26, 2020. [Online]. Available: https://www.aclweb.org/anthology/2020.lifelongnlp-1.3

[25] M. Papadaki, "Data augmentation techniques for legal text analytics," *Department of Computer Science Athens University of Economics and Business*, pp. 1–33, 2017. [Online]. Available: http://nlp.cs.aueb.gr/theses/papadaki_msc_thesis.pdf

[26] J. Risch and R. Krestel, "Aggression identification using deep learning and data augmentation," *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pp. 150–158, 2018. [Online]. Available: https://www.aclweb.org/anthology/W18-4418

[27] B. Banitz, "Machine translation: a critical look at the performance of rule-based and statistical machine translation," *Cad. Tradução*, vol. 40, pp. 54–71, 2020. [Online]. Available: https://doi.org/10.5007/2175-7968.2020v40n1p54

[28] A. López-Pereira, "Neural machine translation and statistical machine translation: Perception and productivity," *Tradumàtica Tecnol. la traducció*, 2019. [Online]. Available: https://doi.org/10.5565/rev/tradumatica.235

[29] X. Wang, Z. Lu, Z. Tu, H. Li, D. Xiong, and M. Zhang, "Neural machine translation advised by statistical machine translation," *AAAI'17: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 3330–3336, 2016.

[30] S. V. Morzhov, "Modern approaches to detect and classify comment toxicity using neural networks," *Modeling and analysis of information systems*, vol. 27(1), pp. 48–61, 2020. [Online]. Available: https://doi.org/10.18255/1818-1015-2020-1-48-61

[31] C. Coulombe, "Text data augmentation made simple by leveraging nlp cloud," 2018, arXiv preprint APIs. [Online]. Available: https://arxiv.org/abs/1812.04718

[32] R. Xiang, E. Chersoni, Y. Long, Q. Lu, and C. Huang, "Lexical data augmentation for text classification in deep learning," *Advances in Artificial Intelligence - 33rd Canadian Conference on Artificial Intelligence, Canadian AI 2020, Ottawa, ON, Canada, May 13-15, 2020, Proceedings*, vol. 12109, pp. 521–527, 2020. [Online]. Available: https://doi.org/10.1007/978-3-030-47358-7_53

[33] W. Wang, B. Li, D. Feng, A. Zhang, and S. Wan, "The ol-dawe model: Tweet polarity sentiment analysis with data augmentation," *IEEE Access*, vol. 8, pp. 40 118–40 128, 2020. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.2976196

[34] A. Nugmanova, I. Chernykh, A. Bulusheva, and Y. Matveev, "Unsupervised training of automatic dialogue systems for customer support,"

[40] A. Kutuzov and E. Kuzmenko, "Webvectors: A toolkit for building web interfaces for vector semantic models," *Ignatov D. et al. (eds) Analysis of Images, Social Networks and Texts. AIST 2016. Communications in Computer and Information Science, vol 661. Springer, Cham*, pp. 155–161, 2017. [Online]. Available: https://doi.org/10.1007/978-3-319-52920-2_15

2019 International Conference "Quality Management, Transport and Information Security, Information Technologies" (ITQMIS), pp. 436–438, 2019.

[35] M. Khovrichev, I. Chernykh, N. Mamaev, and Y. Matveev, "Context-dependent synonym and concept extraction for dialogue systems training," *2019 International Conference "Quality Management, Transport and Information Security, Information Technologies" (IT QM IS)*, pp. 421–423, 2019.

[36] S. Mohammed and S. Al-augby, "Lsa & lda topic modeling classification: Comparison study on e-books," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, pp. 353–362, 2020. [Online]. Available: http://doi.org/10.11591/ijeecs.v19.i1.pp353-362

[37] A. B. Dieng, F. J. Ruiz, and D. M. Blei, "Topic modeling in embedding spaces," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 439–453, 2020. [Online]. Available: https://doi.org/10.1162/tacl_a_00325

[38] K. Kabir, F. Alam, and I. A.B., "Word embeddings for semantic resemblance of substantial text data: A comparative study," *Somani A., Shekhawat R., Mundra A., Srivastava S., Verma V. (eds) Smart Systems and IoT: Innovations in Computing. Smart Innovation, Systems and Technologies, vol 141. Springer, Singapore*, pp. 303–312, 2020. [Online]. Available: https://doi.org/10.1007/978-981-13-8406-6_30

[39] Y. V. Rubtsova, "Constructing a corpus for sentiment classification training," *SOFTWARE & SYSTEMS*, no. 1(109), pp. 72–78, 2015. [Online]. Available: http://swsys.ru/index.php?page=article&id=3962

[41] L. Song, Y. Xu, L. Zhang, B. Du, Q. Zhang, and X. Wang, "Learning from synthetic images via active pseudo-labeling," *IEEE Transactions on Image Processing*, vol. 29, pp. 6452–6465, 2020. [Online]. Available: https://doi.org/10.1109/TIP.2020.2989100

[42] Y. Yao, K. Xu, K. Murasaki, S. Ando, and A. Sagata, "Pseudo-labelling-aided semantic segmentation on sparsely annotated 3d point clouds," *IPSJ Transactions on Computer Vision and Applications (CVA)*, vol. 12, 2, 2020. [Online]. Available: https://doi.org/10.1186/s41074-020-00064-w

[43] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding." *NAACL-HLT.*, 2019.

[44] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8440–8451, 2020. [Online]. Available: https://www.aclweb.org/anthology/2020.acl-main.747

[45] C. Moody, "Mixing dirichlet topic models and word embeddings to make lda2vec," *ArXiv*, vol. abs/1605.02019, 2016.