Two-Step Training and Mixed Encoding-Decoding for Implementing a Generative Chatbot with a Small Dialogue Corpus

Jintae Kim[1], **Hyeon-Gu Lee**[1], **Harksoo Kim**[1], **Yeonsoo Lee**[2] and **Young-Gil Kim**[3]

[1]Kangwon National University Computer and Communication Engineering, Korea
[2]NCSOFT Corp., Korea
[3]Electronics and Telecommunications Research Institute, Korea
[1]{wlsxo1119, nlphglee, nlpdrkim}@kangwon.ac.kr
[2]yeonsoo@ncsoft.com
[3]kimyk@etri.re.kr

Abstract

Generative chatbot models based on sequence-to-sequence networks can generate natural conversation interactions if a huge dialogue corpus is used as training data. However, except for a few languages such as English and Chinese, it remains difficult to collect a large dialogue corpus. To address this problem, we propose a chatbot model using a mixture of words and syllables as encoding-decoding units. In addition, we propose a two-step training method, involving pre-training using a large non-dialogue corpus and re-training using a small dialogue corpus. In our experiments, the mixture units were shown to help reduce out-of-vocabulary (OOV) problems. Moreover, the two-step training method was effective in reducing grammatical and semantical errors in responses when the chatbot was trained using a small dialogue corpus (533,997 sentence pairs).

1 Introduction

Chatbots (also known as conversational agents, such as Alexa, Siri, and Cortana) are software programs that mimic written or spoken human speech for interactions with real people. Chatbot models are divided into two types: retrieval-based and generative models. The retrieval-based models match an input query against predefined queries, select one query with the highest matching score, and return a response paired with the selected query. They simply pick responses from a repository of query-response pairs, and therefore the responses do not contain any unplanned grammatical errors. However, the response coverage is restricted, because retrieval-based models cannot handle unseen queries for which predefined responses do not exist. To overcome this problem, generative models have been proposed with the increasing development of deep learning techniques. Generative models do not rely on predefined responses, but rather generate new responses using well-trained neural networks. Therefore, they have an ability to cope effectively with unseen queries. However, they require a large training corpus, in the form of query-response pairs. If the training corpus is not sufficient, then they make grammatical errors, especially in longer sentences.

Many previous studies on generative chatbot models are based on sequence-to-sequence networks called encoder-decoder models (Vinyals and Le, 2015; Shang et al., 2015). To furnish a chatbot with personal characteristics, Li et al. (2016b) proposed a persona-based model in which individual characteristics of speakers are encoded. However, the persona-based model required a large speaker-specific dialogue corpus for model training. To resolve this problem, Luan et al. (2017) proposed a speaker-role adaptation model based on auto-encoding methods using a non-dialogue corpus. To improve the performances of chatbots, Qiu et al. (2017) proposed a hybrid model that generates answers by selecting the most suitable among those retrieved. These previous models require a huge single-turn dialogue corpus (about ten million paired sentences) for training. For most languages, excluding a few such as English and Chinese, it is not easy to collect a high-quality dialogue corpus with millions of entries. To reduce this problem, we propose a two-step training method for efficiently training a generative chatbot model based on a sequence-to-sequence neural network. In the first step, the proposed

Proceedings of the Workshop on Intelligent Interactive Systems and Language Generation (2IS&NLG), pages 31–35,
Tilburg, The Netherlands, November 5 2018. ©2018 Association for Computational Linguistics

model is pre-trained using a large amount of non-dialogue text, such as novel texts and news articles. We call this the language learning step. In the second step, it is finaly trained using a comparably small single-turn dialogue corpus. We call this the dialogue learning step. Previous models face difficulties in dealing effectively with out-of-vocabulary (OOV) words. To reduce this problem, we propose an encoding-decoding method using a mixture of words and syllables as encoding-decoding units. The proposed model encodes and decodes closed words (i.e., general nouns and verbs) into word forms. Then, it encodes and decodes open words (i.e., proper nouns and OOV words) into syllable forms.

2 Chatbot Based on Two-Step Training Method and Mixed Encoding-Decoding Units

Figure 1 illustrates the network architecture of the proposed chatbot.

