Increasing Diversity with Deep Reinforcement Learning for Chatbots

Cristian Pavel
University Politehnica of Bucharest
313 Splaiul Independentei,
Bucharest, Romania
cristian.pavel@stud.acs.upb.ro

Ştefania Budulan
University Politehnica of Bucharest
313 Splaiul Independentei,
Bucharest, Romania
stefania.budulan@cs.pub.ro

Traian Rebedea
University Politehnica of Bucharest
313 Splaiul Independentei,
Bucharest, Romania
traian.rebedea@cs.pub.ro

DOI: 10.37789/rochi.2020.1.1.19

ABSTRACT

Dialogue generation for open-domain conversations is a difficult and open problem that, so far, has not been able to approach human-level performance. Recently, a popular solution is to apply a sequence-to-sequence architecture, similar to the machine translation problem. These models try to map the input - given as the previous utterances, to the output - the next utterance. Unfortunately, they usually tend to repeat sentences, often preferring dull responses, that end the conversation abruptly. Therefore, Reinforcement Learning techniques have been combined with the standard sequence-to-sequence models in order to avoid their shortcomings. Our model applies a Policy Gradient method that maximizes the expected reward of generating the next utterance given a history of previous utterances. The results show an improvement in diversity up to 0.16 - almost 10x higher than the model without RL, while keeping the responses relevant to the input message.

Author Keywords

Dialogue generation; Reinforcement learning;
Conversational agents; Sequence-to-sequence model.

INTRODUCTION

A simplistic approach in dialogue generation for conversational agents uses a supervised learning method that tries to generate the next turn of a conversation, given a subset of the previous turns. This approach fosters some downsides. Firstly, it uses a Cross-Entropy Loss, which suffers from exposure bias [3] and has no quantifiable relation with the traits that healthy dialogues should have (e.g., informativeness, engagement, or diversity). Reinforcement Learning (RL) manages to overcome these problems by using rewards, aiming to guide the model towards an action space that is consistent with generating a human-like dialogue.

The main focus of this work is the problem of low diversity, representing the tendency of a model trained with Cross-Entropy Loss to output generic responses such as '*I don't know*' or '*I have no idea*' [12, 19]. This issue appears mainly from the sparsity of the data and the high number of inputs that a generic response can match [18].

Our proposed solution resides on the work of Li et al. [13]. We use three models, one (the Reinforcement Learning model) leveraging the other two (sequence-to-sequence) to compute rewards and update its actions. The standard sequence-to-sequence architecture [4, 23] stands as a building block for all the models. The REINFORCE [26] algorithm is used in the training iterations of the final network, which is initialized from a pre-trained chatbot in a supervised manner.

In the context of Human-Computer Interaction, chatbots can pave the way to artificial general intelligence. A survey of the vast number of conversational agents developed in recent years has been performed by Grudin and Jacques [10]. As the survey authors emphasized, constructing an open-domain chatbot is an arduous task and the present work is another proof of that statement. In addition to this, Allen et al. [] analyze the possibility of using a practical dialogue between the user and the system as the main mechanism connecting the two and hypothesize that such a user interface can replace the current popular Graphical User Interfaces (GUI) in the future. Moreover, Følstad and Brandtzaeg [] discuss in more detail the implications, challenges and opportunities that emerge when transitioning towards natural language interfaces and chatbots, a next step which is predicted by multiple tech companies, according to the authors.

RELATED WORK

Neural response generation has been extensively studied in the last few years, starting from the novel idea of Ritter et al. [17] who propose the application of Statistical Machine Translation (SMT) techniques to the problem at hand.

Following this approach, due to the success of the sequence-to-sequence (seq2seq) architecture [4, 23] for the Neural Machine Translation problem, this method has been rapidly transferred to the dialogue generation task [17, 25]. These simple networks manage to respond coherently and even preserve some context, without any prior knowledge or pre-engineered rules. These results motivate our choice of the seq2seq architecture with a Recurrent Neural Network (RNN) based encoder and decoder. In a similar direction, an end-to-end approach is tackled by Sordoni et al. [22]. They propose three different methods to incorporate the context of the conversation into the generation procedure and they decide to use their model as an extra feature to the SMT systems reaching an improvement over the considered baselines. We also experiment with their intuition of concatenating the context to the current message and then pass the result to the encoder.

