Reranking of responses using transfer learning for a retrieval-based chatbot

Ibrahim Taha Aksu, Nancy F. Chen, Luis Fernando D'Haro, Rafael Banchs

Abstract This paper presents how to improve retrieval-based open-domain dialogue systems by re-ranking retrieved responses. The paper uses a retrieval based open domain dialogue system implemented previously, namely Iris chatbot as a case study. We investigate two approaches to re-rank the retrieved responses. The first approach trains a re-ranker using machine generated responses that were annotated by human participants through WOCHAT (Workshops and Session Series on Chatbots and Conversational Agents)[1] and its shared-tasks [5], [6]. The second approach uses transfer learning by training the re-ranker on a large dataset from a different domain. We chose the Ubuntu dialogue dataset as the domain. The human evaluation test asked subjects to rank and review three different dialogue systems, the baseline Iris system, the Iris system enhanced with a re-ranker trained on WOCHAT data, and the Iris system enhanced with a re-ranker trained on the Ubuntu data. The Iris system enhanced with a re-ranker trained on WOCHAT data received the highest ratings from the human subjects.

Ibrahim Taha Aksu
Bilkent University, Ankara, Turkey , e-mail: taha.aksu@ug.bilkent.edu.tr

Nancy F. Chen
Institute For Infocomm Research, A*STAR,Singapore e-mail: nfychen@i2r.a-star.edu.sg

Luis Fernando D'Haro
Universidad Politécnica de Madrid, ETSI de Telecomunicación, Ciudad Universitaria, 28040 Madrid, Spain. ORCID: 0000-0002-3411-7384. e-mail: luisfernando.dharo@upm.es

Rafael Banchs
Nanyang Technological University, Singapore e-mail: rbanchs@ntu.edu.sg

[1] http://workshop.colips.org/wochat/

1 Introduction

Dialogue systems are often classified into two categories with respect to their objectives: Task oriented dialogue systems and open domain dialogue systems [1]. Task oriented dialogue systems are designed to handle specific scenarios such as flight booking or restaurant reservations, whereas open domain dialogue systems do not focus on specific tasks to reach a target, but mostly focus on the continuity of the dialogue. In general, two approaches are used to provide responses for dialogue systems: Retrieval based models [2] and generative models [3].The retrieval based model uses a heuristic to choose a response from a given dataset of predefined responses, whereas the generative model generates new responses. In this work, we focus on re-ranking responses for a retrieval based model.

The Iris chatbot system [4] has access to a dataset of dialogues extracted from movie scripts.At each turn it is given the user utterance, the previous utterance and the dialogue history so far. Using TF-IDF measure it finds the best matches from the dataset to the utterance and to the given dialogue history and retrieves a list of candidate answers from where the system can take one as following utterance in the dialogue.

The retrieved utterances are the best candidates to give as response but usually choosing the best one is not what a heuristic statistic can do. This is where the need for a re-ranker arises. The re-ranker is a network trained on a dataset that sorts the given candidate list with respect to their relevance to the given utterance and history, and chooses the best response to give. This paper uses two different re-rankers trained on two different datasets. One on Ubuntu dialogue corpus and the other is on a dataset that consists of annotated turns of the IRIS chatbot.

2 Related Work

Re-ranking has been commonly used in NLP problems such as parsing and translation [8], and many other studies also use it for response selection [9] [10].

Wang [11] trained two re-rankers using LSTMs. One of the re-rankers is called "strength-based re-ranker", which takes into account how often the answer to the question is encountered in related passages. The other re-ranker is called "coverage-based re-ranker", which ranks candidates higher when the union of all its contexts in different sentences could cover more aspects appearing in the question. The proposed re-rankers in this paper, different to our goal, are intended to find specific responses to a given open-domain question and therefore answers are unique. However, for a chitchat task, where there is no need to have a specific answer but to provide meaningful ones and to keep the conversation similar to what will happen when talking to a real person, the task of re-ranking answers could be more difficult as many candidates can be selected and the selection of one before other could be due to personal or subjective reasons.

