Generating Natural Language Dialogues using Large Language Models with Adapters*

Ellen Zhang Chang[1], Ole Jakob Mengshoel[1,*]

[1]Norwegian University of Science and Technology (NTNU), Høgskoleringen 1, 7034 Trondheim, Norway

Abstract

Communication through natural language dialogue has enabled humans to satisfy social needs, share knowledge, collaboratively solve problems, and negotiate compromises. Unfortunately, language barriers can hinder such communication. Thus, language education grounded in dialogue is essential for those with urgent needs, e.g., refugees or immigrants, and those who desire longer-term careers in an environment with an unfamiliar language. In many cases, there is a need to learn a domain-specific subset of a language for people to communicate in a new language environment.

This work presents a method of creating dialogue for language education through surrounding dialogue generation. Surrounding dialogue generation, as defined, is a complex natural language generation problem. We propose a novel deep learning architecture with adapters; it generates domain-specific conversational dialogues using an open-source pre-trained language model used in several state-of-the-art architectures, namely GPT-2. Its successors, GPT-3 and GPT-4, have shown even more remarkable results. But they are, for a third party, impossible to add adapters to. Our architecture extends a dialogue by generating preceding and following utterances. Several experiments are done to validate the benefits of the architecture as a creative tool for educators. The experiments show promising results and provide a basis for future research.

Keywords

Natural Language Processing, Natural Language Dialogues, Large Language Models, Computer-Assisted Language Learning, Adapter-Based Tuning

1. Introduction

Context. Artificial Neural Networks (ANNs) have been used to learn natural language from large amounts of textual data represented as word embeddings [1, 2, 3, 4]. Sequence-to-sequence (seq2seq) is an encoder-decoder structure commonly consisting of Recurrent Neural Networks (RNNs) [5]. Seq2seq forms the basis of several state-of-the-art language models with numerous applications, e.g., document classification [6], dialogue systems [7], sentiment analysis [8], and opinion mining [9]. These applications solve downstream tasks by adapting pre-trained language models. Vaswani et al. [10] demonstrate that their Transformer ANN architecture, using only an attention mechanism [11], is state-of-the-art. Transformer models employ a seq2seq structure in a Deep Neural Network (DNN) architecture for data-driven language translation [10]. GPT-2 (Generative Pre-trained Transformer, version 2) [2], its predecessor GPT [12], and its successors,

NAIS 2023: The 2023 symposium of the Norwegian AI Society, June 14-15, 2023, Bergen, Norway.

*Corresponding author.

obcellen@gmail.com (E. Z. Chang); ole.j.mengshoel@ntnu.no (O. J. Mengshoel)

https://www.ntnu.no/ansatte/ole.j.mengshoel (O. J. Mengshoel)

0009-0008-9746-8879 (E. Z. Chang); 0000-0003-2666-5310 (O. J. Mengshoel)

GPT-3 [1] and GPT-4 [13], are all causal language models that use the concept of left context (previous text) to predict the next word. Kiros et al. [14] describe an encoder for predicting future sentences and previous sentences of a context. We are in this work interested in using large language models for exercise generation [15, 16, 17, 18], specifically generation of exercises for learning foreign languages [15, 18].

Challenges. Technological advancements and globalization have increased the demand for understanding and speaking multiple languages. For example, the refugee crisis due to the Russo-Ukrainian conflict has uncovered that language barriers are a reason for weak provider-patient communication between healthcare workers and Ukrainian refugees [19]. Developing broad and deep language competence is difficult and time-consuming for a language learner. Ultimately, language learners should focus on their most relevant domains of interest, which typically include their work or professional interests. A language educator must thus craft a curriculum that balances domain-specific content and regular language curricula of appropriate difficulty. This is challenging. Although substantial progress has been made, it is also a challenge for machine learning, including DNN methods, to create such high-quality educational materials. Many applications, including document classification [6], dialogue systems [7], sentiment analysis [8], and opinion mining [9], solve downstream tasks by fine-tuning or adapting pre-trained language models. However, given the large model and dataset sizes, this often comes at a very high computational cost.

Contributions. We define the research question as the following: *how can a simple creative data-driven tool, powered by computationally efficient machine learning, assist an educator in writing dialogues for language and communication training*? To assist educators in assembling scalable domain-specific dialogues, a novel architecture called the BFD (Backward-Forward Dialogue) Generator is developed in this work. The architecture consists of multiple collaborative components utilizing Machine Learning (ML), Natural Language Processing (NLP), and Deep Learning (DL). The BFD Generator produces domain-specific dialogues using GPT-2 tuned using adapters [20] on the conversational dialogue corpus WIZARD OF WIKIPEDIA [21]. The BFD Generator extends a brief input dialogue by generating preceding and following utterances.

The following points provide a summary of our main contributions:[1]

1. A formulation of the surrounding dialogue generation problem, motivated by a need to create domain-centric language education dialogues.
2. A novel architecture, the Backward-Forward Dialogue Generator, that produces surrounding dialogue from a dialogue snippet.
3. A computationally efficient and modular transfer learning approach underlying the Backward-Forward Dialogue Generator, generating utterances using adapter-based tuning of a large-scale language model.
4. Experimental results demonstrating the value of the Backward-Forward Dialogue Generator as a creative tool that aids educators in writing dialogues for language learning exercises.

This paper defines the dialogue generation problem in Section 2. In Section 3, we present related work on dialogue systems. The proposed architecture of the BFD Generator is presented in Section 4. A discussion of the experimental results is in Section 5. We conclude this paper and reflect upon future work in Section 6.