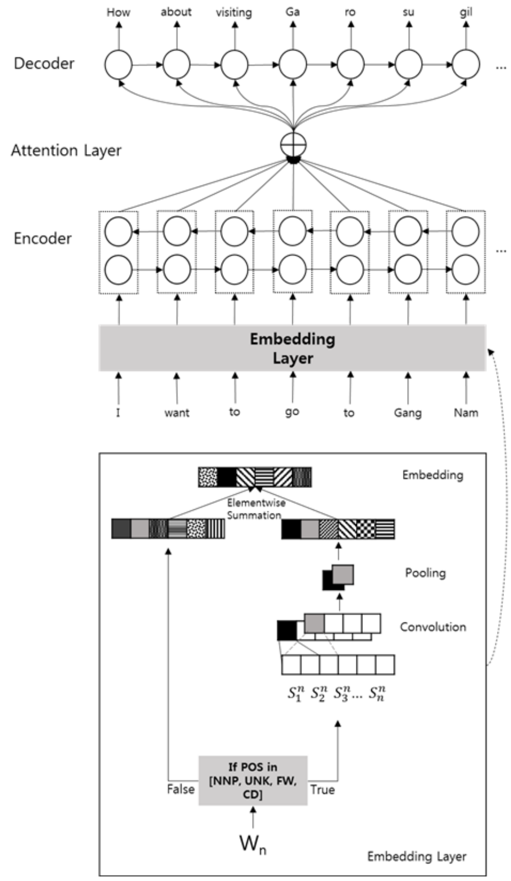

Figure 1: Overall architecture

As shown in Figure 1, the proposed chatbot is based on a sequence-to-sequence network with an attention mechanism (Bahdanau et al., 2015). This differs from conventional sequence-to-sequence networks in the aspect that the encoding and decoding units are not fixed. In the encoder, w_i is the *i*th word embedding vector in an input sentence, and s_k^j is the *j*th syllable embedding vector of the *k*th word in an input sentence. If an input word is included in a closed word category, such as general nouns, verbs, and particles, then the word is input to the sequence-to-sequence network as a word embedding vector. If an input word is not included in a closed word category, but rather in an open word category such as proper nouns or OOV words, then the word is split into syllable sequences, and is merged into an embedding vector using a convolutional neural network (Kim et al., 2016). The merged embedding vector takes the place of an embedding vector for the input word. In the decoder, f_i is the *i*th of the lexical fragments constituting a sentence. The lexical fragment f_i can be a word or syllable. Words in a closed word category are generated in the form of words, and words in an open word category are generated in the form of syllable sequences. To implement this encoding and decoding method, we perform a morphological analysis of the training corpus and split open words into syllable sequences. Then, we use the mixture of words and syllables as an input sequence and output sequence for the sequence-to-sequence network. For example, the single-turn dialogue "A: I want to go to Gangnam. B: How about visiting Garosugil?" is individually split into *[I, want, to, go, to, Gang, nam]* and *[How, about, visiting, Ga, ro, su, gil]*. The former and latter are used as an input and output of the sequence-to-sequence network, respectively.

2.1 Language Learning Step

In the language learning step, we expect that the proposed chatbot learns the grammatical structures of sentences and semantic co-relations between words in a given language. We assume that sentence mimicking can provide some assistance in achieving this goal. To realize this assumption, we adopt an autoencoder mechanism. The proposed chatbot is trained using a large non-dialogue corpus (e.g., news articles) without any turn-taking. During the training, we use each sentence in the non-dialogue corpus as input and output for the sequence-to-sequence network. As a result, the decoder plays the role of a kind of neural language model based on mimicking, which learns how to generate a grammatically and semantically correct sentence. What our model is based on mimicking is a main difference with Ramachandran's model (Ramachandran et al. (2016) based on language modeling (LM).

2.2 Dialogue Learning Step

In the dialogue learning step, we expect that the proposed chatbot learns the degree of association between two sentences in single-turn dialogues. We assume that a chatbot does not require a huge corpus of dialogue examples if it already knows how to generate sentences. To validate this assumption, the dialogue learning step of the proposed chatbot starts after the language learning step is finished. During the training, we employ pairs consisting of a query and response in single-turn dialogue as input and output pairs for the sequence-to-sequence network.

3 Evaluation

3.1 Data Sets and Experimental Settings

For our experiments, we collected two kinds of Korean corpuses: One is a non-dialogue corpus (2,975,918 sentences) consisting of news articles and online forum texts, and the other is a single-turn dialogue corpus (533,997 sentence pairs) collected from mobile chat rooms, in which two users discuss each other's views on a specific topic using the short message service of a commercial telecommunications company. To evaluate the performance of the proposed chatbot, we divided the single-turn dialogue corpus into a dialogue training corpus (499,959 sentence pairs) and a dialogue test corpus (34,038 sentence pairs). We used the whole non-dialogue corpus as training data for the language learning step. Then, we used the dialogue training corpus as training data for the dialogue learning step. The non-dialogue corpus and dialogue training corpus contained 46,334 unique closed words in total. They contained a total of 367,646 unique open words, consisting of 1,120 unique syllables. Therefore, the vocabulary size of the sequence-to-sequence network was set to 47,454 (46,334 unique closed words + 1,120 unique syllables).

