INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Enhancing Semantic Understanding with Self-supervised Methods for Abstractive Dialogue Summarization

Hyunjae Lee, Jaewoong Yun, Hyunjin Choi, Seongho Joe, Youngjune L. Gwon

AI Research Center, Samsung SDS, South Korea
{h8.lee, jw0531.yun, hjjin.choi, drizzle.cho, gyj.gwon}@samsung.com

Abstract

Contextualized word embeddings can lead to state-of-the-art performances in natural language understanding. Recently, a pre-trained deep contextualized text encoder such as BERT has shown its potential in improving natural language tasks including abstractive summarization. Existing approaches in dialogue summarization focus on incorporating a large language model into summarization task trained on large-scale corpora consisting of news articles rather than dialogues of multiple speakers. In this paper, we introduce self-supervised methods to compensate shortcomings to train a dialogue summarization model. Our principle is to detect incoherent information flows using pretext dialogue text to enhance BERT's ability to contextualize the dialogue text representations. We build and fine-tune an abstractive dialogue summarization model on a shared encoder-decoder architecture using the enhanced BERT. We empirically evaluate our abstractive dialogue summarizer with the SAMSum corpus, a recently introduced dataset with abstractive dialogue summaries. All of our methods have contributed improvements to abstractive summary measured in ROUGE scores. Through an extensive ablation study, we also present a sensitivity analysis to critical model hyperparameters, probabilities of switching utterances and masking interlocutors.

Index Terms: abstractive dialogue summarization, self-supervised learning, BERT, BERT2BERT

1. Introduction

In natural language processing, abstractive summarization generates a concise summary for lengthy source text using words that do not necessarily appear in the source. Such creative aspect (in comparison with extractive summarization) makes abstractive summarization one of the most challenging tasks in computational linguistics. In speech, dialogue summarization enables a useful capability to capture salient information scattered in a dialogue containing the utterances by multiple interlocutors and rewrite them into simplified, easy-to-grasp text. With the rapid growth of online communications, providing dialogue summaries is becoming one of the most important features in a speech system. The recent outbreak of the coronavirus pandemic (COVID-19) and other on-going global incidents demand a crisp summary of long speech conversations more useful and appealing than ever.

Self-supervised learning has been used widely to complement or replace entirely human-annotated datasets in training deep networks of language, speech, and visual models. There are numerous approaches for neural dialogue summarization, yet not many have aimed to improve semantic and structural understanding of dialogue with a specifically-designed self-supervised learning method.

In this paper, we propose self-supervised methods for training a neural abstractive dialogue summarization model. We have designed pretext tasks that require the model under training predict whether there are incorrect ordering and irrelevant information in utterances or not. There are also tasks to predict switched and masked interlocutor names. When pre-training with our self-supervised methods, the model seems to learn a better understanding of the semantic relevance between interlocutor and utterance. This can be crucial to capture the essence of a whole dialogue in much shortened text while preserving the most salient information.

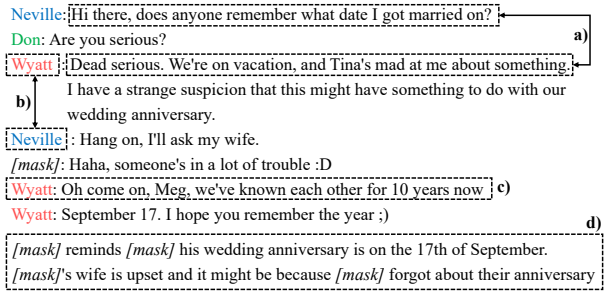

Figure 1: *An overview of proposed self-supervised methods. They are a) switching utterance , b) switching interlocutor , c) inserting utterance , d) masking interlocutor methods. Each method is carried out separately. (Note that some methods can be combined to run beneficially.)*

In Figure 1, we describe an illustrative example about how our self-supervised methods take place during the pre-training of a neural language model upon which abstractive summarization task can be fine-tuned. There are four self-supervised methods, namely switching utterance, switching interlocutor, inserting utterance, and masking interlocutor. Notice that three out of the four self-supervised methods are set up as a simple binary classification problem (*i.e.*, corrupted or not). In masking interlocutor method, mapping to a correct interlocutor's name would be required instead. We remark that our proposed method is fully compatible with a publicly available pre-trained language model like BERT [1]. This makes our approach more appealing and retrofitting in the popular paradigm comprising the pre-training and fine-tuning stages for building contemporary NLP applications.