Aktolga [9] introduced a two steps approach for ranking answers to a question. They first determine the type of the answer the question will be given and then only those candidates with the correct type are compared with the question in terms of their parse structures. This ensures that answers are not accidentally ranked highly if they contain some common sentences with the question. Similar to what we mentioned above, different to a QA system where correctly determining the type of a question is important (e.g. who, when, what, etc), for a conversational agent it could be possible, at some times, to provide a general answer or even to change the topic. For instance, it might be possible to reply to the user asking the chatbot for "Who is your favorite researcher in the area of NLP" with a "I do not know" or "this question gives me goosebumps".

Romeo [10] aims to rank the passages, retrieved as candidate answers to a question. The approach they use is to train an LSTM based network to rank similarities between the asked question and questions from the dataset. In order to do that they have two macro trees representing the original question and the candidate question which they merge to be syntactic trees of sentences composing both questions. They additionally link the trees by connecting the phrases whenever there is at least a lexical match. Our system in order to match similar questions uses word embedding which maps sequences of sentences to vectors in a an n-dimensional space. A possible future improvement to the system might be implementing the used LSTM approach in order to find similarities between questions.

All approaches mentioned above are applied to the problem of question answering (QA) systems, whereas in our work we apply response re-ranking to an open domain dialogue system. The problem we are working on has additional challenges. In QA answering systems the context is usually known which decreases the candidate answer space considerably a lot although for an open domain dialogue systems it may be not the case since the subject of the conversation can move in any direction.

3 Method

The aim of this paper is mainly to re-rank retrieved dialogues of an open domain dialogue system that is already implemented. The dialogue system has a very basic design where it has access to a dataset constituted by extracting dialogues from movie scripts. Given an utterance and the current history of the dialogue, the dialogue system finds the best match for both history and utterance and retrieves a list of responses that might be returned by using TF-IDF statistics. Two approaches are investigated.

The first approach is transfer learning, where the re-ranker is first trained on a larger out-of-domain dataset before it is applied to the target dataset. Ubuntu dialogue corpus, which is mostly on technical dialogue turns between users of the system, is used to train the re-ranker. The corpus has almost 1 million multi-turn dialogues with over 7 million utterances, it provides at each sample the history of the dialogue, the last utterance said by the user, the actual response given to that utter-

ance and 100 randomly chosen responses for each utterance[12]. Randomly chosen responses are provided in order to give user possibly a bad or not correct example along with the correct one for each utterance. The dataset was quite divergent from the purpose of our dialogue system which is just daily talk and chit-chat. The re-ranker worked with 0.7 accuracy on the test data.

The second approach is to use a dataset with the same context of daily talk in order to train an automatic annotator that will order the list of retrieved responses. The dataset consists of annotated responses from user dialogues of the Iris chatbot[13].Each data point has a user utterance and a chatbot response which are evaluated by annotators as valid, acceptable or invalid. The challenge in using this dataset is the limited labeled data it has. There are only 1200 turns of annotated data that can be used to train a re-ranker.

3.1 Word Embeddings

The word embeddings for both approaches were unsupervisely trained using the FastText library[2] on the Ubuntu Dialogue Corpus since it includes near 100 million words. The sentence embeddings were created by averaging embeddings of the words in the sentence.

3.2 Ubuntu Re-ranker

The Ubuntu dataset containing almost 1 million multi-turn dialogues, with a total of over 7 million utterances and 100 million words[14], mostly consists of technical questions and replies. Since we have an open domain dialogue system in hand, using this dataset to re-rank the retrieved responses is not expected to do as good as a dataset, whose context is similar to ours. The advantage in using the specified dataset comes from its size though. This paper uses this dataset in the context of transfer learning where the idea is to store a knowledge in solving one problem and use it to solve another but related problem[15]. Data forums like Ubuntu are more widely used on Q/A systems and this is what makes our study more challenging as we try to apply it to an open domain dialogue system.