[1]This paper builds upon the MS thesis of Ellen Zhang Chang [22].

2. The Surrounding Dialogue Generation Problem

A dialogue is usually structured such that each speaker takes turns making utterances [23]. Two consecutive turns between different speakers make up an exchange. Multiple exchanges are considered a dialogue. After conversing with someone, only some parts (or snippets) of the dialogue may be memorable. From a snippet, one can speculate about the rest of the dialogue. This is the intuition behind our problem description and approach. This section introduces our dialogue generation problem and a gold standard for human evaluation of dialogues.

2.1. Problem Formulation

The *Surrounding Dialogue Generation Problem* (SDGP) is defined as follows. Given a dialogue snippet c (involving two speakers s_1 and s_2), its topic t, and a length l, generate an extended dialogue c' that is on topic t, has l number of turns, and contains the dialogue snippet c. SDGP is a challenging problem, given the current state-of-the-art in natural language processing (NLP). This has implications for solving it, as discussed in Section 4, and how the input and output are treated. The SDGP is part of a more extensive workflow for teaching language, used as a creative tool by educators. That workflow is essential to remember but is not reflected in Figure 1 due to lack of space.

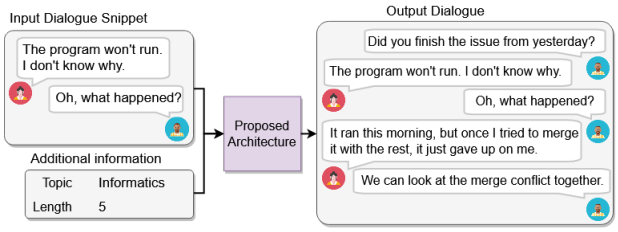


Figure 1: The proposed SDGP architecture takes a dialogue snippet and additional information (topic, dialogue length) as input to generate a more extensive dialogue as output.

Solving the SDGP amounts to generating a dialogue from a dialogue snippet, topic, and dialogue length, as illustrated in Figure 1. This differs from the work by Kiros et al. [14] concerning a model that reconstructs surrounding sentences within the book domain, not the dialogue domain. We are interested in machine learning methods to solve the SDGP, where dialogue datasets are the basis for generating surrounding dialogues.

2.2. Gold-Standard Human Evaluation

Human evaluation is the gold standard for evaluating dialogues, including those discussed in Section 2.1. Humans assess the quality of generated dialogues using metrics like these:

1. **Sensibleness.** How well the utterances make sense in the dialogue context [24].
2. **Specificity.** How specific the utterances in the dialogue are [24].
3. **Interestingness.** How interesting the dialogue is [3].
4. **Informativeness.** The percentage of responses that carry information on the external world that can be supported by the other utterances in the dialogue [3].
5. **Groundedness.** The percentage of utterances that carry information on the external world that can be supported by external sources [3].

6. **Teachability.** Since we focus on language learning, we propose a new metric, teachability, compared to LaMDA [3]. We consider the Common European Framework of Reference for Languages (CEFR), an international standard for describing language proficiency. CEFR organizes language proficiency in six levels, A1 to C2 [25]. Teachability seeks to measure how well the specific dialogue can be understood at a given CEFR proficiency level.

We use these metrics in experiments in Section 5.

3. Related Work

A dialogue system is, for our purposes, a computer system capable of having a dialogue with a user. Three broad classes of dialogue systems are found in the literature: task-oriented, conversational, or question-answering [26]. This work focuses on task-oriented and conversational dialogue systems. We discuss them below, along with how such systems are evaluated and their fit into computer-assisted language learning.

Research on reliable, cheap, and general automatic metrics for **evaluation of dialogue systems** is an active research area [26]. Today, no automated metrics can compete with human judgement. While human decisions are often used in dialogue evaluations, they are expensive and not always reliable. The value of human reviews is further seen in LaMDA [3]. Human judgement is used to generate, label and evaluate dialogue training data. Typically, crowd workers label responses given dialogue contexts and rate them using, in the case of LaMDA, the following metrics: sensibleness [24], specificity [24], interestingness, safety, groundedness, informativeness, citation accuracy, helpfulness, and role consistency (see Section 2.2).

A **task-oriented dialogue system** is characterised by its clearly defined and measurable goal (usually to help a user achieve their goal efficiently), structured behaviour, a specific domain, and efficiency [26]. Typically, the dialogue system initiates the dialogue with a user. Applications include technical support [27] and recommendation systems [28]. Traditionally, task-oriented dialogue systems are designed as a pipeline consisting of a dialogue manager, a natural language understanding unit (NLU), a dialogue state tracker (DST), and a natural language generation unit (NLG). A problem with this traditional pipeline is that each unit is trained and supervised independently, making the pipeline vulnerable to error propagation across the units [29]. Consequently, recent efforts often integrate units. SimpleTOD (Simple Task-Oriented Dialogue) [30] is a simple integrated approach with state-of-the-art performance. SimpleTOD uses GPT-2 [2] to generate responses for task-oriented dialogue. SimpleTOD solves the sub-tasks of the different units in a unified way through multi-task maximum likelihood training. It enables modelling of the inherent dependencies between the sub-tasks of task-oriented dialogue by optimizing for all tasks in an end-to-end manner.

