Avoiding Echo-Responses in a Retrieval-Based Conversation System

Denis Fedorenko(✉), Nikita Smetanin, and Artem Rodichev

Replika.ai @ Luka, Inc., Moscow, Russia
{denis,nikita,artem}@replika.ai

Abstract. Retrieval-based conversation systems generally tend to highly rank responses that are semantically similar or even identical to the given conversation context. While the system's goal is to find the most appropriate response, rather than the most semantically similar one, this tendency results in low-quality responses. We refer to this challenge as the echoing problem. To mitigate this problem, we utilize a hard negative mining approach at the training stage. The evaluation shows that the resulting model reduces echoing and achieves better results in terms of Average Precision and Recall@N metrics, compared to the models trained without the proposed approach.

Keywords: Dialog modeling · Response selection · Lexical repetition
Hard negative mining · End-to-end learning

1 Introduction

The task of a retrieval-based conversation system is to select the most appropriate response from a set of responses given the input context of a conversation. The context is typically an utterance or a sequence of utterances produced by a human or by the system itself. Most of the state-of-the-art approaches to retrieval-based conversation systems are based on deep neural networks (NNs) [14,16]. Under these approaches, the typical response selection pipeline consists of the following steps [2]:

1. Encode the given context and pre-defined response candidates into numeric vectors, or thought vectors, using NNs;
2. Compute the value of a matching function (matching score) for each pair consisting of a context vector and each response candidate;
3. Select the response candidate with the highest matching score.

During step 1, in order to obtain thought vectors that fairly represent semantics of input contexts and responses, the conversation model is preliminarily trained to return high matching scores for true context-response pairs and low for false ones.

D. Ustalov et al. (Eds.): AINL 2018, CCIS 930, pp. 91–97, 2018.
https://doi.org/10.1007/978-3-030-01204-5_9

The challenge we faced while building the above pipeline was that the resulting model often returned high matching scores for semantically similar contexts and responses. Consequently, the model frequently repeated or rephrased input contexts instead of giving quality responses.

Consider the following conversations:

A. Context: “What is the purpose of living?”
Response: “What is the purpose of existence?”

B. Context: “What is the purpose of living?”
Response: “It’s a very philosophical question.”

The effect of rephrasing, or echoing, in conversation A in contrast to the appropriate response in conversation B can be explained by the above pipeline. It is a result of the fact that contexts and responses often contain the same concepts [4,13], hence during training on conversational datasets the NNs simply end up trying to fit the semantics of the input. The similar effect, named “lexical repetition”, was also observed in [9].

In this paper, we suggest a simple and natural solution to the echoing problem for end-to-end retrieval-based conversation systems. Our solution is based on a widely used hard negative mining approach [10], which forces the conversation model to produce low matching scores for similar contexts and responses.

The paper is organized as follows. First, we describe the hard negative mining method and how we utilize it to overcome the echoing problem. Then, we introduce the evaluation metrics, our results and benchmarks for the echoing problem. We also provide the evaluation dataset used in the experiments for further research.

2 Hard Negative Mining

Let $D = \{(c_i, r_i)\}$, $i \in \{1..N\}$ be a dataset of conversational context-response pairs, where c_i, r_i – i-th context and response, respectively.

Our goal is to build a conversation model $M : (\mathit{context}, \mathit{response}) \rightarrow \mathbb{R}$ that satisfies the following condition:

$$M(c_i, r_i) > M(c_i, r_j) \tag{1}$$

$\forall i, j \neq i$ and r_j is not an appropriate response for c_i. In other words, the resulting model should return a higher matching score for appropriate responses than for inappropriate ones.

To train this model, we also need false context-response pairs as negative examples in addition to the positive ones presented in D. Consider two approaches to obtain the negative pairs: random sampling and hard negative mining. Under the first approach, we randomly select r_j from D for each c_i. If D is large and diverse enough, then a randomly selected r_j is almost always inappropriate for a corresponding c_i.

In contrast to random sampling, hard negative mining imposes a special constraint on responses selected as negatives. Let M_0 be a conversation model trained on random pairs used as negative training examples. Then, we search for a new set of negative pairs (c_i, r_j), so that their matching score satisfies the following condition:

$$M_0(c_i, r_i) - M_0(c_i, r_j) \leq m \tag{2}$$

where m is a margin (hyperparameter) between the scores of positive and negative pairs [3]. The new set of pairs is used to train the next model M_1, which, in turn, used to search for negative pairs to train M_2, and so on [1].

The intuitive idea behind hard negative mining is to select only negatives that have relatively high matching scores, and thus can be interpreted as errors of the conversation model. As a result, the model converges faster compared to random sampling [10].

Following this intuition, we can solve the echoing problem by considering contexts as possible responses, therefore the pairs (c_i, c_i) can be selected as hard negatives. In the next section, we demonstrate that this approach can ultimately prevent the conversation model from assigning a high rank to responses that are similar to contexts.

