Which IR model has a better sense of humor?

Valeria Bolotova Vladislav Blinov Kirill Mishchenko Pavel Braslavski
Ural Federal University

ABSTRACT

This paper describes experiments on humorous response generation for short text conversations. Firstly, we compiled a collection of 63,000 jokes from online social networks (VK[1] and Twitter[2]). Secondly, we implemented several context-aware joke retrieval models: BM25 as a baseline, query term reweighting, word2vec-based model, and learning-to-rank approach with multiple features. Finally, we evaluated these models in two ways: on the community question answering platform Otvety@Mail.ru[3] and in laboratory settings. The evaluation that an information retrieval(IR) approach to humorous response generation yields satisfactory performance. The collection, test questions, evaluation protocol, and assessors' judgments create a ground for future research.

KEYWORDS

computational humor, dialog systems, information retrieval approach, natural language processing

1 INTRODUCTION

Following recent trends in the widespread use of dialog systems like Apple Siri, Google Now and others, it becomes important to incorporate sense of humor into them. Humorous responses can help to deal with out-of-domain queries as well as make dialog systems more human-like. The aim of our study is to examine the effectiveness of information retrieval approach (a sub-field of computer science that deals with the automated storage and retrieval of documents) to humorous response generation.

2 METHODS AND RESULTS

2.1 Retrieval Models

As a baseline model we chose BM25 [2] scoring, which is based on textual similarity between queries and documents. Stimuli in this model are mapped to lemmatized bag-of-words representations without stop words and then are queried against an inverted index.

Query Term Reweighting (QTR) The proposed approach follows the idea of "humor anchors" introduced in [6]. To figure out what kinds of words are important for comic effect, we analyzed which morphological tags appear frequently in both questions and corresponding answers. Based on the acquired data, we composed a set of rules to adjust weights of anchor words using empirically derived boosting weights. These rules were applied to every stimulus before using BM25 weighting. All non-anchor words were excluded, and tf-idf weights of anchor words were multiplied by the corresponding boost values.

[1]https://vk.com/
[2]https://twitter.com/
[3]https://otvet.mail.ru/humor/

womENcourage 2017, September 2017, Barcelona, Spain

Word2vec-Based Document Embeddings The word2vec [4] method is a way to obtain word vectors such that semantically similar words have close vectors in terms of cosine similarity. Specifically, we used a word2vec model trained on a Russian news corpus and provided by the service *RusVectores* [3]. We precalculated document vectors for our joke collection, and then, given a stimulus, we calculated its vector and found the closest jokes in terms of cosine similarity between vectors.

Learning-to-Rank (LETOR) Analogously to [5], we used a learning-to-rank algorithm with a diverse set of features to re-rank responses of other models. In particular, we built a pool of answer candidates using top-50 answers returned by the BM25, QTR, and word2vec-based models described above. We used RankLib implementation of RankBoost algorithm to obtain a ranking function. The algorithm was trained on the humorous question-answer(Q&A) pairs, employing the set of semantic features for a question-answer pair.

2.2 Evaluation

We evaluated the models in two ways: in the Humor category of the Otvety@Mail.ru Q&A platform and in laboratory settings. Tables 1 and 2 provides the results respectively.

In the first case we automatically posted top-1 ranked responses of each model for randomly sampled questions from humor category during four days and gathered user reactions after a week. In total, 267 questions were answered.

In laboratory settings assessors evaluated top-3 results for each model on the 80 questions from the Q&A platform on a four-point scale (from 0 to 3, with the corresponding emotions in the evaluation interface). We used pooling, each model was evaluated by four people independently. We also calculated Cohen's kappa [1] as a measure of inter-annotator agreement. Averaged pairwise kappa statistics for four assessors in our experiments is 0.21.

3 CONCLUSION

The results of the evaluation on the Q&A platform show that the learning-to-rank approach provides the best performance. Moreover, it's answers on average have more likes than around 17% of user answers on the Otvety@Mail.ru. However, QTR approach has the biggest amount of "best answers". The word2vec-based approach has comparable performance as well. The most surprising aspect of the manual evaluation is that the LETOR method shows the lowest value in both top-1 and DCG@3 metrics. The one of the explanations for this could be the low inter-annotator agreement. The findings suggest that information retrieval approach is a promising direction in humorous response generation. It is also clear that morphological and word2vec-based features are effective for the task. Nevertheless, the results of the "oracle" model indicate that there is an abundant room for the improvement of the answer ranking.

Table 1: User reactions from Otvet@Mail.ru (267 questions)

Model	*Likes*	*Amount of best answers*	*Users below*
BM25	148	8	15.47%
QTR	142	16	14.91%
word2vec	147	14	15.82%
LETOR	156	12	16.93%
Oracle	197	50	24.67%

Table 2: Lab evaluation results (50 questions)

Model	*top-1*	*DCG@3*
BM25	0.76	1.48
QTR	0.85	1.58
word2vec	0.77	1.62
LETOR	0.74	1.41
Oracle	1.63	2.95

REFERENCES

[1] Jean Carletta. 1996. Assessing agreement on classification tasks: the kappa statistic. *Computational linguistics* 22, 2 (1996), 249–254.

[2] Karen Spärck Jones, Steve Walker, and Stephen E. Robertson. 2000. A probabilistic model of information retrieval: development and comparative experiments. *Information Processing & Management* 36, 6 (2000), 779–840.

[3] Andrey Kutuzov and Elizaveta Kuzmenko. 2017. WebVectors: a Toolkit for Building Web Interfaces for Vector Semantic Models. (2017).

[4] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013). http://arxiv.org/abs/1301.3781

[5] Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. 2011. Learning to rank answers to non-factoid questions from web collections. *Computational Linguistics* 37, 2 (2011), 351–383.

[6] Diyi Yang, Alon Lavie, Chris Dyer, and Eduard Hovy. 2015. Humor Recognition and Humor Anchor Extraction. In *Proc. of EMNLP*. 2367–2376.