A **conversational dialogue system** seeks to keep an engaging conversation with the user [26]. The dialogues are usually unstructured and open-domain, with context and variability in utterances being essential features. An interesting and engaging dialogue is maintained if there is satisfying variation in topic and language. However, to keep the user's attention, the context should not fluctuate too much. The two main approaches to building dialogue systems are rule-based and data-driven. A data-driven dialogue system typically uses either utterance classification or utterance

generation.[2] Example conversational dialogue agent designs include chatbots with personality [31] and agents mimicking movie characters [32]. In the Second Conversational Intelligence Challenge, the conversational dataset PERSONA-CHAT was introduced. TransferTransfo [33] uses this dialogue-only data to fine-tune a dialogue system. TransferTransfo proved to be a state-of-the-art conversational dialogue system, winning the automatic metrics track.

There are several **hybrid and general dialogue systems**. Sun et al. [23] propose a task-oriented dialogue system enhanced with chit-chat. Their system consists of two language models and a switch module that decides their interactions depending on the context. Hybridization, in the form of integration of conversational dialogue elements, led to a more natural and engaging dialogue. LaMDA (Language Models for Dialog Applications) is a pre-trained deep learning language model [3]; it is closely related to the LaMDA metrics discussed in the paragraph about the evaluation of dialogue systems. LaMDA is computationally costly but achieves excellent results with the ability to generate and rank its generated responses. When fine-tuned on specific metrics, LaMDA can achieve near-human performance on sensibleness, specificity, and interestingness. While fine-tuning on a small set of safety and groundedness labeled data showed increased performance, LaMDA's gap to human performance is still significant.

Among **computer-assisted language learning (CALL)** systems, the research most closely related to our work includes research on exercise generation [15, 16, 17, 18], specifically exercise generation for foreign language learning [15, 18]. Unlike research on AI-based generation of question-answer assessments [17, 34], we seek to generate dialogues that include questions and answers, but not only those two types of utterances. Thus, our research is also related to CALL research on generating exercises from texts [18] with an emphasis on machine learning [35] and conversational agents for (gamified) language practice [36, 37, 38].

4. Proposed Architecture: The BFD Generator

The main point of our novel BFD Generator is to take a dialogue snippet and extend it to a more extensive, yet meaningful, dialogue using forward and backward utterance generation. Figure 2 depicts the architecture of the BFD Generator and its components, thus detailing Figure 1. The BFD Generator's components, discussed in this section, include a pre-trained causal language model that can switch between using two adapters, a decoding method, and a selection module.

We study both the BFD Generator in general as well as two special cases, the BD Generator and the FD Generator. The BFD Generator can generate, with respect to an input snippet, both preceding and following utterances. The BD Generator only generates preceding utterances, creating BD-extended dialogues. The FD Generator only generates following utterances, creating FD-extended dialogues.

Input and Output. The BFD Generator pipeline starts with the input to the BFD Generator, consisting of a dialogue snippet c, topic t, and length l. The snippet c consists of the utterances u_i for $i = 1, ..., n$, which alternate between the two speakers s_1 and s_2. The snippet and topic pair $d = (c, t)$ is transformed into a sequence of tokens using a pre-trained tokenizer and sent into the

[2] Some dialogue systems do not generate utterances but consider it a classification problem and pick from a selection of human-written utterances. Other systems generate utterances by looking at the context (either the dialogue or elements of the dialogue). This dichotomy applies to both conversational and task-oriented systems.

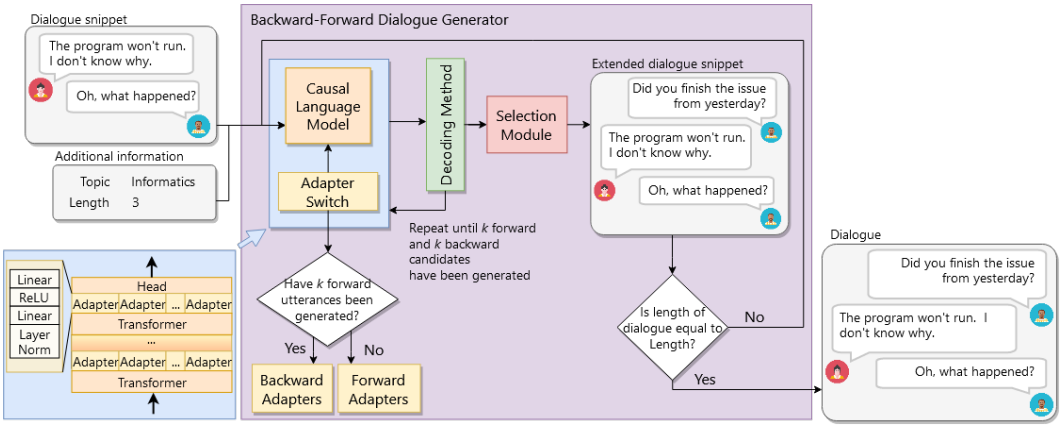


Figure 2: Pipeline of the BFD Generator, detailing the "Proposed Architecture" in Figure 1. Inputs are sent as a sequence of tokens into the Causal Language Model to generate $2k$ candidate utterances. The Adapter Switch applies the appropriate adapters to the language model, making it a forward or backward expert. The layers of the adapters and how they are applied to the Causal Language Model is shown to the bottom left. The candidates are scored in the Selection Module, and the highest-scoring candidate is added to the dialogue snippet. The cycle continues until the dialogue length is equal to the input length. Finally, the extended dialogue is output.

causal language model. The output from the causal language model is a probability $P(x|t, c')$ for the next word to be x given a sequence consisting of t and the current extended dialog snippet c', initialized as $c' = c$. Here, V is the vocabulary, and we consider all words $x \in V$.

