Detection texts, generated by bots, in social net comments/short posts on Russian

Putilova Elena

December 2023

Abstract

This project presents research of different approaches for task of detection texts on Russian, generated by bots, and proposes relevant solution. Previous researches for this problem for some last years were analysed. Globally now 3 main type of methods exists: text-based, feature-based and graph-based. The dataset is corpus of Russian tweets, marked up binary as 1 for bot-generated text and as 0 for human-generated. Approaches used in that research are text-based: TF-IDF logistic regression and 2 models of transformers. Comparative analysis was done and results presented. For additional details please refer to project code `https://github.com/lenaptv/Detection-of-texts-generated-by-bots`.

1 Introduction

Problem of separation human-generated texts from machine-generated is important now. For example, for business is very important don't waste time for comments/short posts in social nets, generated by bots. And for business is also important do not ignore comments of real people for good publicity. Therefore task to detect text, generated by bots, in social net comments/short posts is really useful. However not so many researches were done till now for that task for Russian domain.

1.1 Team

Elena Putilova prepared this document.

2 Related Work

As was mentioned above, there are not so much researches for task of texts on Russian, generated by bots. Below some relevant researches for some last years were considered, not only for Russian (because such researches in Russian domain is very limited).

Now all methods for detection of bot-generated texts are divided into 3 groups: text-based, feature-based and graph-based.

In November 2023 were published research of V. Gromov and Q. N. Dang [1]. Scientists propose to use two approaches for bot-detection in russian texts (both are text-based): clustering (special variant) and entropy-complexity plane. Results are impressive: accuracy from 0.879 to 0.992. However dataset for human-generated texts was collected form Russian literature. But usually human in social networks do not speak literary. And real human tweets (not texts from books) is harded to distinguish from bot-generated.

In April 2023 Yuhan Liu and group of scientists [3] proposed impressive approach for detection of bot-generated texts: Community-Aware Mixtures of Modal-Specific Experts. Actually it is mix of all methods (graph-based, feature-based and text-based). And, of course, dataset is not Russian.

In February 2023 group of scientists from University of Washington and University of Virginia [5] published results of research for bot-detection on combination of text-based and graph-based methods. Approach is named BIC: Twitter **B**ot detection framework with text-graph **I**nteraction and semantic **C**onsistency. Datasets used in research on English.

In 2021 Andres Garcia-Silva and others published research with text-based approach for bot-detection [4]. Used in that research method was finetuning transformers. F1-score were from baseline 0.80 to 0.86 for best result. Actually these results are very close to results of these project (please refer to section "Results"). However, of course, transformers were finetuned not for Russian.

In 2020 Ilia Karpov and Ekaterina Glazkova [2] proposed for bot-detection used Graph Embedding approach in combination with machine algorithms for classifications. Dataset used in this research was obtained from real russian social network VKontakte. However actually many other features were used for that approach (for example, personal characteristics from account data), not only text data. Therefore it is mix of feature-based and graph-based approaches.

In 2014 G.V. Ovchinnikov and others published research of using graph-based method for russian tweet datasets [6].

So, for some last years there were very limited researches for solving problem to detection of texts on Russian, generated by bots. Therefore this project is actual.

3 Model Description

Actually task of detection of bot-generated text we can consider as task of binary classification. Also I have selected text-based approach for that task, because now we have good enough and open-source transfromers (BERT-type), which are able to give high score for text classification task.

Currently SOTA in text classification tasks are encoder-only transformers. Therefore for my project I have used this type models.

The baseline is required because there is no previous art on the problem of bot-detection in texts on Russian in social networks. For baseline I have selected TF-IDF logistic regression, because it is simple method. I have used Tf-idf vectorizer and logistic regression algorithm from sklearn library.

After that I have finetuned two pre-trained on Russian texts BERT-type transformers: rubert-tiny2 and ruBert-base.

Rubert-tiny2 is comparatively small (not above 250 Mb), therefore fast for finetuning [1].

