Lightning Talk - Humor Recognition in Russian Language

Valeriia Baranova-Bolotova
Ural Federal University
Yekaterinburg, Russian Federation
lurunchik@gmail.com

Vladislav Blinov
Ural Federal University
Yekaterinburg, Russian Federation
vladislav.blinov@urfu.ru

Pavel Braslavski
Ural Federal University
Yekaterinburg, Russian Federation
pbras@yandex.ru

ABSTRACT

In this lighting talk paper, we present a dataset of jokes in Russian and deep learning model for solving humor recognition task. The new large dataset was collected from various online resources and complemented carefully with unfunny texts with similar lexical properties. In total, there are more than 300,000 short texts, which is significantly larger than any previous humor-related corpus. Manual annotation of 2,000 items proved the reliability of corpus construction approach. Further, we applied language model fine-tuning for text classification and obtained an F1 score of 0.91, which constitutes a considerable gain over baseline methods.

KEYWORDS

natural language processing; humor recognition; neural networks; language modelling

ACM Reference Format:
Valeriia Baranova-Bolotova, Vladislav Blinov, and Pavel Braslavski. 2019. Lightning Talk - Humor Recognition in Russian Language. In *Companion Proceedings of the 2019 World Wide Web Conference (WWW '19 Companion), May 13–17, 2019, San Francisco, CA, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3308560.3316473

1 DATA

Our goal was to expand Stierlitz dataset of Russian jokes with jokes and non-jokes lexically similar to them.

Firstly, we collected more than 1M jokes from multiple humorous public pages from the social network *VK*[1] and from the website *anekdot.ru*[2]. Further, we downloaded 10M posts from a large online forum *E1.ru*[3], and indexed them with Elastic[4] and returned the first post with Jaccard similarity less than 0.4 to the collection from the BM25-ranked list of matching forum posts for each existing joke. In total, we had 314,269 jokes with their non-jokes pairs. We also used Puns [1] dataset of 213 puns for evaluation.

To verify our automatically created collection, we conducted an evaluation using an online interface, where 2,000 random jokes and non-jokes were assessed on a 3-point scale: 'not a joke', 'an unfunny joke' and 'a joke'. More than 100 volunteers took part in evaluation and 1,877 examples were labeled by at least three assessors. Majority voting resulted in 94% of non-jokes marked as 'not a joke' and 95% of jokes marked as either 'an unfunny joke' or 'a joke', which demonstrates a good performance of the automatic procedure.[5]

Table 1: Detection quality – F1 scores on Stierlitz, Fun, Gold, and recall on jokes-only Puns dataset

Model	Stierlitz Test	Fun Test	Gold	Puns
Trained on Stierlitz train				
Baseline SVM	0.91	0.71	0.64	0.73
Stierlitz SVM [1]	0.88	0.74	0.64	0.7
ULMFun	**0.97**	0.77	0.66	**0.92**
Trained on Fun train				
Baseline SVM	0.8	0.8	0.8	0.43
ULMFun	0.92	**0.91**	0.88	**0.92**

2 CLASSIFICATION METHODS

Initially, we pretrained a language model on 10M online forum texts for 15 epochs with architecture and parameters directly transferred from [2]. Texts were tokenized using unigram subword tokenization method implemented in SentencePiece library [3] with the vocabulary size of 100,000. Further, we used either Stierlitz or Fun dataset to fine-tune the model for five epochs. Finally, we replaced the target task with humor classifier by augmenting the model, further referred to as ULMFun, with linear blocks and trained the model with gradual unfreezing followed by 14 consecutive epochs. We chose linear SVM classifier on top of *tf.idf* features as a baseline.

3 RESULTS

The goals of the experiment were to estimate the impact of the increased dataset size and its construction methods; to introduce a strong baseline based on deep neural network approach, to compare it with a baseline and published work, as well as to evaluate generalization abilities of the model. In the first series of experiments we trained a simple SVM baseline on *tf.idf* features and ULMFun model on Stierlitz train set. We tested the obtained models on held-out test sets of Stierlitz and Fun, as well as on smaller manually annotated Gold and Puns collections. In addition, we were able to apply the best model from [1] to the test data. In the second series, we trained simple baseline and ULMFun on a larger Fun training set and evaluated these two models on the same test data as in the previous stage. Table 1 summarizes the results. ULMFun significantly outperforms Stierlitz SVM and our baseline. The lower part of Table 1 shows results of baseline model and ULMFun

[1] http://vk.com/ - the largest Russian social network
[2] https://www.anekdot.ru/ - the oldest website of jokes on the Russian Web
[3] https://www.e1.ru/talk/forum/
[4] https://www.elastic.co/

[5] For example, manual verification of one-liners dataset [4] revealed 9% of noise.

WWW '19 Companion, May 13–17, 2019, San Francisco, CA, USA

ACM ISBN 978-1-4503-6675-5/19/05.
https://doi.org/10.1145/3308560.3316473

trained on a larger Fun training set. As expected, more data significantly improves classification quality on Fun test set in case of both methods.

REFERENCES

[1] Anton Ermilov, Natasha Murashkina, Valeria Goryacheva, and Pavel Braslavski. 2018. Stierlitz Meets SVM: Humor Detection in Russian. In *Artificial Intelligence and Natural Language*. 178–184.

[2] Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, Vol. 1. 328–339.

[3] Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. 66–71.

[4] Rada Mihalcea and Carlo Strapparava. 2005. Making computers laugh: Investigations in automatic humor recognition. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*. 531–538.