Forward and Backward Adapters. Since GPT-2 is a decoder-only architecture, all inputs for adapting the language model are expressed as sequences of tokens. We adapt the TransferTransfo approach [33] to our topic-specific dialogue domain by swapping personality (represented as 4-6 sentences describing the identity of the speaker) with topic t in the input. This is to keep the topic consistent throughout the generated dialogue for the generation of more relevant phrases for good language learning. Each dialogue in the dataset is split into multiple dialogue snippets, which are part of the input sequences with the following structure: a beginning-of-sequence token, a topic (noun or noun phrase), a dialogue snippet with utterances separated with two different speaker tokens, and finally an end-of-sequence token. A critical difference between the backward and forward experts[3] is the utterances' order in the input sequence. For the forward expert, the utterances are ordered from the oldest to the most recent utterance. For the backward expert, the opposite ordering is used. The final input token sequence combines the word, positional, and segment embeddings of the input sequence, which adds the meaning of the words and strengthens their positional information and which speaker each utterance belongs to [33].

Our backward and forward experts are trained separately to optimize their individual performance and ensure independence. The adapters are optimized over a combination of two loss functions: a next-utterance classification loss and a language modeling loss, where distractors are randomly sampled utterances [33]. We hypothesize, and validate experimentally in Section 5.3, that adapter-based tuning requires fewer resources to train to an adequate level and is more space

[3]Backward adapters are placed between the transformer layers in the language model to construct a backward expert. When the backward adapters replaced by with forward adapters, we get a forward expert.

efficient than fine-tuning. Adapter-based tuning also allows for swapping of active experts in the large-scale language model for forward and backward utterance generation.

Adapter Switch. Since the experts are trained separately, we ensure that the appropriate set of adapters is used for candidate utterance generation. For the BFD Generator, after forward utterances have been generated, the Adapter Switch swaps to the backward adapters. For the BD and FD Generators, only backward or forward adapters are used, respectively.

Causal Language Model. We adapt the large language model GPT-2 [2] to the topic-specific conversational dialogue domain by adapter-based tuning, taking inspiration from the Transfer-Transfo technique [33]. Specifically, we develop an expert for the forward-generation task and an expert for the backward-generation task. Both experts write text in a forward manner, word by word, using the causal language model's ability to predict the probability of the next word given a word sequence. The experts utilize Residual Adapters [20], which are trainable, simple, and small neural networks placed between each decoder layer within the large-scale language model.

Decoding Method. Dialogue generation is an open-ended task, where each context can have many reasonable preceding and following utterances. It is difficult to judge if an utterance is suitable for a context while constructing the utterance. Thus, we generate a pool of candidate utterances that are compared to the context and against each other.

Language models only output the probabilities for the next word given a context, while decoding methods are used to construct sentences given these probabilities. Top-p sampling is a decoding method shown to be proficient at generating natural, engaging, and interesting text while reducing the chance of trailing off the topic [39, 40]. Thus, to keep the candidates diverse while staying on-topic, we use top-p sampling with temperature as the decoding method. We construct $2k$ candidate utterances in natural language using top-p sampling. The candidates are denoted u_{i-1}^m and u_{n+1}^m, for $m = 1, ..., k$, for backward and forward utterances respectively.

Additionally, we ensure that each candidate (for each direction) are unique. Half of the candidates are generated with one of the speaker tokens first, and the rest with the other, as the model may rely on apprioriate speaker assignment.

Selection Module. The Selection Module scores the candidate utterances and adds the highest-scoring candidate to the dialogue. Due to the independence of the experts, only a single candidate (not one from each expert) is added to the current snippet c in each iteration. This is to avoid discrepancies in the dialogue since the context may change when a candidate is added.

Candidates are scored using a weigthed sum consisting of proper noun count and utterance disimilarity. Let the snippet be $c = U^{\text{self}} \cup U^{\text{other}}$. Here, U^{self} is all of the utterances by the candidate's speaker and U^{other} the utterances by the other speaker, ordered from the oldest to most recent utterance. The self utterance disimilarity d_{self} between the candidate u_c and U^{self} is:

$$d_{\text{self}}(u_c, U^{\text{self}}) = \sum_{i=0}^{|U^{\text{self}}|} (g_{\text{self}} - sim(u_c, u_i) * k_4^i), \tag{1}$$

where $g_{\text{self}}, k_4 \in (0, 1)$ and $sim(u_c, u_i) = u_c \cdot u_i / (|u_c||u_i|)$ is the cosine similarity between two utterance embeddings, a measurement for the relatedness of the ground truth and the generated response in conversational dialogue systems [41]. Here, g_{self} denotes the ideal cosine similarity score between u_c and U^{self}. The other utterance disimilarity d_{other} swaps U^{self}, g_{self}, and k_4 with

U^{other}, g_{other}, and k_5, respectively. Parameters k_4 and k_5 reduce the influence of utterances that are further away from u_c. Thus, the score for a candidate u_c generated from the snippet c is:

$$s(u_c, c) = k_1 * n_c - k_2 * d_{\text{self}} - k_3 * d_{\text{other}}, \tag{2}$$

where k_j for $j = 1, 2, 3$ are positive coefficients and n_c is the proper noun count of the candidate u_c. The candidate score $s(u_c, c)$ is a weighted sum of proper noun count n_c (measuring specificity and being on topic) and two scores for how dissimilar the candidate is to other utterances by the same speaker d_{self} and the other speaker d_{other}.

5. Experimental Results

This section aims to validate the benefits of the BFD Generator as a creative tool for creating dialogues for language learning through various experiments.