In overcoming the lack of diversity of these models [12, 19], there have been proposed several solutions. A variational auto-encoder can be used to add additional variance into the model [20]. Similarly, promising results have been obtained by using a Generative Adversarial Network (GAN) usually combined with Reinforcement Learning (RL) to backpropagate the error from the discriminator to the generator [14]. Moreover, Li et al. [13] utilize Reinforcement Learning with heuristic rewards that try to capture relevant attributes of a dialogue and increase considerably the diversity of the baseline. The model in our paper also makes use of these rewards and their architecture stands as a starting point for our implementation.

The Transformer architecture [24] has shown great potential in Natural Language Processing (NLP), especially with the emergence of BERT [8] and the possibility to fine-tune this architecture depending on the task at hand. Recently, this robust model has also been applied to the dialogue generation task. One example of this is the Meena chatbot [1] that outperforms previous well-known chatbots such as Cleverbot [5] or Xiaoice [27]. The authors also propose a new human evaluation metric, Sensibleness and Specificity Average (SSA), that incorporates both sensibility and specificity and show that this metric is correlated with perplexity. For the current experiments, we do not utilize a Transformer network, but future work can aim to improve our results by incorporating a Transformer-based seq2seq model.

DATASET

The dataset we used for training is called Cornell Movie Dialogs [7] and it contains metadata-rich exchanges extracted from various movies. There are 221,282 sentence-response pairs accompanied by information about the speakers involved and the movies in which each exchange takes place. Figure 1 shows the distribution of the utterances' lengths in the dataset. The choice of this dataset is motivated by its relatively small size compared with other datasets (e.g., OpenSubtitles [15]) while also being easy to parse and use. It also contains less noise and thus a model can be trained without needing too many input-output pairs. Training on such a small dataset we do not set about or expect to construct a state of the art final model. The goal remains to tackle the diversity problem, while being able to respond to simple input messages. We chose to eliminate the input-output pairs in which either the message or the response had more than 10 tokens. After this elimination we remain with approximately 28% of the data.

Because the concerning issue is related to generic utterances, Table 1 shows the most frequent sentences in responses from the training data, multiplied by the number of different bigrams in the input messages and scaled by the total number of bigrams in the vocabulary. This is done to differentiate frequent input-output pairs from generic responses that fit multiple different inputs. These responses will later be used in the calculation of the rewards.

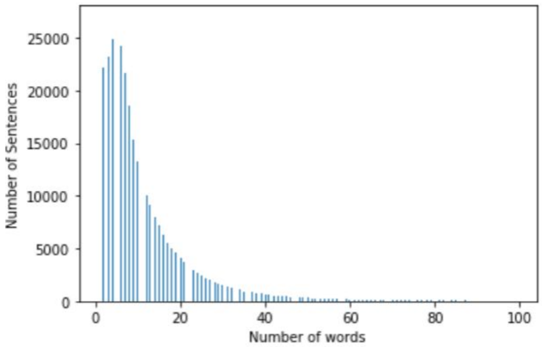


Figure 1. Histogram showing the distribution of the lengths of the utterances in the dataset

Table 1. Frequent sentences in responses from the training data

Response	Scaled Frequency (x 10^{-10})
I don't know.	1.32
Yeah.	1.28
Well.	1.17
I.	1.12
No.	0.76
Yes.	0.64
What?	0.61
Okay.	0.52

MODELS

The model proposed draws inspiration from Li et al. [13]. Three main components play a role in the final dialogue generation network: a forward network **F**, a backward network **B**, and a final network **R** trained using Reinforcement Learning. The next sections will analyze each one of them individually.

Forward Network

This encoder-decoder network functions as a map between a message paired with a dialogue context and a response, similar to Sordoni et al. [22]. Therefore, our encoder receives as input the concatenation of the context and the input message.

Regarding the details of the implementation, GRU cells [4] are used, due to their relative simplicity compared to the more complicated LSTM cell [11]. A bidirectional RNN is used for the encoder as it has been shown that it is successful in similar tasks [2].

Backward Network

This network receives an utterance as input, and it has to predict the previous sentence that would have occurred in a natural-sounding dialogue. The same implementation details as for the Forward network are used. In both cases the cost function is the Cross-Entropy Loss.

Reinforcement Learning Network

The final network is trained using Policy Gradient optimization techniques. The policy is parameterized using a seq2seq model and its weights are initialized from the weights of the F network. Using examples from the dataset, training is achieved by performing a Monte-Carlo roll-out of one or multiple transitions according to the decoder policy. The REINFORCE algorithm is implemented together with a baseline value to reduce the variance that occurs while training. T, considering the dull utterances from Table 1 when calculating the reward score referred as *Ease of Answering* by the authors.