To construct a neural abstractive dialogue summarizer, we enhance a pre-trained BERT with our self-supervised methods and use it as both encoder and decoder in sequence-to-sequence model [2, 3] while sharing the weights between the encoder and the decoder (see Figure 2). We fine-tune and evaluate empirically our BERT-based summarizer using the SAMSum corpus [4]. Our self-supervised methods indicate a significantly improved performance compared to the baseline (BERTSHARE), which is using the pre-trained BERT as is (without applying the

http://dx.doi.org/10.21437/Interspeech.2021-1270

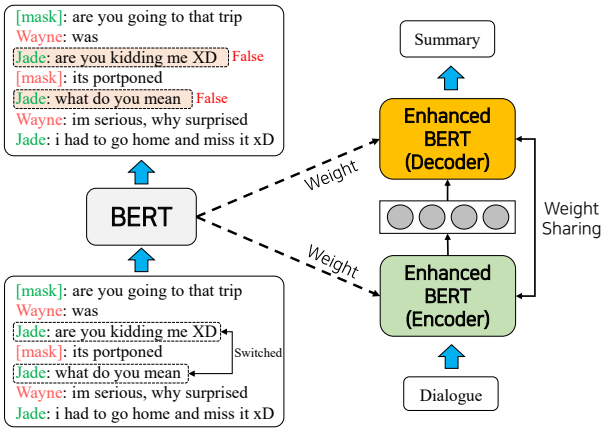

Figure 2: *Illustration of our approach on abstractive summarization task. First, we enhance dialogue context understanding of BERT via (a) proposed self-supervised methods. Then, we initialize the traditional encoder-decoder model with enhanced BERT and fine-tune on abstractive summarization task.*

proposed self-supervised methods).

2. Related Work

2.1. Self-supervised pre-training for text summarization

In recent years, self-supervised learning has pushed the performance of a wide range of natural language processing (NLP) tasks to new state-of-the-art and becoming a dominant paradigm in NLP. Numerous pre-training methods based on self-supervised learning for text summarization have been introduced. A common approach lets a model predict the original input tokens from randomly masked tokens (or sentences) in a document to resemble a target downstream task, *i.e.*, abstractive summarization [5, 6, 7]. Zhang *et al.* [8] adopt masked sentence prediction in pre-training stage and employed it as a sentence encoder for extractive summarization on large-scale news corpus. The most similar approach to ours is Wang *et al.* [9], where they introduced three self-supervised pre-training tasks for extractive summarization with CNN/DM datasets [10]. Compared to previous work, our approach focuses more on how to incorporate the heterogeneous attributes of a dialogue to self-supervised methods in order to overcome the challenge that fine-tuning is often unstable on small datasets and causes performance degradation [3].

2.2. Leveraging pre-trained model for text summarization

Liu & Lapata [11] has shown that BERT can beneficially be applied to both extractive and abstractive document summarization. For abstractive task, their model consists of a BERT pre-trained on extractive summarization task as the encoder and randomly initialized the 6-layer Transformer blocks for the decoder. On the contrary, we aim to leverage the full power of the proposed self-supervised methods by employing BERT as both an encoder and a decoder, which is pre-trained on the task destined to enhance semantic understanding of a dialogue. In recent empirical studies by Google Research [3, 12], it is possible to achieve state-of-the art results on text summarization without any auxiliary task with the encoder-decoder network utilizing pre-trained BERT, RoBERTa, GPT-2 (so-called BERT2BERT, BERT2GPT2). Our implementation of BERT2BERT architecture, however, could not have reached higher performance than baselines because the dataset of our choice is much smaller (about 10 times smaller) than previous work. This means that BERT2BERT architecture has a great potential for a nice warm-starting model but may not be sufficient for low-resource datasets to achieve a better performance.

