Building language model according to users' reviews about goods and using model in the context advertising

Alexandre Kalendarev
Georgy Chernov

December 2020

Abstract

This document will provide you with guidelines for your project final report. You will learn how to structure the report and present your results. Please provide a link to your project code right here: https://github.com/akalend/mobile_nlp_analisys.

1. Introduction

Modern systems of context advertisement do not always use analysis of user's discussions and statements. If we analyze user's expressions in the social networks, we can get an idea of their main topic, understand their attitude to some goods and then use all this information in the advertisement of some products and services. For example, user made a statement: "my mobile phone's battery is poor". Our system recognizes this comment's purpose: it is about a mobile phone, and a user is not satisfied with it. In this case, social network's advertisement algorithms can advise the user to purchase a new gadget. This system would increase CPA (click-per-action) rates and its conversion (a relation between advertisement cost and its revenue). As the result, it can help us gain some investors and advertisers, which will make the overall income grow.

1.1 Team

This project was completed groups by ***Alexandre Kalendarev*** and ***Georgy Chernov.***
The responsibilities in the project:

Alexandre Kalendarev (github @akalend): common concept, design parser, dataset collecting and cleaning, toxic model and summarization model

Georgy Chernov (github @rehcoeg): attention and topic modeling, summarization model

Related Work

This research consists from complex two parts: Attention mechanism and toxic detection. The Attention is one of the most influential ideas in the Deep Learning community. The base idea was description in the article «An Introductory Survey on Attention Mechanisms in NLP Problems» of Dichao Hu. Thus, in article «Attention in Natural Language Processing» more detailed description of the Attention mechanism.

The Recurrent Neural Networks, ones capable to handle the sequence of inputs, empowered with the Long Short Term Memory [Hochreiter et al., 1997] [Mikolov et al., 2015], provided powers for machine understanding of the textual sequence by utilizing the hidden state of the Encoder. Many of following works, for example [Zhou et al., 2016] and [Gopalakrishnan et al., 2020], demonstrated the capability to perform the Sentiment Analysis using the LSTM Encoder.

Several researchers have been researching various topics that will make the system more tolerant of different methodologies used to evade the filters. In some earlier works, n-grams with parts of speech were used in an SVM model to classify the language sentiment (Warner and Hirschberg, 2012). In 2016 (Nobata et al., 2016) followed a similar methodology to use linguistic features to train a Machine Learning algorithm that will detect 'Hate Speech.' [5]. Authors at (Alorainy et al., 2018)'s researched hate speech by identifying words that contribute to hate speech. They designed a custom pronoun lexicon and semantic relationships to capture the linguistic differences when describing the messages in the in-and out-, and trained word embedding model on that data.

Therefore, the toxic detection in Russian social networks was study in publication J. Rubtsova «Development and research of subject-independent classifiers of texts by sentiment».

3. Model Description

The basic idea consist from composition of two part: The first part is understanding of text based on attention mechanism. If attention is mobile devices or other goods from adversting, the second part has been to work. The second part is toxic binary classification. After classification, if respondent недоволен товаром, то the adv system show the new information about mobile device.

The dataset was classified with help of the pre-trained 'bert-base-uncased' model implemented with PyTorch. In our case, there is a problem of binary classification: we need to distinguish some comments concerning mobile phones from those, which are not.

1. Firstly, we preprocess and prepare our dataset it with BertTokenizer.
2. Secondly, Adam optimizer and a suitable learning rate are used to tune BERT for 10 epochs.
3. We use BinaryCrossEntropy as the loss function since the detection of mobile phone comments is a two-class problem. Then we make sure that we use Sigmoid, passing output through it before calculating the loss between the target and itself.
4. After training, we can plot a diagram with loss, which was received on both train and validation data.
5. At the end, we evaluate the results achieved by our model using F1-score. We can also print out the confusion matrix to see how much data our model predicts correctly and incorrectly for each class.

   Though we were not able to implement this model on our dataset, according to the similar problem solved with it, we can conclude that this model is efficient scoring more than 90% accuracy even on a relatively small dataset.

Following model was used as the base for this experiment: https://huggingface.co/bert-base-uncased

In both cases, we applied the following preprocessing techniques: replacing URLs and usernames with keywords, removing punctuation marks, and converting strings into lowercase. The first one was Multinomial Naive Bayes (MNB), which tended to perform well in the text classification task [14]. To build the MNB model, we used the Bag-of-Words model and the TF-IDF vectorization. The second one was Bidirectional Long Short-Term Memory (BiLSTM) neural network, which demonstrated high classification scores in recent sentiment analysis studies. For the embeddings layer of the neural network, we pre-trained Word2Vec embeddings (*dim* = 300) [15] on the collection of Russian language tweets from RuTweetCorp.