I have finetuned rubert-tiny2 on both types of training set: "initial" and augmented (see details of augmentation approach in "Dataset" section).

RuBert-base is bigger than rubert-tiny2 (above 700 Mb), therefore not so fast for finetuning [2], but usually give good score of metrics for classification tasks.

4 Dataset

To collect corpus with Russian human-generated and bot-generated comments/short posts in social networks I have decided to use corpus with tweets on English [3]. This corpus was translated on Russian and cleared (hereinafter "corpus 1"). As result I

After that I have decided to extent this dataset by corpus of Russian tweets human-generated [4]. It was done in order to add to corpus of real Russian language, used of social network's (hereinafter "corpus 2"). I didn't extent dataset based on Russian literature, because people speak in social networks not literary.

I have cleared corpus 2 from doubtful (bot or human-generated) comments, and as result obtained 11,5 thousands Russian human-generated tweets. After that I have added 5 thousands human-generated tweets translated from corpus 1. And finally for balanced classes I have added 16,5 thousands bot-generated tweets translated from corpus 1.

As result final corpus of tweets on Russian, generated by bots (16,5 thousands tweets labelled 1) and generated by human (16,5 thousands tweets labelled 0) has the structure presented on Fig. 1. As you see no disbalance of classes. It is very important for classification task.

Some examples for each classes is presented on Fig. 2. As you see human-generated comments are more emotional, sometimes with humor and more logical, than samples, generated by bots.

Within experimental stage I have decided to do augmentation for train dataset. For text augmentation I have selected the method, proposed by Franco M. Luque in 2019 [7]. The method is simple. Shortly this method: we take 2 samples of one class A and B. After that we divide both A and B into 2 halfes and mix these halfes. So as result we obtain A-augmented=A1+B2 and B-augmented=B1+A2.

[1] Link to the website, where the tokenizer and bins can be downloaded: here

[2] Link to the website, where the tokenizer and bins can be downloaded: here

[3] Link to the website, where the tweets corpus on English could be downloaded: here

[4] Link to the website, where the tweets corpus on Russian could be downloaded: here

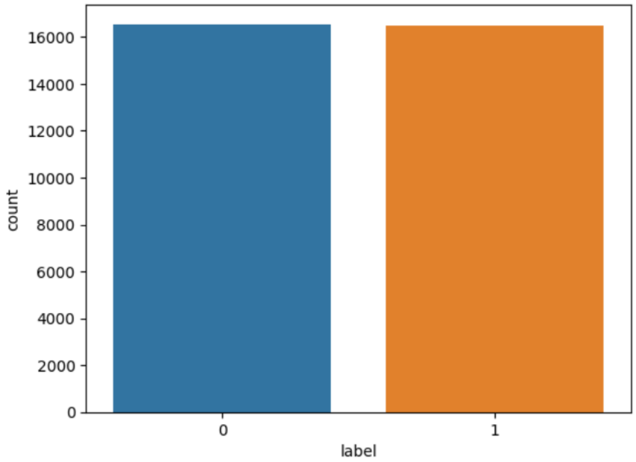
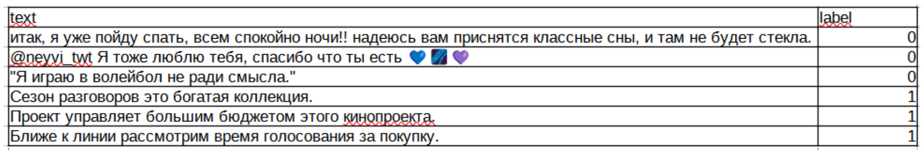


Figure 1: The classes disritbution.



text	label
итак, я уже пойду спать, всем спокойно ночи!! надеюсь вам приснятся классные сны, и там не будет стекла.	0
@neyvi_twt Я тоже люблю тебя, спасибо что ты есть 💙 🌌 💜	0
"Я играю в волейбол не ради смысла."	0
Сезон разговоров это богатая коллекция.	1
Проект управляет большим бюджетом этого кинопроекта.	1
Ближе к линии рассмотрим время голосования за покупку.	1

Figure 2: The classes disritbution.