Setup. We use the small GPT-2 architecture provided by the `transformers` [42] library. Residual Adapters are provided by the `adapter-transformers` library [43]. The seed is set to 24. Top-p sampling is used with p = 0.9 with temperature t = 0.7. The parameters of the Selection Module $(g_{\text{self}}, g_{\text{other}}, k_1, k_2, k_3, k_4, k_5)$ are respectively set to $(0.42, 0.43, 0.1, 5, 4, 0.01, 0.1)$ based on pilot experiments with human evaluations by two educators. Following [33], we train over two epochs, and the learning rate of the Adam optimizer is set to $6.25 * 10^{-5}$. The batch size is set to 2 due to the limited resources. The 4 closest utterances of a candidate are used during training and utterance generation. The Universal Sentence Encoder is provided by `spaCy`[4] in the `en_core_web_sm` package, which is also used for noun phrase recognition.

Dataset. The open-source conversational dialogue corpus WIZARD OF WIKIPEDIA [21] is used for adapting the language model. Its dialogues are labeled by topic and have informative dialogues grounded by crowd workers using Wikipedia, which we hope the BFD Generator can mimic. We use a training, validation, and test data split from the `ParlAI` framework [44]. The training set contains 129.6k dialogues with a total of 669.4k utterances. Redundant whitespaces, dialogues containing file extensions (e.g., ".png"), square brackets, or utterances of more than 200 tokens are removed. Only fifteen dialogues are removed in this way. Using a rule-based algorithm we find that the dataset mainly contains dialogues on CEFR level B2 or higher, which may be reflected in the generated utterances.

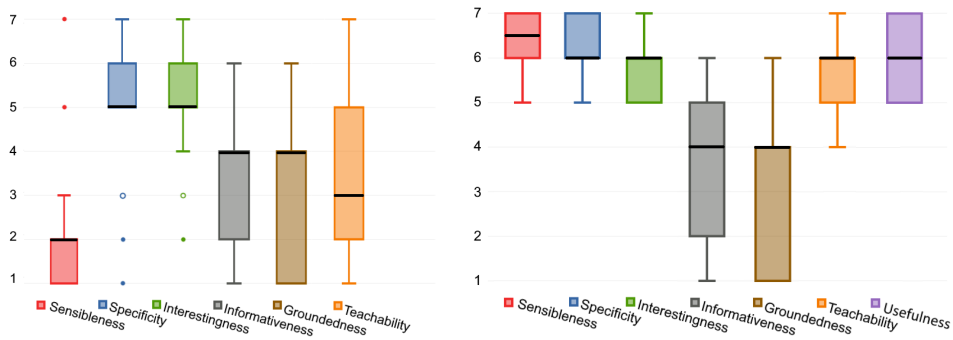
5.1. User Study

Since automatic metrics for dialogue systems have shown a low correlation to human judgments [45, 46], we rely on expert evaluations of 30 dialogues. In our small user study, two highly interested educators were each presented with 15 different dialogues generated from 5 snippets.[5]

Quantitative Study. The educators score the dialogues with our metrics (see Section 2.2) using a 7-point Likert scale. Afterward, they are given five minutes to adjust a dialogue to be suitable as an exercise at a specified CEFR level. Finally, they evaluate the adjusted dialogue on

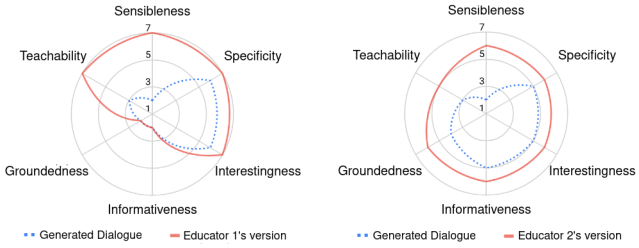
[4] https://spacy.io/

[5] Dialogues were created using the BFD, BD, and FD Generators. There were 15 dialogues in total (5 dialogues for each model), covering 5 different topics and 4 different CEFR levels (A1 to B2) [22, Appendix A.4].



(a) Generated dialogues (b) Dialogues after human adjustments

Figure 3: Boxplot of human evaluations of generated dialogues before (3a) and after (3b) educators' dialogue adjustments. The BFD-generated dialogues and their edited versions are scored on the six metrics (horizontal axis) using a 7-point Likert scale, with 4 being neutral and 7 being best. Feedback on the usefulness (purple) of the BFD-generated dialogue for creating language learning content on the specific topic and CEFR level is given on the same scale.



(a) Educator 1 (b) Educator 2

Figure 4: The educators' individual evaluations, in radar plot format, of the dialogues created from the snippet shown in Table 1. The evaluations differ most notably on the groundedness and informativeness metrics (southwest and south in the radar plots). However, they are similar in teachability, sensibleness, specificity, and interestingness (the other directions), where both educators increased the quality in less than five minutes.

the same metrics and how useful the BFD-generated dialogue was in creating language learning content. The user study results are shown in Figure 3. From the results, we see that the educators consistently increased the dialogues' sensibleness, interestingness, and teachability. Notably, educators can consistently add significant value to sensibleness and teachability. However, we observe a big interquartile range in informativeness and groundedness, both in the generated and human-adjusted dialogues. We hypothesize that (i) these two metrics are less important for our purposes or (ii) the time limit was too tight. Throughout the study, the generated dialogues (with only forward, only backward, or both utterance types) were helpful to the educators in making language learning content, see the usefulness boxplot in Figure 3b (in purple, to the right).