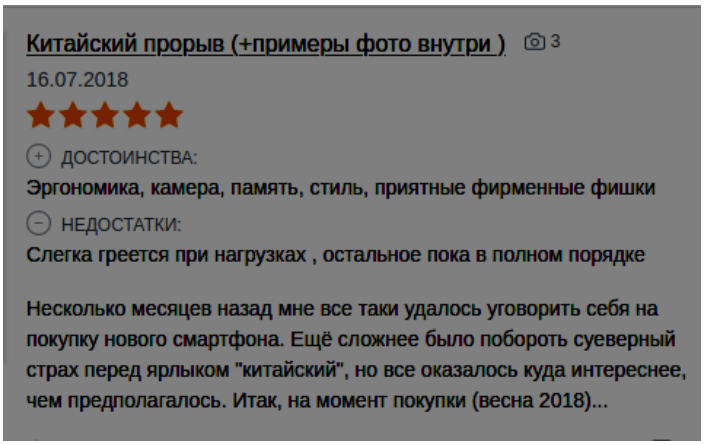
On the top of the Word2Vec embeddings, we added two stacked Bidirectional LSTM layers. Next, we added a hidden fully connected layer and sigmoid output layer. To reduce overfitting, regularization layers with Gaussian noise and Dropout layers were also added to the neural network. We used Adam optimizer with the initial learning rate of 0.001 and categorical binary cross-entropy as a loss function. We trained our network with frozen embeddings for the 10 epochs.

Based on the BERT, which supports 104 languages, including Russian, with 12 stacked Transformer blocks, a hidden size of 768, 12 self-attention heads, and 110M parameters in general. The fine-tuning stage was performed with the recommended parameters from the paper [16] and the official repository: a number of three train epochs, a number of warmup steps of 10%, a max sequence length of 128, a batch size of 32, and a learning rate of 5e-5.

4. Dataset

The corpus of words was collected from ***Отзовик (***https://otzovik.com) website in the mobile phone category.

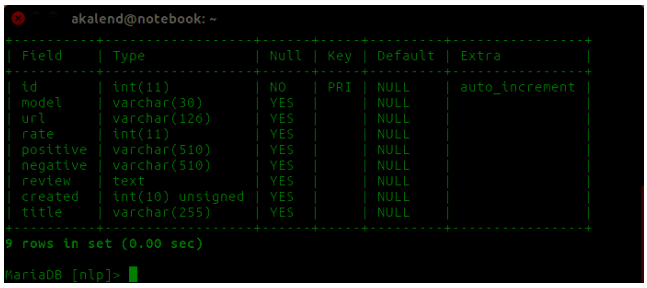
You can see example of one review on the picture 1.



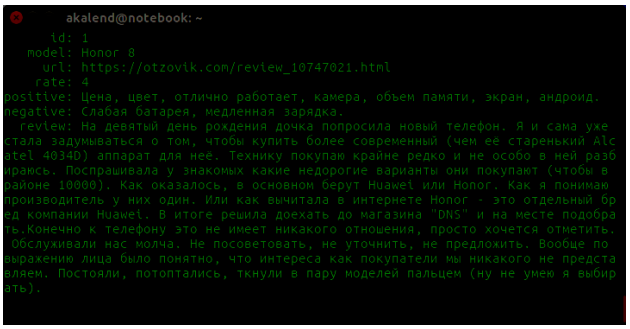
Picture 1

The database have the fields:
- Title
- The rating (count of stars)
- Advantages of the product
- Disadvantages of the product
- User’s review

The structure of data table you can see on the picture 2 and example of one row showed tail on the picture 3:



Picture 2



Picture 3

The corpus was created from 2589 reviews, scrapped from the website and saves in database.

Negative reviews were filtered, and positive phrases such as "no", "no disadvantages", "didn't find any", "didn't encounter any", "all good", "no disadvantages", etc. were removed from them.