2.3. Abstractive dialogue summarization

We have chosen a dialogue dataset of messenger-like natural conversations that have very distinct features from formally-written styled documents like a news article. Most salient pieces of conversations are scattered across the utterances by multiple interlocutors, and it makes difficult to decide what the key point of a dialogue is. Moreover, There are no large enough annotated datasets for abstractive dialogue summarization to train deep neural generative models. To address these problems, Ganesh and Dingliwal [13] propose a two-phase pipeline method that uses discourse labels and an existing document summarizer in the zero-shot learning perspectives. Feng *et al.* [14] propose the first to incorporate commonsense knowledge into abstractive dialogue summarization with a graph neural net that includes both utterance and knowledge nodes. Another approach using the graph structure for dialogue summarization is in Zhao *et al.* [15]. They have tackled the problem of previous sequence-to-sequence models [2] about not paying an attention to handle the sentence-level long-distance dependency and capture the cross-sentence relations by proposing a method that can construct the whole dialogue as a graph for abastractive dialogue summarization. These two approaches constitute the most recent work on the SAMSum corpus. In this work, we choose them as the baselines to compare our model's score with.

3. Methods

In this section, we describe how to enhance BERT's semantic understanding of dialogue in self-supervised fashion.

- **Switching Utterance**: Similar with previous works [16, 17] of predicting switched sentences, this task is to predict whether each utterance (not sentence) is switched or not. We switch some utterances selected with the probability P_u. Hence, the number of switched utterances changes dynamically at each training step. At the same time, we mask some interlocutor's name with the probabilty P_n. This optional constraint of input utterances makes the task more challenging.
- **Switching Interlocutor**: For this task, we switch interlocutor's name in utterances instead of utterances with the probabilty P_i and we did not mask the name of interlocutors. Additionally, we concatenate corresponding reference summary with the utterances for each input sequence, so it can help the model find a mismatch between interlocutor and what they said.
- **Inserting Utterance**: For this task, we insert K utterances from other dialogues selected with a pre-defined probability into randomly selected K positions from inter-utterances and the model predict whether each utterance is from other dialogue or not. If interlocutor's name in inserted utterances remains same as the original, which look obviously unfamiliar, the task has a lit-

Table 1: *The number of each components of SAMSum corpus, where "Dial." stands for "Dialogue.", "Utter." for "Utterance", "Inter." for "Interlocutor's name" appearing in dialogues. "OOV." is short for the number (proportion) of utterance that contains at least one out of vocabulary token which is mostly facial emojis such as in "Nadine: caaaaaat 😻😻😻😻😻😻😻" , and "OOV. w/ FE" for the case "with facial emoji appended vocabulary".*

Type	# Dial.	# Utter.	# Inter.	# OOV.	# OOV. w/ FE
Train	14,732	164,505	4,289	3,315 (2.0%)	1,165 (0.7%)
Valid	819	8,860	894	179 (2.0%)	44 (0.4%)
Test	818	9,212	912	179 (1.9%)	82 (0.9%)

tle worth to solve. Hence, we replace the name of them properly to camouflage where they come from.

- **Masking Interlocutor**: This task is similar to masked language modeling [1] task which a model needs to recover masked token from a vocabulary. In our work, we only masked interlocutor's name appeared in reference summary. Then, the model predicts masked names using the information of utterances.

4. Experiments

4.1. Dataset

We evaluate our approach on the SAMSum corpus [4] that is constructed by linguists fluent in English, which contains over 16k chat dialogues consists of over 182K utterances (see Table 1 for statistics on dataset). Each utterance has the specified format that a colon at the beginning of an utterance separates the interlocutor's name and their speaking content. And also, we append 311 facial emojis into our vocab that appear most often in the train dataset in order to avoid including too many *[unk]* in the input sequence. The effect of appended tokens is on Table 2.