Qualitative Study. We study how the educators used the BFD Generator for an example from the user study. Table 1 shows a BFD-generated dialogue and an educator's version of it after being given five minutes to adjust it. The other educator created a similar adjusted dialogue,

BFD Generator Dialogue	Dialogue after Educator 1's Adjustments
A: I have! What is your favorite travel destination? B: I love traveling. I love to travel, do you? A: Yes, I've been to Greece. Have you been to Greece? B: **What are your plans for the summer holiday?** A: **We are going to Greece. I can't wait!** B: **That's great, I love Greece! Have you been there before?** A: **No, it's my first time. Do you have any recommendations?** B: I don't really know much about travel, but I know that I love to travel. A: I have heard that travel is one of the most important activities for the human race. Do you know if that is true?	A: I love to travel, do you? B: Yes, I love traveling. I'm going to Italy this summer. A: That sounds nice! B: **What are your plans for the summer holiday?** A: **We are going to Greece. I can't wait!** B: **That's great, I love Greece! Have you been there before?** A: **No, it's my first time. Do you have any recommendations?** B: I haven't been in Greece, so I don't know. Have you been in Italy and can give me some recommendations? A: Yes, I have been in Italy. You should visit the Colosseum in Rome!

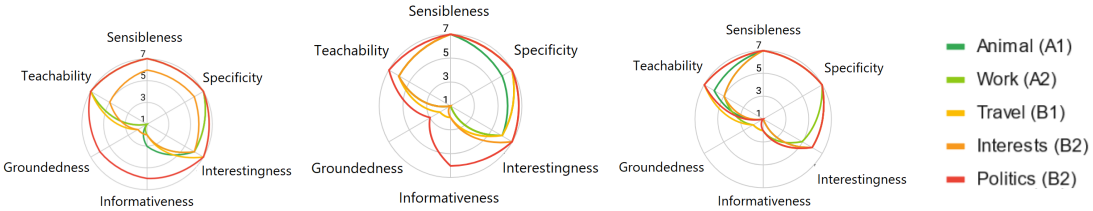
Table 1
A dialogue (left) created by the BFD Generator using a dialogue snippet (bold) and the topic "Travel". The educators are given five minutes to adjust the generated dialogue to make it suitable for CEFR level B1 given the topic. An example dialogue after an educator's edits (right) must contain the dialogue snippet, but everything before and after it can be changed.

which is left out for space reasons. The educators' evaluations and the dialogues' improvements across the metrics are seen in Figure 4. Both of the adjusted dialogues use parts of the generated utterances, notably more in the backward utterances to the dialogue snippet. They found a phrase that suited the CEFR level (i.e., "I love traveling") and used repetition to make it more suitable for a language exercise. In the forward utterances of the dialogue snippet, there is less resemblance between the generated dialogue and the educators' versions. It is clear that the generated forward utterances lack sensibleness and do not fit as well with the dialogue snippet. Thus, the educators themselves increase the quality of the dialogue. Interestingly, Educator 1 uses the uncertainty in the generated utterance "I don't really know much about travel, but I know that I love to travel" by changing the reasoning behind the uncertainty to make it fit the context better.[6] This example suggests that educators can productively improve the BFD-generated output, even though the adjusted dialogue may have lower-than-ideal quality for some metrics.

5.2. Generating Surrounding Dialogues

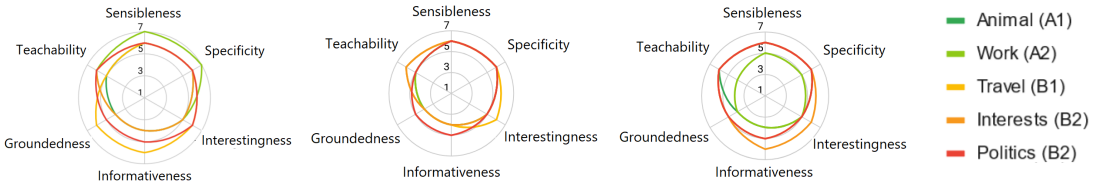
We now compare the self-evaluations of the adjusted BFD-extended dialogues with adjusted BD- and FD-extended dialogues. In Figure 5 and Figure 6, self-evaluations of all adjusted BFD-, BD-, and FD-extended dialogues by the educators are shown. The first observation we make is the following. While there is some variation among the educators, the overall shapes of the radar plots are similar between adjusted BFD-, BD-, and FD-extended dialogues when looking at the evaluations from the educators separately. We generally observe that adjusted BFD-extended dialogues for both educators score the highest across all metrics. This suggests that while both preceding utterances (adjusted BD-extended dialogues) and following utterances (adjusted FD-extended dialogues) are helpful separately, it is when joined in the BFD setting that

[6]Another interesting point is how the adjusted dialogues are similar, starting with expressing interest for traveling and ending with recommendations for attractions. Additionally, the evaluations of the groundedness and informativeness of the dialogues are significantly different between the educators. However, the educators' scores for the sensibleness, specificity, interestingness, and teachability of the generated dialogue align well. This shows the subjectivity of human judgments.



(a) BFD-extended (b) BD-extended (c) FD-extended (d) Labels

Figure 5: Evaluations of all adjusted BFD-, BD-, and FD-extended dialogues by Educator 1.



(a) BFD-extended (b) BD-extended (c) FD-extended (d) Labels

Figure 6: Evaluations of all adjusted BFD-, BD-, and FD-extended dialogues by Educator 2.

the best resulting extended dialogues may be achieved.