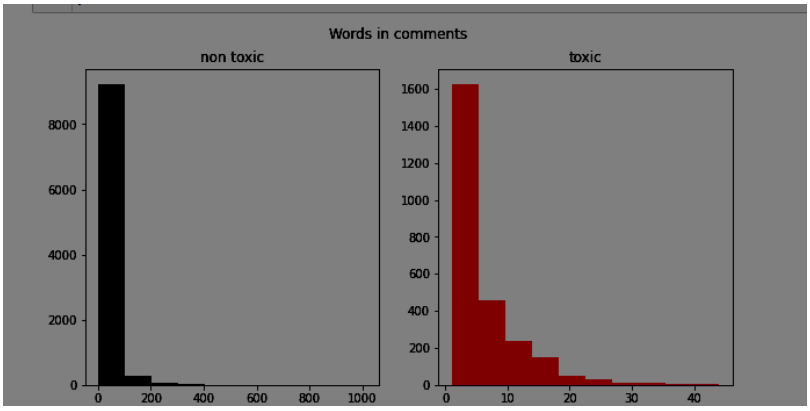
Since the developed detector is designed to determine the user's dissatisfaction with the item, in order to further advertise another item to him, the data was filtered according to the economic component. Word combinations containing the strings: "price", "expensive", "too expensive", "does not match the price", "overpriced", etc.

Further, words with negation were replaced by tokens: for example, "does not hang" was replaced by token “does_not_hang”, because in Russian the separate word "hang" has the opposite meaning together with the particle "not" and the word "hang" itself, coming one after another.

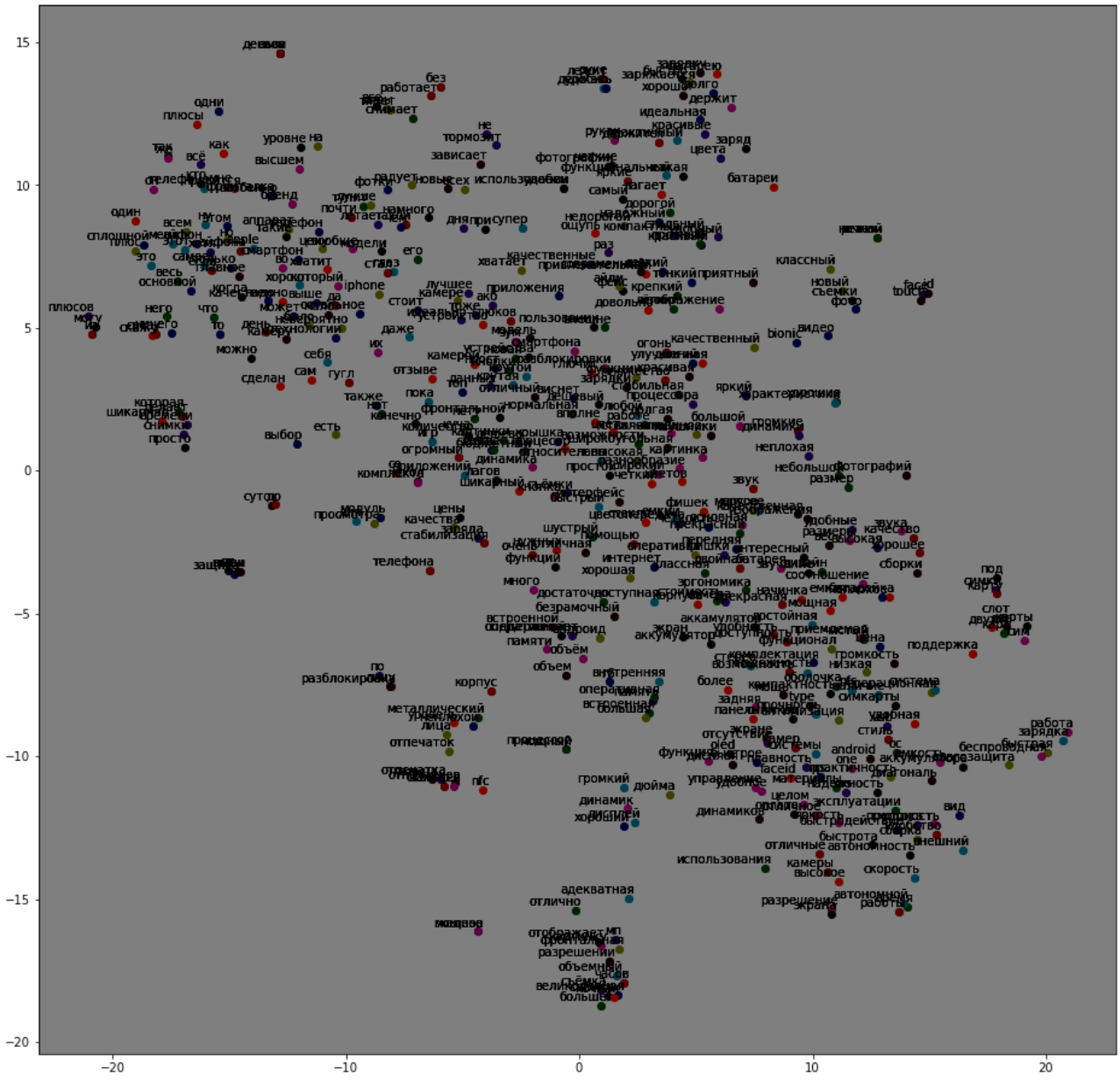
The data from «review» field with rating less 3 and less was included into dataset.