4.2. Implementation and training details

All our experiment start from publicly available 'bert-base-uncased' version. For self-supervised tasks, we preprocess the dialogue datasets for our experiments following steps: a) add *[SEP]* token at every end of utterance which improves structural information of dialogue [15] and is used as the utterance representation in our self-supervised methods, b) replace each name with a single token included in our vocabulary to simplify the problem so that model can learn the semantic relations between the subject and action ("Who did What"), rather than distribution of name tokens. Then, we add a linear layer with dropout (0.1) on top of BERT, where *[SEP]* and *[MASK]* tokens are fed into for classification task. Throughout abstractive summarization task, we use the shared encoder-decoder architecture (so called BERTSHARE in [3]) where both model's weights are shared and initialized with BERT trained by one of proposed self-supervised methods. Note that weight sharing between encoder and decoder is necessary rather than optional in that it reduces model parameters and improves the final performance as well in our experimental setup.

The hyper-parameters of self-supervised learning and summarization are almost identical except training epochs. We use the AdamW optimizer [18] for both self-supervised and summarization task, with batch size 128, input sequence length 512 and learning rate from 2e-5 to 5e-5, warmup steps are 500. For self-supervised learning, we train the model until the train loss converged (upper bounded by 5K steps). After that, we finetune the model for summarization task until the validation loss converges.

Table 2: *Results in terms of ROUGE metric on the SAMSum corpus test set.*

Model	R-1	R-2	R-L	R-AVG
LONGEST-3	32.46	10.27	29.92	24.22
Transformer [19]	37.27	10.76	32.73	26.92
Fast Abs RL Enhanced [20]	41.95	18.06	39.23	33.08
D-HGN [14]	42.03	18.07	39.56	33.22
TGDGA [15]	43.11	**19.15**	40.49	34.25
BERTSHARE	39.07	12.74	36.05	29.29
+ Facial Emoji	39.97	13.7	36.42	30.03
w/ Masking Interlocutor	40.29	13.91	37.46	30.55
w/ Inserting Utterance	44.17	18.76	41.68	34.87
w/ Switching Interlocutor	44.01	18.03	41.42	34.49
w/ Switching Utterance	**44.78**	19.12	**42.21**	**35.37**

4.3. Evaluation Metrics

ROUGE [21] is one of standard measures to evaluate machine generated text over many natural language processing fields. However, the metric based on only n-gram overlapping may not be the best choice for abstractive dialogue summarization [4]. Such measure is lacking aspects of fluency, intelligibility, and repetition [22]. In order to address this issue, we also report consine-similarity between model's prediction (i.e. generated summary) and ground truth using the sentence encoder [23], 'stsb-roberta-large (available at https://www.sbert.net/)' fine-tuned on Semantic Text Similarity dataset. More precisely, we use only the longest 100 samples for calculating cosine-similarity in order to uncover the differences between good and poor quality summaries more clearly.

5. Results

5.1. Evaluation of Self-supervised Learning

Table 2 shows the result of the proposed self-supervised learning methods on SAMSum dataset. The upper part of Table 2 is the scores reported in each paper or [4]. We use the official PyRouge package (https://pypi.org/project/py-rouge) to compute the ROUGE score. In contrast to BERTSHARE 's promising results [3] on a large corpus of news like CNN/DM, it shows the relatively lower performance on our dataset compared to recent strong baselines. However, all combinations of BERTSHARE and our methods improve performance dramatically by up to 6.08% (Switching Utterance) in averaged ROUGE score.

On the other hand, Masking Interlocutor methods shows the poor result compared to other methods. This is because the generative training objective that predicts masked tokens from thousands of candidates usually requires much longer training steps with a larger corpus than other binary decision task (i.e. switched or not), and we will investigate proper training setup in future work.

In addition, we confirm that the inclusion of facial emojis in the vocabulary is advantageous for abstractive summarization task, even though they do not appear in the reference summary at all. This is also evidence that if the encoder is enhanced and is employed as decoder too, it leads to better performance on sequence generation task.