Rewards

The heuristically determined rewards used are the ones proposed by Li et al. [13]. The first reward (r_1) aims to drift the model away from generating responses that may lead the conversation towards dull sequences. In the formula below, S is a hardcoded list of generic responses, dependent on the dataset, that contains the most frequent target answers (e.g., *'I don't know'*), a is the response generated by the network, $N_S = |S|$, and N_s is the number of tokens in s.

$$r_1 = -\frac{1}{N_S} \sum_{s \in S} \frac{1}{N_s} log(p_F(s|a))$$

The conditional probability p_F is calculated using the pre-trained sequence-to-sequence network F. In our case, the hardcoded list of frequent responses is, also, presented in Table 1.

The second reward (r_2) penalizes the agent if it generates similar responses in consecutives turns. Thus, considering the dialogue A, B, C, we transform each of the sequences A and C into fixed vector representations, h_A and, respectively, h_C, through an encoder layer and then compute the logarithm of the cosine similarity of the two embeddings. We also add a threshold $e > 0$ to deal with the fact that the logarithm is defined only for positive values. In our experiments the value is set to $e = 10^{-10}$.

$$r_2 = -log(max(\frac{h_A \cdot h_C}{\|h_A\| \cdot \|h_B\|}, e))$$

The third reward (r_3) keeps the agent from diverging and generating unintelligible sequences by rewarding semantic coherent responses. Here we also consider the dialogue sequence A, B, C, with N_B the number of tokens in the sequence B, and N_C the number of tokens in C.

$$r_3 = \frac{1}{N_C} log(p_F(C|B,A)) + \frac{1}{N_B} log(p_B(B|C))$$

Given these three equations, the final reward, at the end of a transition, can be computed by:

$$R = \lambda_1 \cdot r_1 + \lambda_2 \cdot r_2 + \lambda_3 \cdot r_3 ,$$

where the coefficients suggested by the authors are $\lambda_1 = 0.25$, $\lambda_2 = 0.25$ and $\lambda_3 = 0.5$. We have experimented with different values for these coefficients as shown in the next section.

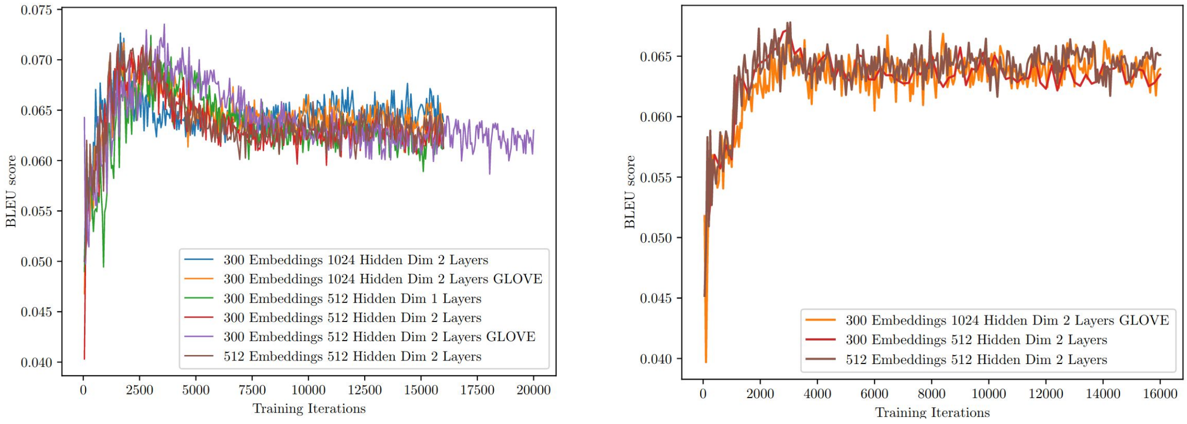
EXPERIMENTS AND RESULTS

One of the first decisions that we made in our experiments was to eliminate the context. Initially, training using one utterance as the context, we observed the inability of the model to generalize and generate relevant and coherent messages for a conversation that spanned multiple turns.

An example of this behaviour is shown in Table 2. This is caused by the small size of the training dataset and, therefore, the model's unpredictability when it receives unseen pairs of contexts and messages.