On the Tab. 1 you can see the statistics for the mentioned dataset.

	Train	Train-augmented	Test
comments	29744	52764	3305

Table 1: The distribution of samples of train, augmented train and test sets.

5 Experiments

5.1 Metrics

F1-score was selected as main metric for that task. Because actually task is text classification. And because for business important both metrics: Precision and Recall. And F1-score is the harmonic mean for Precision and Recall:

$$F1score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Precision is division of true positive outcomes (TP) by total of TP and false positive (FP):

$$Precision = \frac{TP}{TP + FP}$$

Recall is division of TP outcomes by total of TP and false negative (FN):

$$Precision = \frac{TP}{TP + FN}$$

Precision and Recall are both important for business because, in one hand, it is bad to waste time for bot's comments, but, in another hand, business should not ignore comments of real people for good publicity.

5.2 Experiment Setup

5.2.1 Hyper-parameters for TF-IDF logistic regression:

- stop words for Russian from nltk libriary
- n-jobs = -1
- C = 50

5.2.2 Hyper-parameters for transformers:

- batch size = 16
- epoch of epoch = 5
- learning rate = 2e-6
- optimizer = AdamW()
- loss function = CrossEntropyLoss()

5.3 Baseline

Logistic regression over TF-IDF embedding for text classification was selected as the baseline, because it is simple approach.

6 Results

The classification report on test dataset for Logistic Regression is presented on Fig. 3.

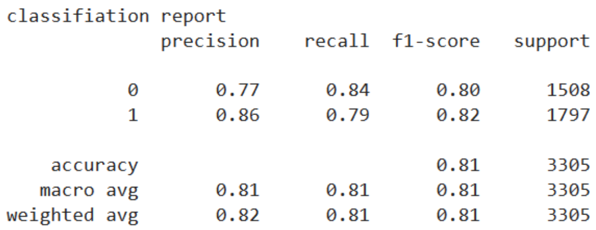
The classification report on test dataset for transformer rubert-tiny2 is presented on Fig. 4.

The classification report on test dataset for transformer rubert-tiny2, finetuned on augmented train dataset, is presented on Fig. 5.

The classification report on test dataset for transformer ruBert-base is presented on Fig. 6.

On the Tab. 2 you can see the final F1-score for all models in experiments.

The best score gave finetuned ruBert-base.

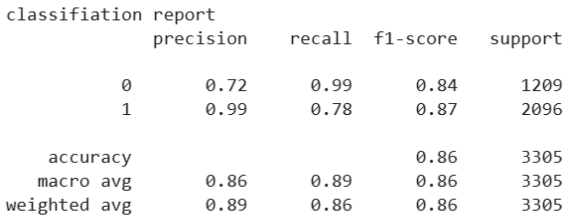


```
classifiation report
              precision    recall  f1-score   support

           0       0.77      0.84      0.80      1508
           1       0.86      0.79      0.82      1797

    accuracy                           0.81      3305
   macro avg       0.81      0.81      0.81      3305
weighted avg       0.82      0.81      0.81      3305
```

Figure 3: The classification report on test dataset for Logistic Regression

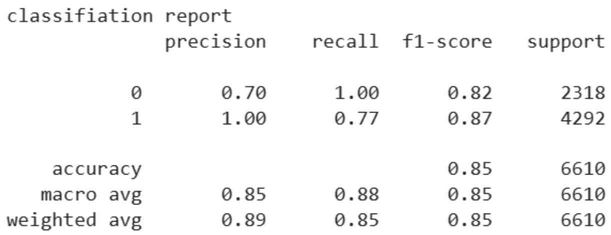


```
classifiation report
              precision    recall  f1-score   support

           0       0.72      0.99      0.84      1209
           1       0.99      0.78      0.87      2096

    accuracy                           0.86      3305
   macro avg       0.86      0.89      0.86      3305
weighted avg       0.89      0.86      0.86      3305
```