We performed an automatic evaluation, as well as a manual evaluation. Automatic evaluation measures for chatbots have been not agreed universally. Portions of word-overlaps between gold-standard answers and chabot's responses are widely used as a practical choice for automatic evaluation. Therefore, we used BLEU (BiLingual Evaluation Understudy) (Papineni et al., 2002) and ROUGE (Recall-Oriented Understudy for Gisting Evaluation) (Lin et al., 2004) as automatic evaluation measures. BLEU was designed to evaluate the quality of text that has been machine-translated from one natural language to another. ROUGE was for designed for evaluating automatic summarization and machine translation software in natural language processing. BLEU and ROUGE were automatically calculated using the test dialogue corpus. It has been reported that these automatic measures may not be suitable for evaluating chatbots (Liu et al. 2016a). Thus, to supplement these evaluation measures, we manually examined outputs of the proposed chatbot from grammatical and semantic viewpoints. For the manual evaluation, we collected 100 new queries from four university students who were not involved in the research. The four students input the queries to the chatbot.–Then, they assigned scores of 0~2 points to each response generated by the chatbot, from both syntactic and semantic viewpoints, as shown in Table 1.

Score	Syntactic Score	Semantic Score
0	A response includes many grammatical errors.	A response is not associated with a query at all.
1	A response includes a few grammatical errors.	A response is partially associated with a query.
2	A response does not include any grammatical errors.	A response is fully associated with a query.

Table 1: Scores for the manual evaluation

3.2 Implementation

We implemented the proposed chatbot using TensorFlow 1.0 (Abadi et al., 2015). Training and prediction were carried out on a per-sentence level. We set the sizes of word embedding vectors and syllable embedding vectors in Figure 1 to 50 and 10, respectively. In the language learning step, the training spanned one epoch, and was performed by mini-batch stochastic gradient descent with a fixed learning rate of 0.001. Each mini-batch consisted of 32 sentences. In the dialogue learning step, the training spanned five epochs, and was performed by mini-batch stochastic gradient descent with a fixed learning rate of 0.001.

Each mini-batch consisted of 32 sentences. During the error backpropagations, the cross-entropy was used as a loss function. The optimal parameters were empirically obtained.

3.3 Experimental Results

The first experiment was designed to show the usefulness of the proposed architecture, where mixtures of words and syllables are used as input and output sequences, from the aspect of OOV problems. Table 2 shows the how the performance of the proposed chatbot varies according to changes in encoding-decoding units.

Measure	Word-Only	Syllable-Only	Mixture
Vocabulary Size	57,102	1,534	55,568
Training Time (h)	2.3	3.2	2.9
BLEU	0.3693 (0.4646)	0.4394 (0.4234)	0.4230 (0.4710)
ROUGE-1	0.2840 (0.3646)	0.4279 (0.4026)	0.3654 (0.4194)
ROUGE-L	0.2657 (0.3861)	0.4177 (0.3920)	0.3518 (0.4009)
Syntactic Score	0.43	0.98	0.70
Semantic Score	0.62	1.26	0.98

Table 2: Performance comparison according to different encoding and decoding units

In Table 2, *Mixture* is the proposed model. *Word-Only* and *Syllable-Only* represent chatbots that use only words and syllables, respectively, as encoding-decoding units. The parenthesized scores are the performances when functional words (i.e., ending words, postpositional words, and so on) in Korean are excluded from the performance evaluations. In other words, they are the performances with respect to generated content words (i.e., nouns, verbs, and so on). All of the models were trained using only the dialogue training corpus, like conventional chatbot models. As shown in Table 2, *Mixture* exhibited a better performance than *Word-Only* for all measures. Although *Mixture* achieved an inferior performance to *Syllable-Only* for the measures with respect to all types of words, it outperformed *Syllable-Only* for the measures with respect to just content words. We found that *Word-Only* and *Syllable-Only* made many mistakes in generating content words that were unseen in the training data. This fact indirectly shows that the proposed architecture may contribute to reducing OOV problems. We analyzed the cases in which *Mixture* showed lower syntactic and semantic scores than *Syllable-Only*. The reasons are as follows: *Syllable-Only* showed relatively high syntactic and semantic scores because it often returns short and general responses like “Okay” and “Yes, I see”. Moreover, *Syllable-Only* more correctly generated functional words than *Word-Only* and *Mixture* did. As a result, *Syllable-Only* obtained higher syntactic and semantic scores. Although *Mixture* well generated content words, it showed lower syntactic and semantic scores than *Syllable-Only* because it less correctly generated functional words.