3 Experiments

For our experiments, we implement a model similar to Basic QA-LSTM described in [12]. It has two bidirectional LSTMs of size 2048 (1024 units in each direction), with separate sets of weights that encode a context and a response independently. We use a max pooling operation to calculate final thought vectors of these LSTMs. We use a cosine similarity as the output matching function. We represent input words as embeddings of size 256, which are initialized by the pre-trained word2vec vectors [8] and are not updated further during the model training. Word sequences longer than 20 words are trimmed from the right, and the context encoder is fed with only one dialog step at a time.

3.1 Models

In order to study the impact of hard negative mining on the echoing problem, we train three models using the following strategies: random negative sampling (RN), hard negative mining based on responses only (HN_r), and hard negative mining based on both responses and contexts (HN_{r+c}). We also consider the following baseline approach (BL): we use RN model to rank responses in the testing stage and then just filter out responses equal to the given context.

3.2 Datasets

We train the models on 79M of tweet-reply pairs from a Twitter data archive[1].

We perform an evaluation based on our own dataset[2]. This dataset consists of 759 context-response pairs from human text conversations, where context and

[1] https://archive.org/details/twitterstream.

[2] https://github.com/lukalabs/replika-research/tree/master/context-free-dataset.

Table 1. Evaluation dataset sample (see Sect. 3.2)

Context	Response
What happened to your car?	I got a dent in the parking lot
The beatles are the best	They are the best musical group ever
I'm joining the army	You're kidding. You might get killed

response both consist of a single sentence (see Table 1). We split the dataset into validation and test subsets consisting of 250 and 509 pairs, respectively. We use this dataset because it is clear, diverse and covers multiple topics of real-life conversations. Also we find it suitable for validating the echoing problem, as well as for estimating the overall model quality.

3.3 Training

The models are trained with the Adam optimizer [5] with the size of mini-batches set to 512. Intermediate models that show the highest values of the Average Precision metric on the validation set (see Sect. 3.4) are selected as the resulting models.

We use a triplet loss [3] as an objective function:

$$max(0, m - M(c_i, r_i) + M(c_i, r_j)) \tag{3}$$

where the margin m is set to 0.05. For each positive pair (c_i, r_i), a negative (c_i, r_j) is only selected within the current mini-batch using an intermediate model M trained by the moment of this batch. We only select the hard negative r_j with the highest matching score $M(c_i, r_j)$ satisfying the following condition:

$$0 \leq M(c_i, r_i) - M(c_i, r_j) \leq m \tag{4}$$

The constraint $0 \leq M(c_i, r_i) - M(c_i, r_j)$ is used to filter out the "hardest" negatives, which in practice affect convergence and lead to bad local optima [10].

We noticed that while training the HN_{r+c} model, the fraction of (c_i, c_i) negative pairs constitute up to 50% of the mini-batch.

3.4 Evaluation Methodology and Metrics

For each $context_i$ from the evaluation set, we compute matching scores for all available pairs $(context_i, answer)$, where $answer$ comes not only from the responses, but also from the all available contexts. To evaluate these results, we sort the answers by the matching score in descending order and compute the following metrics: Average Precision [7], Recall@2, Recall@5, and Recall@10 [6]. The last three metrics are indicator functions that return 1, if the ground-truth response occurs in the top 2, 5 and 10 candidates, respectively. We also introduce the context echoing metrics:

Table 2. Evaluation results based on the context-response test set (See footnote 2).

	RN	BL	HN_r	HN_{r+c}
Average Precision	0.12	0.16	0.13	**0.17**
Recall@2	0.18	0.26	0.23	**0.29**
Recall@5	0.36	0.37	0.4	**0.43**
Recall@10	0.45	0.48	**0.54**	0.53
$rank_{context}$	0.9	-	0.49	**19.43**
$diff_{top}$	0.008	-	0.01	**0.07**
$diff_{response}$	-0.15	-	-0.25	$\mathbf{-0.09}$

- $rank_{context}$ – position (starting from zero) of the input context in the sorted results. The greater the rank, the less the model tends to return the input context among the top results
- $diff_{top}$ – difference between the top result score and the input context score. The greater the difference, the less the model tends to return relatively high scores for the context
- $diff_{response}$ – difference between the ground-truth response score and the input context score. The greater the difference, the less the model tends to return similar scores for the ground-truth response and for the context

For each metric, we compute the overall quality as an average across all test contexts. Note that for BL model we don't present context echoing metrics, since echo-responses are filtered out from the results in this approach.

3.5 Results

The results of the evaluation based on the test set are presented in Table 2. As we can see, the proposed HN_{r+c} model achieves the highest values in almost all metrics compared to other approaches. According to $rank_{context}$, it turns out that this model does not tend to highly rank input contexts and have them in the top response candidates. Still, according to the $diff_{response}$ metric, the average score of a ground-truth response is lower than the score of a context, which means that the context can be ranked higher than the ground-truth response.