Our training data is prepared in a way to include the utterance, the actual response and one of the wrong responses at each sample. Figure 1 shows what the re-ranker network is fed and what it outputs in return. The column of random and correct response in the data is chosen randomly with a label indicating which column belongs to the correct answer. This data is fed to the network hoping that given an utterance, a random answer and a correct answer it would learn to classify which answer is the one that is closer to an actual response. The network used is a multilayer perceptron

[2] https://fasttext.cc/

with two hidden layers with 500 neurons. The learning rate is 0.001, batch size is 64. We used Adam optimizer, and had a dropout rate of 0.5. The network trained has 72% accuracy on the test data.

The next step was to write an algorithm that chooses the best response out of the candidate list given a re-ranker that provides an accuracy of 72%. The re-ranker is deterministic, so it gives the same answer every time it is fed with the same input, but the probability that the answer is correct is only 72%. The algorithm implemented is shown in figure 2 and does the following: First it shuffles the candidate list randomly, then it sorts it using a standard sorting function, having the re-ranker network as the binary comparator and then assigns scores to candidates according to their places in the list. This procedure is repeated several times and scores are added to candidates. At the end, the candidate with the highest score is chosen as the response. This approach works as the following: due to initial random shuffling, each call to the sorting algorithm generates unique sorting network and since 72% of the comparisons are right then as the number of turns of shuffle-order increase the sorting will be satisfactory. Since the binary comparator is not perfect, it could have some inconsistencies in the results it gives. For example lets say we have an ut-

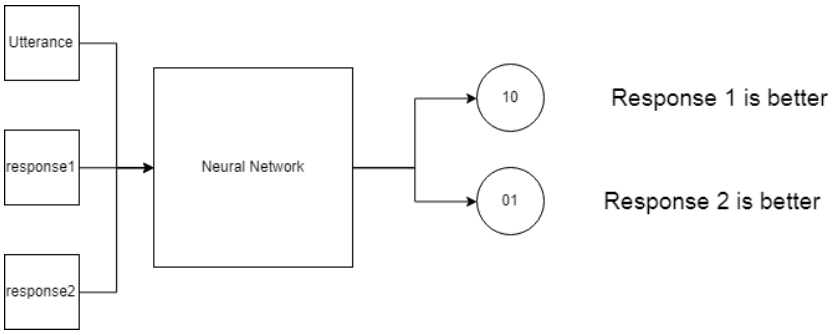

Fig. 1 Ubuntu Network

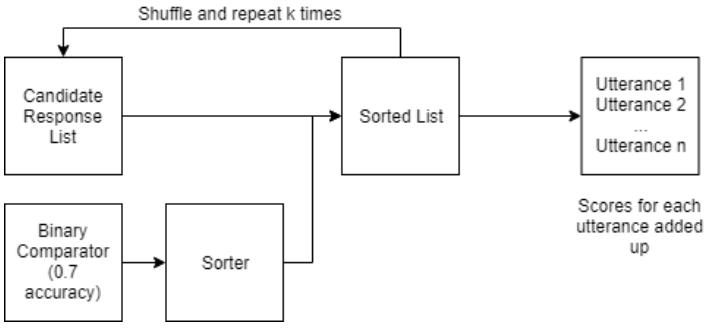

Fig. 2 Ubuntu Re-ranking algorithm

terance and 3 different responses A, B and C. We feed response pairs (A,B), (A,C), and (B,C) and the results the comparator gives in terms of validity is as follows respectively: A $>$ B, C $>$ A, B $>$ C that cannot be correct; however, our algorithm shuffles the response pairs multiple times and changes the set of binary comparisons at each turn. Thus, at the end, it is expected that it will choose the best or one of the top responses out of the n-best list.

3.3 Wochat Re-ranker

The Wochat re-ranker is trained on a dataset that consists of human annotations done on turns of IRIS chatbot's dialogues. As this dataset consists of turns created by the baseline dialogue system itself, it is a more convenient source for the training; unfortunately, its main drawback is that it has only 1500 annotated turns.

The network in this re-ranker is a multilayer perceptron with one hidden layer with 128 neurons.The learning rate is 0.008. We used Adam optimizer, and had a dropout rate of 0.8. The network, as shown in figure 3, is fed with an utterance and a response and classified as one of the 3 classes: Valid, acceptable or invalid. 1400 of the sample data used as training data while 100 for test and the results showed an accuracy of 75%; however, it would have been better to use k-fold cross validation to train and test the network which we did not to use during the implementation phase of the project.