To evaluate our models we employ a diversity metric [12], representing the number of unique unigrams and bigrams generated normalized by the total number of tokens generated, to objectively measure the diversity of the responses. A higher diversity metric correlates to more diverse responses. Also, we make use of the Bilingual Evaluation Understudy (BLEU) [16] score to choose between different hyperparameters when training the Forward and the Backward networks. BLEU score is not a perfect measure, but it has a correlation with human judgment as shown by Galley et al. [9].

In Figure 2, the BLEU score is plotted for the validation dataset throughout the training steps.



a) Forward network **b) Backward network**

Figure 2. BLEU score while training for validation sets. An overfitting pattern can be observed

Both models appear to overfit, this phenomenon being more visible in the case of the Forward network. Inspecting the test data manually, we conclude that, if we allow the model to further learn, the responses tend to be more diverse and accurate, although some mismatches emerge. Some relevant examples are depicted in Table 3. The diversity metric [12] stands as another proof of this occurrence, showing an increase from 0.006 to 0.017 when unigrams are considered, and from 0.02 to 0.13, for bigrams. The first metrics are calculated before the overfitting occurs, while the second is computed after the process. We observe that diversity is no longer a problem after this pattern occurs, so the final network will only be used on the models whose training was stopped before the validation scores would have decreased. A similar observation is made by Csaky [6], who uses a Transformer model on the same dataset and also notes this behaviour.

Table 2. Example of the model degenerating. Once it outputs UNK, it stops generating meaningful responses

User: Hello.
Bot: Hello.
User: How are you?.
Bot: Fine.
User: What's your name?
Bot: UNK.
User: How are you?.
Bot: I don't know I don't know.

We continue to train the model that suffers considerably from the diversity issue, the one before overfitting occurs on the BLEU score (0.071 for the Forward network and 0.069 for the Backward network), and apply Reinforcement Learning to improve its diversity.

Table 3. Comparison between the model before score decreasing and the one at convergence

Input	Before overfitting	At convergence
Hello.	Hi.	It's me.
How old are you?	I don't know.	Five.
How are you?	I don't know.	Head still secure to the neck.

Mixed Incremental Cross-Entropy Reinforce (MIXER) [17] is utilized first with just the third reward, as suggested by Li et al. [13]. The R network is initialized with the parameters of the F network and then, we train for T - Δ steps in the same supervised fashion as before, and for the remaining Δ steps we use the REINFORCE algorithm. We increase gradually the value of Δ until all the sequence is trained with Reinforcement Learning. The diversity increases considerably from 0.006 to 0.032 for unigrams and from 0.017 to 0.144 for bigrams.

Following this, we train the model by using all the three rewards for 5 transitions per episode, setting the discount factor to 0 because no clear distinction could be made between a final and a non-final state. Following multiple experiments, the coefficients used for the rewards are changed to 1.0 for the first reward, 5.0 to the second reward and 0.1 to the third reward, to eliminate the model's tendency to converge to a safe space where it generates a single generic response to all inputs. These values are found through an empirical search, by observing that the third reward highly influences the agent and forces it to output the same response for multiple turns, while the second reward scarcely has an impact on the final model. Moreover, we changed the way the similarities of two sentences are computed in the second reward to use the embeddings and not the encoder final hidden states as suggested by Li et al. [13]. This is due the model's inability to generalize well to unseen messages sampled in the Monte-Carlo generation process.

The final diversities scores are shown in Table 4, where the F model is the model before overfitting occurs, the F_O model is the one after overfitting, R_M is the model after MIXER and R is the final model. The main observation is that MIXER and overfitting lead to the greatest relative increase, but the final model manages to achieve the best diversity scores.

Table 4. Final diversity scores

	Unigram	Bigram
F	0.006	0.017
F_O	0.027	0.137
R_M	0.032	0.144
R	**0.033**	**0.16**

In Table 5 a comparison is shown between our model and the model implemented by Li et al. [13]. The responses for their model are taken explicitly from their paper. Both models offer diverse and relevant responses, but from these examples one can observe that their model is more interactive, as it asks more questions, due to its exposure to more data and epochs for learning.

Table 5. Final responses

Input	R	Chatbot [13]
How old are you?	Twenty eight.	I'm 16, why are you asking?
What's your full name?	Roy.	What's yours?
How much time do you have here?	Not enough. What do you want?	Ten seconds.

CONCLUSIONS

The experiments conducted in this paper have shown the ability of Reinforcement Learning to allow the model to deflect from the diversity issue. This approach is valid even for a smaller dataset with short sentences as the one used in our research.