Table 3: *An ablation study for two options in three self-supervised methods on SAMSum corpus test set.*

Method	Reference	Interlocutor	R-1	R-2	R-L	R-AVG	COS.
Switching Interlocutor	-	✓	44.01	18.03	41.42	34.49	0.6289
	✓	✓	42.96	17.74	40.84	33.85	0.6453
Switching Utterance	✓	✓	44.04	18.61	41.52	34.72	**0.6521**
	-	✓	44.21	18.64	41.94	34.93	0.6396
	✓	-	43.5	18.14	41.22	34.29	0.6498
	-	-	**44.78**	**19.12**	**42.21**	**35.37**	0.6367
Inserting Utterance	✓	✓	43.38	18.04	41.02	34.15	0.6401
	-	✓	44.17	18.76	41.68	34.87	0.6292
	✓	-	43.14	17.28	40.5	33.64	0.6428
	-	-	43.43	18.25	41.19	34.29	0.6431

5.2. Ablation Test

5.2.1. Adding reference summary or elimination of interlocutor's name

In the three among our proposed methods, the reference summary (gold label) and interlocutor's name can be added or removed freely. For example, a model can detect the inconsistent utterance order (Switch Utterance) or mismatch of ⟨interlocutor, utterance⟩ pair (Switch Interlocutor) by only using the information flows in utterances without summary and interlocutor's name. We conduct ablation studies to investigate how the model incorporates these additional information and the result is on Table 3. From the result, adding these two factors does not lead to better results in most cases in terms of ROUGE score. Note that the highest ROUGE score achieved without any factors.

Interestingly, in most cases, using the concatenation of summary and dialogue as the input sequence for our three methods helps achieve higher scores in terms of cosine-similarity of reference summaries and generated ones. It implies that there exists inconsistency between ROUGE and cosine-similarity metric and reference summary can help a model learn useful semantic information to generate better summary.

5.2.2. Probability of Switching and Masking Name

We also investigate the effects of probability of switching, P_u and masking name, P_n for Switching Utterance task. We set the P_u a range of [0.33, 1] and (0, 1] for P_n. Even if P_u is 1.0, that does not mean all utterances are switched, because there are chances to restore the original order of utterances while reordering utterances randomly. The numbers in Figure 3 refer to the average score of ROUGE-1, 2 and L measure, obtained from the model pre-trained by Switching Utterance method with each probability and fine-tuned on SAMSum dataset for abstractive summarization. As shown in Figure 3, we found that combination of two probabilities on the edges in the matrix, i.e. (1.0, 1.0), (1.0, 0.0), (0.33, 0), tends to show better performance. On the other hand, the combination of 0.5 and 0.5 yields the worst result.

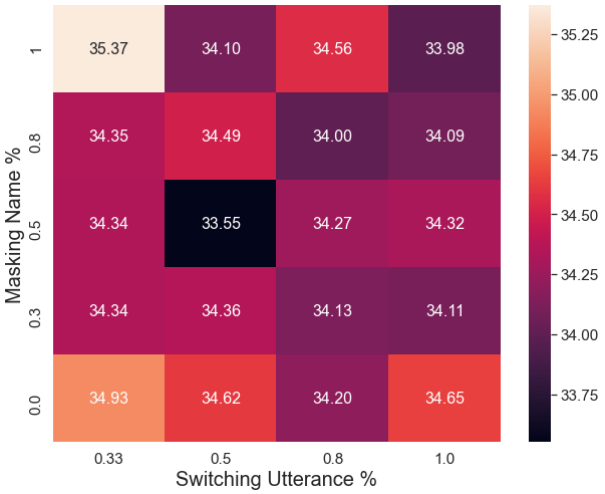

Figure 3: *Ablation results of Switching Utterance method according to combinations of two probabilities in terms of average ROUGE scores.*

6. Conclusion

We proposed four self-supervised methods to enhance semantic understanding of conversational text by multiple interlocutors for abstractive dialogue summarization. The methods, switching utterance, switching interlocutor, inserting utterance, and masking interlocutor, specifically strengthen the learning of structural and component information in dialogues. By enhancing an off-the-shelf pre-trained BERT with our methods, we build an abstractive summarizer in a shared encoder-decoder architecture for sequence-to-sequence training. Our experimental results on the SAMSum corpus indicate a substantive improvement measured in ROUGE scores. Through a careful ablation study, we provide a practical insight for the crucial hyperparameter settings of the proposed self-supervised methods.