Additionally, the educators told us that all of the extended dialogues were on-topic. This supports our hypothesis about being able to keep the topic of the dialogue consistent by adapting a transfer learning technique for keeping the personality of the speakers consistent. The educators also found the extended dialogues to be on a too-high CEFR level in most cases. This may be due to the CEFR imbalance in the dataset. In the user study, the educators adjusted BFD-extended dialogues first, then BD-generated, and finally FD-generated. Thus, the results may be affected by how familiar the tasks in the user study become to the educators.

In summary, Figure 5 and Figure 6 suggest that: (i) generating in both directions generally led to higher scores on the targets (compared to 1-way generation), for both educators; (ii) measuring the qualities of texts can vary with different experts (especially when it comes to groundedness and informativeness) and is therefore still an interesting research area to explore; and (iii) generating in different directions led to texts with similar qualities.

5.3. Adapter-Based Tuning versus Traditional Fine-Tuning

We now study the hypothesized better scaling of adapter-based tuning, see the discussion in Section 4, compared to traditional fine-tuning of language models. We fine-tune GPT-2 for forward utterance generation using the technique described in Section 4.[7] On a single GeForce 980 Ti GPU with 6 VRAM, it took about 4.9 hours to train the backward adapters and 4.8 hours to train the forward adapters. Fine-tuning GPT-2 requires more VRAM than is available. Thus, we reduce the batch size to 1 and fine-tune it over forward utterance generation over the same loss and same input representation, hyperparameters, and dataset. Fine-tuning GPT-2 for forward utterance generation took 15 hours. This is a 306% increase in training time from the individual

[7]While GPT-2 [12] has, along certain dimensions, been succeeded by GPT-3 [1] and GPT-4 [13], they are lacking in some respects. It is not currently possible to perform the studies presented here with these later models.

adapters. By running some qualitative studies on input-output pairs, we observe that both models can generate utterances that are on-topic and possible to be interpreted by humans, as discussed in Section 5.1. Additionally, the storage space required for the fine-tuned models (500MB each) was bigger than the adapters (151MB each) combined with the small base GPT-2 model (500MB) that we used in the BFD Generator for the user studies. Since base language models like GPT-2 are available online, it is possible to retrieve the base models from open sources like the `transformers` [42] library. In the case of adding more experts to the architecture, it is cheaper both in training time, VRAM requirements, and storage space to use adapter-based tuning instead of fine-tuning. This result supports the hypothesized benefit of using adapters (see Section 4).

6. Conclusion and Future Work

Conclusion. To solve the SDG problem (SDGP), we suggest the BFD Generator (Backward-Forward Dialogue Generator) as a creative data-driven tool to help language educators. This architecture solves the SDGP by dividing it into two tasks: backward and forward utterance generation. By iteratively adding backward and forward utterances to a dialogue snippet, the resulting dialogue is an extended version of the snippet. Finding the best position of the dialogue snippet in the resulting dialogue is solved implicitly and flexibly. The BFD Generator is data-driven and uses an adapter-based tuned large-scale language model, specifically GPT-2,[8] to generate the surrounding on-topic dialogue. Compared to the traditional tuning of large language models, the adapter-based approach requires fewer resources during training and had lower storage requirements for models. In a small user study, we consider how the BFD Generator can assist an educator in writing dialogues for language education. In our study, educators successfully use BFD-generated dialogues as an aid to create language learning exercises, which is only one of its many applications.

Future Work. Most of the BFD-generated utterances are on CEFR level 2 or higher in our experiments, and the educators need to reformulate to other CEFR levels if required. Developing a system that creates dialogues for a specified CEFR level is of interest for future work. Second, while our user study demonstrates the benefits of the BFD Generator, a more extensive user study on additional topics and CEFR levels with more educators would improve the understanding of its capabilities. Third, while the BFD Generator can generate dialogues with high specificity, they could be more grounded and informative. A separate knowledge-retrieval system to enhance the groundedness and informativeness of its generated utterances [3] can improve the correctness of domain-specific knowledge in dialogue. Finally, there is the issue of evaluation by students. Bodnar identifies three areas of research on automatic exercise generation [47]: evaluation of exercise generation technology, human expert judgments of exercise quality, and analysis of students' usage of generated exercises. While we have made progress on the first two points in this paper, we have yet to address the third.

[8] There has been rapid development in the GPT family of large-scale language models. At the time of this writing, GPT-4 [13] was recently released. Unfortunately, GPT-4 is a proprietary model that does not enable the type of research on adapters being studied here. However, we hypothesize that the general architecture and research direction being pursued here would be valuable for GPT-4, should it be released in the future.

References

[1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. J. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, ArXiv abs/2005.14165 (2020).

[2] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, OpenAI blog 1 (2019) 9.

[3] R. Thoppilan, D. D. Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H. Cheng, A. Jin, T. Bos, L. Baker, Y. Du, Y. Li, H. Lee, H. S. Zheng, A. Ghafouri, M. Menegali, Y. Huang, M. Krikun, D. Lepikhin, J. Qin, D. Chen, Y. Xu, Z. Chen, A. Roberts, M. Bosma, Y. Zhou, C. Chang, I. Krivokon, W. Rusch, M. Pickett, K. S. Meier-Hellstern, M. R. Morris, T. Doshi, R. D. Santos, T. Duke, J. Soraker, B. Zevenbergen, V. Prabhakaran, M. Diaz, B. Hutchinson, K. Olson, A. Molina, E. Hoffman-John, J. Lee, L. Aroyo, R. Rajakumar, A. Butryna, M. Lamm, V. Kuzmina, J. Fenton, A. Cohen, R. Bernstein, R. Kurzweil, B. Aguera-Arcas, C. Cui, M. Croak, E. Chi, Q. Le, Lamda: Language models for dialog applications, CoRR abs/2201.08239 (2022). URL: https://arxiv.org/abs/2201.08239. arXiv:2201.08239.