The Wochat re-ranking algorithm is depicted in Figure 4 and works as follows: the network scores each utterance according to each classification type (valid, invalid and acceptable).Using these scores a total score for each response is found taking valid and acceptable scores as positive effects and invalid score as a negative effect on the final score. At the end, the response with the maximum score is returned. For example, an utterance might have a valid score of 0.6 an invalid score of 0.2 and an acceptable score of 0.8 while another one has 0.4, 0.7 and 0.5 respectively. In such a case, the score for the first one will be greater than the second since

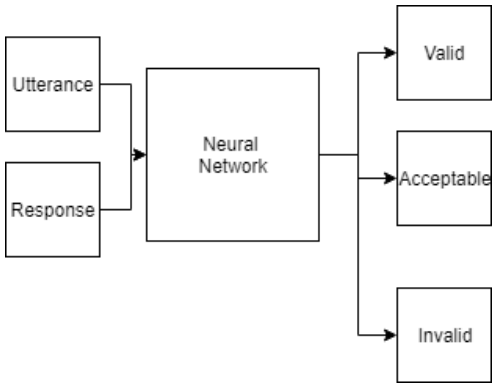

Fig. 3 Wochat Network

valid and acceptable scores are positive effects while the invalid score is a negative effect.

4 Evaluation

The evaluation of the paper was done in two ways. First by getting into similar conversations with the bots and checking their responses. In many cases, the improvements to the responses for bots combined with re-ranker is considerable.

Table 1 Wochat Bot Dialogue

Speaker	Utterance
User	Hi
Iris	Hey
User	How are you doing man
Iris	Okay..
User	What fruit do you like the most?
Iris	It's a vegetable.
User	Okay what is it?
Iris	Sir?
User	What is your favorite vegetable
Iris	Agua.

The second evaluation metric is the actual users: An interface through a website, similar to [7], was implemented to get users interact with the chatbots. Since the main concern was to see the improvement of the responses of the baseline chatbot, all three bots are put anonymously to the website. Users interact with each of the

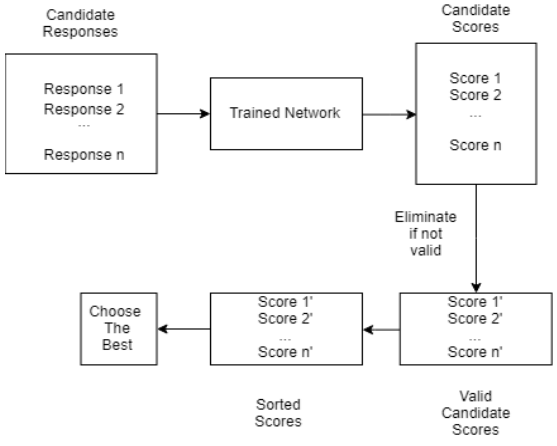

Fig. 4 Wochat Re-ranking algorithm

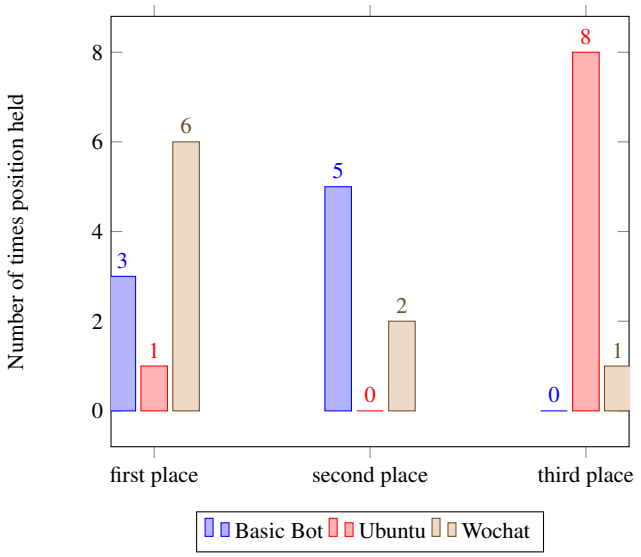

Fig. 5 Results of user evaluations,each bar refers to the number of times a bot is ranked in first,second or third place.