We believe there is still much room for improvements in our setting. We used only 'bert-base-uncased' version to prove our concept, "Can intuitively enhanced representation of encoder produce better performance in the sequence generation task?". Nevertheless, this base model has already achieved promising results so that it will inspire more investigation into engaging large language models with more sophisticated experimental setup.

7. References

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational*

Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. [Online]. Available: https://www.aclweb.org/anthology/N19-1423

[2] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2015.

[3] S. Rothe, S. Narayan, and A. Severyn, "Leveraging pre-trained checkpoints for sequence generation tasks," *Transactions of the Association for Computational Linguistics*, vol. 8, 2020.

[4] B. Gliwa, I. Mochol, M. Biesek, and A. Wawer, "Samsum corpus: A human-annotated dialogue dataset for abstractive summarization," in *Proceedings of the 2nd Workshop on New Frontiers in Summarization*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 70–79. [Online]. Available: https://www.aclweb.org/anthology/D19-5409

[5] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 7871–7880. [Online]. Available: https://www.aclweb.org/anthology/2020.acl-main.703

[6] J. Zhang, Y. Zhao, M. Saleh, and P. Liu, "PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, H. D. III and A. Singh, Eds., vol. 119. PMLR, 13–18 Jul 2020, pp. 11 328–11 339. [Online]. Available: http://proceedings.mlr.press/v119/zhang20ae.html

[7] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. Liu, "Mass: Masked sequence to sequence pre-training for language generation," in *International Conference on Machine Learning*, 2019, pp. 5926–5936.

[8] X. Zhang, F. Wei, and M. Zhou, "HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Jul. 2019.

[9] H. Wang, X. Wang, W. Xiong, M. Yu, X. Guo, S. Chang, and W. Y. Wang, "Self-supervised learning for contextualized extractive summarization," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Jul. 2019.

[10] A. See, P. J. Liu, and C. D. Manning, "Get to the point: Summarization with pointer-generator networks," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 1073–1083. [Online]. Available: https://www.aclweb.org/anthology/P17-1099

[11] Y. Liu and M. Lapata, "Text summarization with pretrained encoders," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Nov. 2019.

[12] S. Goodman, Z. Lan, and R. Soricut, "Multi-stage pretraining for abstractive summarization," *CoRR*, vol. abs/1909.10599, 2019. [Online]. Available: http://arxiv.org/abs/1909.10599

[13] P. Ganesh and S. Dingliwal, "Restructuring conversations using discourse relations for zero-shot abstractive dialogue summarization," 2020.

[14] X. Feng, X. Feng, B. Qin, and T. Liu, "Incorporating commonsense knowledge into abstractive dialogue summarization via heterogeneous graph networks," *CoRR*, vol. abs/2010.10044, 2020. [Online]. Available: https://arxiv.org/abs/2010.10044

[15] L. Zhao, W. Xu, and J. Guo, "Improving abstractive dialogue summarization with graph structures and topic words," in *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain (Online): International Committee on Computational Linguistics, Dec. 2020, pp. 437–449. [Online]. Available: https://www.aclweb.org/anthology/2020.coling-main.39

[16] J. Wu, X. Wang, and W. Y. Wang, "Self-supervised dialogue learning," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 3857–3867. [Online]. Available: https://www.aclweb.org/anthology/P19-1375

[17] L. Logeswaran, H. Lee, and D. Radev, "Sentence ordering and coherence modeling using recurrent neural networks," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/11997

[18] I. Loshchilov and F. Hutter, "Fixing weight decay regularization in adam," 11 2017.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017. [Online]. Available: https://arxiv.org/pdf/1706.03762.pdf

[20] Y.-C. Chen and M. Bansal, "Fast abstractive summarization with reinforce-selected sentence rewriting," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 675–686. [Online]. Available: https://www.aclweb.org/anthology/P18-1063

[21] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81. [Online]. Available: https://www.aclweb.org/anthology/W04-1013

[22] P. Ganesh and S. Dingliwal, "Abstractive summarization of spoken and written conversation," *CoRR*, vol. abs/1902.01615, 2019. [Online]. Available: http://arxiv.org/abs/1902.01615

[23] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. [Online]. Available: http://arxiv.org/abs/1908.10084