The second experiment was designed to show the effectiveness of the proposed training method. Table 3 shows how the performance of the proposed chatbot varies according to different training methods. In Table 3, *Single-Step Training* is a conventional training method in which a chatbot is trained using only the dialogue training corpus. *LM Training* is the training method proposed by Ramachandran et al. (2016). In LM Training, a chatbot is pre-trained using a language model, and re-trained using the dialogue training corpus. *Two-Step Training* is the proposed method, in which the chatbot is pre-trained using the non-dialogue corpus and re-trained using the dialogue training corpus. The parenthesized scores give the performances when functional words are excluded from the performance evaluations.

Measure	Single-Step Training	LM Training	Two-Step Training
BLEU	0.4230 (0.4710)	0.4592 (0.5094)	0.4591 (0.5076)
ROUGE-1	0.3654 (0.4194)	0.3833 (0.4231)	0.4045 (0.4673)
ROUGE-L	0.3518 (0.4009)	0.3858 (0.4379)	0.4004 (0.4666)
Syn. Score	0.70	0.81	0.94
Sem. Score	0.98	1.09	1.30

Table 3: Performance comparison according to different training methods

As shown in Table 3, *Two-Step Training* outperformed *Single-Step Training* for most measures and showed competitive performances compared with *LM Training* for the ROUGE scores of content words. In particular, *Two-Step Training* achieved much higher scores than *Single-Step Training* and *LM Training* in the manual evaluation. This fact reveals that the proposed training method can be effective in reducing grammatical errors of responses and in generating responses associated with input queries when a dialogue corpus is not sufficient to train chatbots based on sequence-to-sequence networks.

4 Conclusion

We have proposed a chatbot model using a modified architecture of a sequence-to-sequence network. The chatbot used a mixture of words and syllables as encoding-decoding units, in order to reduce OOV problems. In addition, we proposed a new training method to pre-train the chatbot using a large non-dialogue corpus, and to re-train the chatbot using a small dialogue corpus. This training method contributed to reducing syntactic and semantic mistakes when a dialogue corpus for training is not large enough.

Acknowledgments

This work was supported by 2017 research grant funded by NCSOFT corp. and Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (R0126-15-1117, Core technology development of the spontaneous speech dialogue processing for the language learning). In addition, this work was partly supported by the National Research Foundation of Korea Grant funded by the Korean Government (NRF-2017M3C4A7068188).

References

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

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

Li, J., Galley, M., Brockett, C., Spithourakis, G. P., Gao, J., & Dolan, B. (2016). A persona-based neural conversation model. *arXiv preprint arXiv:1603.06155*.

Luan, Y., Brockett, C., Dolan, B., Gao, J., & Galley, M. (2017). Multi-Task Learning for Speaker-Role Adaptation in Neural Conversation Models. *arXiv preprint arXiv:1710.07388*.

Qiu, M., Li, F. L., Wang, S., Gao, X., Chen, Y., Zhao, W., ... & Chu, W. (2017). Alime chat: A sequence to sequence and rerank based chatbot engine. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (Vol. 2, pp. 498-503).

Kim, Y., Jernite, Y., Sontag, D., & Rush, A. M. (2016, February). Character-Aware Neural Language Models. In *AAAI* (pp. 2741-2749).

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

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 on association for computational linguistics* (pp. 311-318). Association for Computational Linguistics.

Lin, C. Y. (2004). Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*.

Liu, C. W., Lowe, R., Serban, I. V., Noseworthy, M., Charlin, L., & Pineau, J. (2016). How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023*.

M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane,´ R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. War- ´ den, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. (2016, November). TensorFlow: A System for Large-Scale Machine Learning. In *OSDI* (Vol. 16, pp. 265-283).

Ramachandran, P., Liu, P. J., & Le, Q. V. (2016). Unsupervised pretraining for sequence to sequence learning. *arXiv preprint arXiv:1611.02683*.