The main observation is that the model trained in a supervised fashion, using the standard Cross Entropy Loss, suffers considerably from the diversity issue. This problem can be alleviated by allowing the model to overfit on the training dataset, but actually the best results appear after Reinforcement Learning is applied.

The limitation of the final chatbot comes from the fact that, being trained on a small dataset, the agent cannot perform a coherent and consistent conversation that spawns more than a few turns. That being said, the results achieved in this paper are significant with respect to future research in exploring other variants of rewards, datasets or architectures combined with Reinforcement Learning. This kind of empirical research is beneficial in understanding the capabilities of the neural networks employed for building deep learning chatbots.

REFERENCES

1. Adiwardana, D., Luong, M. T., So, D. R., Hall, J., Fiedel, N., Thoppilan, R., ... & Le, Q. V. (2020). Towards a human-like open-domain chatbot. arXiv preprint arXiv:2001.09977.

2. Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

3. Bengio, S., Vinyals, O., Jaitly, N., & Shazeer, N. (2015). Scheduled sampling for sequence prediction with recurrent neural networks. In Advances in Neural Information Processing Systems (pp. 1171-1179)

4. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

5. Cleverbot https://www.cleverbot.com/. Accessed May 20, 2020.

6. Csaky, R. (2019). Deep learning based chatbot models. arXiv preprint arXiv:1908.08835.

7. Danescu-Niculescu-Mizil, C., & Lee, L. (2011). Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. arXiv preprint arXiv:1106.3077.

8. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

9. Galley, M., Brockett, C., Sordoni, A., Ji, Y., Auli, M., Quirk, C., ... & Dolan, B. (2015). deltableu: A

discriminative metric for generation tasks with intrinsically diverse targets. arXiv preprint arXiv:1506.06863.

10. Grudin, J., & Jacques, R. (2019, May). Chatbots, humbots, and the quest for artificial general intelligence. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (pp. 1-11). DOI:https://doi.org/10.1145/3290605.3300439

11. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

12. Li, J., Galley, M., Brockett, C., Gao, J., & Dolan, B. (2015). A diversity-promoting objective function for neural conversation models. arXiv preprint arXiv:1510.03055.

13. Li, J., Monroe, W., Ritter, A., Galley, M., Gao, J., & Jurafsky, D. (2016). Deep reinforcement learning for dialogue generation. arXiv preprint arXiv:1606.01541.

14. Li, J., Monroe, W., Shi, T., Jean, S., Ritter, A., & Jurafsky, D. (2017). Adversarial learning for neural dialogue generation. arXiv preprint arXiv:1701.06547.

15. Lison, P., & Tiedemann, J. (2016). Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles.

16. Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002, July). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics* (pp. 311-318).

17. Ranzato, M. A., Chopra, S., Auli, M., & Zaremba, W. (2015). Sequence level training with recurrent neural networks. arXiv preprint arXiv:1511.06732.

18. Ritter, A., Cherry, C., & Dolan, W. B. (2011, July). Data-driven response generation in social media. Proc. Conference on Empirical Methods in Natural Language Processing 2011 (pp. 583-593).

19. Serban, I. V., Sordoni, A., Bengio, Y., Courville, A., & Pineau, J. (2016, March). Building end-to-end dialogue systems using generative hierarchical neural network models. In Thirtieth AAAI Conference on Artificial Intelligence.

20. Serban, I. V., Sordoni, A., Lowe, R., Charlin, L., Pineau, J., Courville, A., & Bengio, Y. (2017, February). A hierarchical latent variable encoder-decoder model for generating dialogues. In Thirty-First AAAI Conference on Artificial Intelligence.

21. Shang, L., Lu, Z., & Li, H. (2015). Neural responding machine for short-text conversation. arXiv preprint arXiv:1503.02364.

22. Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., ... & Dolan, B. (2015). A neural network approach to context-sensitive generation of conversational responses. arXiv preprint arXiv:1506.06714.

23. Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Advances in neural information processing systems (pp. 3104-3112).

24. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems(pp. 5998-6008).

25. Vinyals, O., & Le, Q. (2015). A neural conversational model. arXiv preprint arXiv:1506.05869.

26. Williams, R. J., & Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. Neural computation, 1(2), 270-280.

27. Zhou, L., Gao, J., Li, D., & Shum, H. Y. (2020). The design and implementation of xiaoice, an empathetic social chatbot. Computational Linguistics, 46(1), 53-93.