The positives sentences was gotten from Kaggle dataset [17] and dataset from Russian toxic corpus [10].

For train of Topic modelling used complex dataset from users’ review field, marked as 1, and sentences from twitter [10], marked as 0. The distribution of words in the dataset is shown on the graph:



When building word2wec we consider the following graph:



Negative reviews were filtered out, some of them were removed. When analyzing the vector of words, we found an interesting fact: that the probability of occurrence of the following words next to the word "inexpensive" is relatively high :
- lags
- glitches
- lags
- hangs

But on the other hand it was quite predictable for cheap phone models

Binary classification was used to recognize the type of goods. Training of the neural network was performed on a prefabricated dataset - one part of which had no features and was taken from Twitter [10], and the second part, the marked part, were comments on product reviews collected from the site Otzovik.

5. Experiments

The experiment tested the mechanism of attention and determination of toxicity, in relation to the object of attention.

We tested and evaluated correlation between the occurrences of some words related to the goods' qualities and the users' attitude towards those goods. We also tested how good some NLP models perform on the tasks of binary classification according to the given context.

At the end, we concluded, that toxicity testing using BERT on the binary classification gave better results.

6. Results

We collected corpus of reviews on mobile phones.

To do this, we performed annotation and validation of Russian Language Toxic Comments Dataset. Next, we built classification models by exploring transfer learning of Bidirectional Encoder Representations from Transformers (BERT). The top-performing model BERT-Toxic achieved F_1 = 82.20% in a binary classification task.

References

1. Sepp Hochreiter. Long Short-Term Memory 1997

2. A Roadmap towards Machine Intelligence. Tomas Mikolov, Armand Joulin, Marco Baroni. 2015

3. Betty van Aken, Julian Risch, R.K., Loser, A.: Challenges for toxic comment classification:an in-depth error analysis

4. Bertie Vidgen, Dong Nguyen, R.T.A.H.S.H.H.M.: Challenges and frontiers in abusive content detection

5. Brassard-Gourdeau, E., Khoury, R.: Impact of sentiment detection to recognize toxic and subversive online comments (Dec 4, 2018)

6. Chikashi Nobata, Learning algorithm that will detect ’Hate Speech. [5]. Authors at (Alorainy et.

al., 2018)’s researched hate speech by identifying words that contribute to hate speech.

7. Hu, A.K.K.: Toxic speech detection

8. PhD, Patrick Adigun: Identification and classification of toxic comments on social media using machine-learning techniques

9. Serhiy Shtovba, Olena Shtovba, M.P.: Detection of social network toxic comments with usage of syntactic dependencies in the sentences

10. Julia Rubtsova Разработка и исследование предметно независимого классификаторв текстов по тональности 2014

11. Julia Rubtsova. Автоматическое построение и анализ корпуса коротких текстов (постов микроблогов) для задачи разработки и тренировки тонового классификатора. 2012

12. Dichao Hu. An Introductory Survey on Attention Mechanisms in NLP Problems

13 Andrea Galassi , Marco Lippi , and Paolo Torroni. Attention in Natural Language Processing

14. *Frank, E., Bouckaert, R.:* Naive bayes for text classification with unbalanced classes. In: Fürnkranz, J. et al. (eds.) Knowledge discovery in databases: PKDD 2006. pp. 503–510. Springer Berlin Heidelberg, Berlin, Heidelberg (2006).

15. Mikolov, T. et al.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th international conference on neural information processing systems—volume 2. pp. 3111–3119. Curran Associates Inc., Red Hook, NY, USA (2013).

16. Sun, C. et al.: How to fine-tune bert for text classification? In: Sun, M. et al. (eds.) Chinese computational linguistics. pp. 194–206. Springer International Publishing, Cham (2019).

17. Russian Language Toxic Comments https://www.kaggle.com/blackmoon/russian-language-toxic-comments