Table 2 Dialogue with basic chatbot

Speaker	Utterance
User	Hello, how are you?
Iris	Were you in an accident? Were you attacked? It's okay, I'm a doctor. my name is Miranda Grey...
User	I'm fine. So, you are a doctor, that's great
Iris	Is that it?
User	Yeah. you save lives
Iris	So why would you hesitate to lie to save lives?
User	I'm not talking about lives now. I'm surprised to know you are a doctor
Iris	What do you mean... what am i talking about? do you realize that you say that to me everyday of your life? i am dying. do you know... do you realize the death you are spreading around to the people who are around you

bots separately without knowing their identities. In the end they report their comments on each and order them from best to worst in terms of continuity and overall quality of the dialogue. There were 8 people taking the experiment and the results of the evaluations can be found in figure 5.

The comments are discussed through some example snippets of the chats that users had with the bots.

Table 3 shows an example snippet from a dialogue between the user and the baseline chatbot. The bot gives an overly dramatic response to the greeting coming from the user which is also irrelevant of the context.

Table 3 Dialogue with the original chatbot

Speaker	Utterance
User	Hello
Iris	Did you find it okay?
User	What do you mean?
Iris	I mean you've done a great thing. you should be rewarded monetarily
User	What did i do?
Iris	You threatened to drive her downtown. She has agoraphobia.
User	Who is she?
Iris	She's your wife!
User	But i am a woman and not gay what do you think about i?
Iris	I mean, hello!

Table 4 Example of dialogue using a re-ranker trained with Ubuntu corpus

Speaker	Utterance
User	I will need help with some school assignments
Iris	Yeah
User	It is about maths actually
Iris	weapons down
User	I got no weapons, only an old hp calculator :)
Iris	Who is it?
User	It is me Rafael
Iris	Who ?
User	Your friend from chatbot high
Iris	You hear me now?
User	Yes, let me know if you are helping with the maths assignment or not
Iris	Against
User	Ok, bye bye
Iris	Sorry about that.

The comments on Ubuntu chatbot are not promising as people rarely find it of good quality and define its responses as irrelevant and reckless. It does not only give short and non-informative answers but also gives answers that are out of context and irrelevant. Table 4 shows an example from one of the dialogues initiated with the Ubuntu chatbot.

The Wochat dialogue system seems to have a tendency to give safe and short answers which makes its answers correct but not quite informative. Users that interact with it mostly say its answers are sensible and it is just like someone who is trying to be cool. Table 5 and 6 might be a good example that demonstrates these attributes, at most all of the answers the bot gives are acceptable, however, there are no intention of trying to get the dialogue going and providing any extra information which is not desirable.

Table 5 Example of dialogue using a re-ranker trained with Wochat corpus

Speaker	Utterance
User	Hello
Iris	Hello
User	How do you feel today?
Iris	Dirty
User	Why?
Iris	An opportunity
User	Are you ready for tonight?
Iris	I'm ready.
User	What do you think about the dress I showed?
Iris	I think you need help.
User	What type of help I need?
Iris	Jean?

Table 6 Example of dialogue using a re-ranker trained with Wochat corpus

Speaker	Utterance
User	Why are you doing this?
Iris	Doing what?
User	Saying stupid things.
Iris	It's fun
User	I don't think so.
Iris	About what?
User	Your stupid things.
Iris	It's fun
User	Who cares?
Iris	I care.
User	I think I should not talk to you anymore
Iris	Well –

5 Discussion

Overall with the results from Figure 5 and all the user comments combined the Wochat bot seems to be the best one. However, there are still some improvements needed to work on it.

Regarding the problem of Wochat bot to provide short or safe answers, it can be minimized in two ways: either in the process of training the network or during the algorithm while choosing the best response.Also, checking the conversation on table 1, giving last few utterance matches rather than the whole dialogue history might as well improve the performance of the chatbot.

An improvement to the baseline system might be truncating the responses. Since the responses are taken directly from movie databases, they may sometimes be quite dramatic and long and the baseline gives mostly relevant even though too long answers. Thus truncating those responses might help increase the quality of the bot.