We also studied the model's output. Examples of top-ranked responses for different contexts are presented in Table 3. As we can see, oftentimes the RN and HN_r models select identical or very similar responses, while the proposed HN_{r+c} model selects appropriate responses that are not necessarily semantically similar to the context. Based on this observation, we suggest that the proposed model filters out not only exact copies of the context, but also candidates with similar semantics.

Table 3. Top 3 responses for a few input contexts sorted by matching score.

RN	HN_r	HN_{r+c}
Input: What is the purpose of dying?		
1. What is the purpose of dying?	1. What is the purpose of dying?	1. To have a life
2. The victim hit his head on the concrete steps and died	2. What is the purpose of living?	2. When you die and go to heaven, they will offer you beer or cigarettes
3. To have a life	3. What is the purpose of existence?	3. It is to find the answer to the question of life
Input: What are your strengths?		
1. What are your strengths?	1. What are your strengths?	1. Lust, greed, and corruption
2. Lust, greed, and corruption	2. What are your three weaknesses?	2. I'm a robot. a machine. 100% ai. no humans involved
3. A star	3. What do you think about creativity?	3. Dunno. i mean, i'm a robot, right? robots don't have a gender usually
Input: I can't wait until i graduate		
1. I can't wait until i graduate	1. I can't wait until i graduate	1. What college do you go to?
2. What college do you go to?	2. What college do you go to?	2. School is hard this year
3. School is hard this year	3. How many jobs have you had since leaving university?	3. What subjects are you taking?

4 Related Work

In the previous works on dialog systems there was not enough attention paid to the echoing problem. The possible reason for this are "soft" evaluation conditions: test samples are constructed from a relatively small number of negative responses [3,6,14] which usually do not "echo" the test context. In [9] the "lexical repetition" is regularized by utilizing a word overlap feature during training a SMT-based dialog system. In [11,13,14] the echoing is avoided by considering only responses the dataset's contexts of which have high TF-IDF similarity with the given context. However, the latter approach is not applicable if only a set of responses is available for ranking during the testing stage, which can be the case for some domains and applications [15].

5 Conclusion

In this study, we applied a hard negative mining approach to train a retrieval-based conversation system to find a solution to the echoing problem, that is, to reduce inappropriate responses that are identical or too similar to the input context. In addition to responses, we consider contexts themselves as possible hard negative candidates. The evaluation shows that the resulting model avoids echoing the input context, tends to select candidates that are more appropriate as responses and achieves better results in terms of Average Precision and Recall@N metrics compared to the models trained without the proposed approach.

References

1. Canévet, O., Fleuret, F.: Efficient sample mining for object detection. In: Proceedings of the 6th Asian Conference on Machine Learning (ACML), No. EPFL-CONF-203847 (2014)
2. Chen, H., Liu, X., Yin, D., Tang, J.: A survey on dialogue systems: recent advances and new frontiers. SIGKDD Explor. Newsl. **19**(2), 25–35 (2017). https://doi.org/10.1145/3166054.3166058
3. Feng, M., Xiang, B., Glass, M.R., Wang, L., Zhou, B.: Applying deep learning to answer selection: a study and an open task. In: 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), pp. 813–820. IEEE (2015)
4. Jurafsky, D., Martin, J.: Dialog systems and chatbots. In: Speech and Language Processing, vol. 3 (2017). https://web.stanford.edu/~jurafsky/slp3/29.pdf
5. Kingma, D., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
6. Lowe, R., Pow, N., Serban, I., Pineau, J.: The ubuntu dialogue corpus: a large dataset for research in unstructured multi-turn dialogue systems. CoRR abs/1506.08909 (2015). http://arxiv.org/abs/1506.08909
7. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, New York (2008)
8. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR abs/1301.3781 (2013). http://arxiv.org/abs/1301.3781
9. Ritter, A., Cherry, C., Dolan, W.B.: Data-driven response generation in social media. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 583–593. Association for Computational Linguistics (2011)
10. Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: a unified embedding for face recognition and clustering. CoRR abs/1503.03832 (2015). http://arxiv.org/abs/1503.03832
11. Serban, I.V., et al.: A deep reinforcement learning chatbot. arXiv preprint arXiv:1709.02349 (2017)
12. Tan, M., Xiang, B., Zhou, B.: LSTM-based deep learning models for non-factoid answer selection. CoRR abs/1511.04108 (2015). http://arxiv.org/abs/1511.04108
13. Wang, H., Lu, Z., Li, H., Chen, E.: A dataset for research on short-text conversations. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 935–945 (2013)
14. Wu, Y., Wu, W., Zhou, M., Li, Z.: Sequential match network: a new architecture for multi-turn response selection in retrieval-based chatbots. CoRR abs/1612.01627 (2016). http://arxiv.org/abs/1612.01627
15. Yan, Z., et al.: DocChat: an information retrieval approach for chatbot engines using unstructured documents. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), vol. 1, pp. 516–525 (2016)
16. Zhou, X., et al.: Multi-view response selection for human-computer conversation. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 372–381 (2016)