Figure 4: The classification report on test dataset for transformer rubert-tiny2

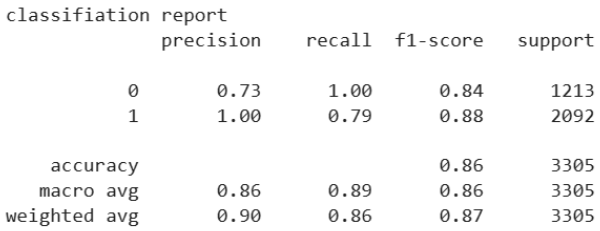


```
classifiation report
              precision    recall  f1-score   support

           0       0.70      1.00      0.82      2318
           1       1.00      0.77      0.87      4292

    accuracy                           0.85      6610
   macro avg       0.85      0.88      0.85      6610
weighted avg       0.89      0.85      0.85      6610
```

Figure 5: The classification report on test dataset for transformer rubert-tiny2, finetuned on augmented train dataset



```
classifiation report
              precision    recall  f1-score   support

           0       0.73      1.00      0.84      1213
           1       1.00      0.79      0.88      2092

    accuracy                           0.86      3305
   macro avg       0.86      0.89      0.86      3305
weighted avg       0.90      0.86      0.87      3305
```

Figure 6: The classification report on test dataset for transformer ruBert-base

Model	F1-score
LogReg	0.81
rubert-tiny2, finetuned on augmented dataset	0.85
rubert-tiny2	0.86
ruBert-base	0.87

Table 2: The final F1-score for models in experiments.

Score of rubert-tiny2, finetuned on intial training set, is close to the best score, dispite the fact, that rubert-tiny2 significantly smaller than ruBert-base, and therefore finetuning was faster.

Unfortunately, text-augmentation was useless - probably used method (described in section "Dataset") made human-generated text less logical and therefore more liked as bot-generated.

Also we see on final score TF-IDF logisitc regression give good result for such base method (fast, simple, no GPU is required).

Important fact, that the range of scores of used method of this project is very close to range of scores obtained in analogical project for English language [4]: 0.80 - 0.86 is the range of scores in research [4]. So it signs, that research was conducted with good quality.

7 Conclusion

This project proposes a solution for task of detection of bot-generated texts in social networks on Russian. Dataset was collected, using both translated binary labelled English dataset of tweets and Russian dataset of human-generated tweets (as corpus real Russian "social networks language"). The task was managed by text-based methods: TF-IDF logistic regression and two types of transformers (both are encoder-only). Comparative analysis was done: the best result is finetuned transformer ruBert-base.

So, this project result is first open-source model for detection of bot-generated texts in social networks on Russian, finetuned on Russian tweets dataset.

References

[1] Vasilii Gromov, Quynh Nhu Dang *Spot the Bot: Distinguishing Human-Written and Bot-Generated Texts Using Clustering and Information Theory Techniques* arXiv:2311.11441v1 [cs.CL] 19 Nov 2023

[2] Ilia Karpov, Ekaterina Glazkova *Detecting Automatically Managed Accounts in Online Social Networks: Graph Embedding Approach* arXiv:2010.07923

[3] Yuhan Liu and others *BotMoE: Twitter Bot Detection with Community-Aware Mixtures of Modal-Specific Experts* arXiv:2304.06280

[4] Andres Garcia-Silva and others *Understanding Transformers for Bot Detection in Twitter* arXiv:2104.06182v1 [cs.CL] 13 Apr 2021

[5] Anant Shukla and others *Social Media Bot Detection using Dropout-GAN* arXiv:2311.05079v1 [cs.LG] 9 Nov 2023

[6] Ovchinnikov and others *Algebraic reputation model RepRank and its application to spambot detection* arXiv:1411.5995v1 [cs.SI] 20 Nov 2014

[7] Franco M. Luque *Atalaya at TASS 2019: Data Augmentation and Robust Embeddings for Sentiment Analysis* arXiv:1909.11241v1