Another improvement on the systems might be filtering the responses retrieved from database before choosing one of them to give as a response. Since the Wochat dataset is composed of movie dialogues, which might turn out to be dramatic and long from time to time, some external filtering to answers such as removing short responses, duplicate responses and responses with undesirable words might increase the quality of the bot considerably [17].

Acknowledgments

The authors would like to thank Esra Deniz for spending their time on long discussions on the responses of the dialogue system, Kheng Hui Yeo and Benoit Matet for their help on technical aspects and insightful discussions, and the volunteers that participated in the evaluation of the system.

References

1. Serban I.V., Lowe R., Henderson P., Charlin L., Pineau J.: A Survey of Available Corpora for Building Data-Driven Dialogue Systems https://arxiv.org/abs/1512.05742,ARXIV (2017)
2. Yan R., Song Y., Wu H.: Learning to Respond with Deep Neural Networks for Retrieval-Based Human-Computer Conversation System.http://www.ruiyan.me (2016)
3. Vinyals O., Le, Q.: A Neural Conversational Model. In: ARXIV (2015)
4. Banchs R.E., Li H., IRIS: a chat-oriented dialogue system based on the vector space model.ACL '12 Proceedings of the ACL 2012 System Demonstrations. (2012)
5. Kong-Vega, N. and Shen, M. and Wang, M and D'Haro, L. F., Subjective Annotation and Evaluation of Three Different Chatbots WOCHAT: Shared Task Report. IWSDS, Singapore. (2018)
6. D'Haro, Luis F., Bayan Abu Shawar, and Zhou Yu. RE-WOCHAT 2016Shared task Description Report. Workshop on Collecting and Generating Resources for Chatbots and Conversational Agents-Development and Evaluation (RE-WOCHAT). 2016.
7. Lin, Lue, D'Haro, L.F., and Banchs, R. "A web-based platform for collection of human-chatbot interactions." Proceedings of the Fourth International Conference on Human Agent Interaction. ACM, (2016).
8. Quan V. H., Federico M. , Cettolo M. : Integrated N-best Re-ranking for Spoken Language Translation,Proc. of the 9th European Conference on Speech Communication and Technology, Interspeech05(2005)
9. Aktolga E., Allan J., Smith D.A.: Passage Reranking for Question Answering Using Syntactic Structures and Answer Types, Lecture Notes in Computer Science, vol 6611 (2011)
10. Romeo S., Martino G.D.S., ón-Cedeño A.B.,Moschitti A. ,Belinkov Y. ,Hsu W. , Zhang Y., Mohtarami M. , Glass J. R. :Neural Attention for Learning to Rank Questions in Community Question Answering, COLING (2016)
11. Wang S., Yu M. , Jiang J., Zhang W., Guo X., Chang S., Klinger Z.W.T., Tesauro G., Campbell M. : evidence aggregation for answer re-ranking in open-domain question answering, International Conference on Computational Linguistic (2018)
12. Lowe R.T. and Pow N. and Serban I.V. and Charlin L. and Liu C.W. and Pineau J.: Training end-to-end dialogue systems with the ubuntu dialogue corpus (2017)
13. D'Haro L.F., Banchs R.E. : Learning to predict the adequacy of answers in chat-oriented humanagent dialogs, Lecture Notes in Computer Science, vol 6611 (2017)

14. Lowe R., Pow N., Serban I., Pineau J.: The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems, https://arxiv.org/abs/1506.08909, ARXIV (1995)
15. Yu J., Qiu M., Jiang J., Huang J., Song S., Chu W., Chen H.: Modelling Domain Relationships for Transfer Learning on Retrieval-based Question Answering Systems in E-commerce. https://arxiv.org/abs/1711.08726, ARXIV (2017)
16. Hulth A.: Improved automatic keyword extraction given more linguistic knowledge Proceedings of EMNLP '03, ACL, 2003
17. Yusupov I. , Kuratov Y. : NIPS Conversational Intelligence Challenge 2017 Winner System: Skill-based Conversational Agent with Supervised Dialog Manager, International Conference on Computational Linguistic (2017)