[4] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, CoRR abs/1810.04805 (2018). URL: http://arxiv.org/abs/1810.04805. arXiv:1810.04805.

[5] I. Sutskever, O. Vinyals, Q. V. Le, Sequence to sequence learning with neural networks, in: Advances in Neural Information Processing Systems, volume 27, 2014. URL: https://proceedings.neurips.cc/paper_files/paper/2014/file/a14ac55a4f27472c5d894ec1c3c743d2-Paper.pdf.

[6] A. Adhikari, A. Ram, R. Tang, J. Lin, Docbert: BERT for document classification, CoRR abs/1904.08398 (2019). URL: http://arxiv.org/abs/1904.08398. arXiv:1904.08398.

[7] A. Xu, Z. Liu, Y. Guo, V. Sinha, R. Akkiraju, A new chatbot for customer service on social media, in: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, CHI '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 3506–3510. URL: https://doi.org/10.1145/3025453.3025496. doi:10.1145/3025453.3025496.

[8] M. Hoang, O. A. Bihorac, J. Rouces, Aspect-based sentiment analysis using BERT, in: Proceedings of the 22nd Nordic Conference on Computational Linguistics, Linköping University Electronic Press, Turku, Finland, 2019, pp. 187–196. URL: https://aclanthology.org/W19-6120.

[9] A. R. Abas, I. El-Henawy, H. Mohamed, A. Abdellatif, Deep learning model for fine-grained aspect-based opinion mining, IEEE Access 8 (2020) 128845–128855. doi:10.1109/ACCESS.2020.3008824.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, CoRR abs/1706.03762 (2017). URL: http://arxiv.org/abs/1706.03762. arXiv:1706.03762.

[11] O. Agbodike, C.-H. Huang, J. Chen, Cognitive attention network (can) for text and image multimodal visual dialog systems, in: 2020 6th International Conference on Applied System

Innovation (ICASI), 2020, pp. 37–41. doi:10.1109/ICASI49664.2020.9426334.
[12] A. Radford, K. Narasimhan, Improving language understanding by generative pre-training, 2018.
[13] OpenAI, GPT-4 technical report, 2023. arXiv:2303.08774.
[14] R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, S. Fidler, Skip-thought vectors, in: C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 28, Curran Associates, Inc., 2015. URL: https://proceedings.neurips.cc/paper/2015/file/f442d33fa06832082290ad8544a8da27-Paper.pdf.
[15] L. Perez-Beltrachini, C. Gardent, G. Kruszewski, Generating grammar exercises, in: Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, 2012, pp. 147–156.
[16] J. J. Almeida, E. Grande, G. Smirnov, Exercise generation on language specification, in: Recent Advances in Information Systems and Technologies: Volume 1 5, Springer, 2017, pp. 277–286.
[17] G. Kurdi, J. Leo, B. Parsia, U. Sattler, S. Al-Emari, A systematic review of automatic question generation for educational purposes, International Journal of Artificial Intelligence in Education 30 (2020) 121–204.
[18] T. Heck, D. Meurers, Generating and authoring high-variability exercises from authentic texts, in: Proceedings of the 11th Workshop on NLP for Computer Assisted Language Learning, 2022, pp. 61–71.
[19] W. H. O. R. O. for Europe, Ukraine crisis. Public health situation analysis: refugee-hosting countries, 17 March 2022, 2022. URL: https://apps.who.int/iris/handle/10665/352494.
[20] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan, S. Gelly, Parameter-efficient transfer learning for NLP, in: K. Chaudhuri, R. Salakhutdinov (Eds.), Proceedings of the 36th International Conference on Machine Learning, volume 97 of *Proceedings of Machine Learning Research*, PMLR, 2019, pp. 2790–2799. URL: https://proceedings.mlr.press/v97/houlsby19a.html.
[21] E. Dinan, S. Roller, K. Shuster, A. Fan, M. Auli, J. Weston, Wizard of Wikipedia: Knowledge-powered conversational agents, in: Proceedings of the International Conference on Learning Representations (ICLR), 2019.
[22] E. Zhang Chang, Surrounding Dialogue Generation using Deep Learning with Adapters, Master's thesis, Norwegian University of Science and Technology (NTNU), 2022. URL: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/3034872.
[23] K. Sun, S. Moon, P. A. Crook, S. Roller, B. Silvert, B. Liu, Z. Wang, H. Liu, E. Cho, C. Cardie, Adding chit-chat to enhance task-oriented dialogues, ArXiv abs/2010.12757 (2021).
[24] D. Adiwardana, M. Luong, D. R. So, J. Hall, N. Fiedel, R. Thoppilan, Z. Yang, A. Kulshreshtha, G. Nemade, Y. Lu, Q. V. Le, Towards a human-like open-domain chatbot, CoRR abs/2001.09977 (2020). URL: https://arxiv.org/abs/2001.09977. arXiv:2001.09977.
[25] C. of Europe, The CEFR levels, https://www.coe.int/en/web/common-european-framework-reference-languages/level-descriptions, 2022. Accessed: 2022-05-18.
[26] J. Deriu, Á. Rodrigo, A. Otegi, G. Echegoyen, S. Rosset, E. Agirre, M. Cieliebak, Survey on evaluation methods for dialogue systems, CoRR abs/1905.04071 (2019). URL: http:

//arxiv.org/abs/1905.04071. arXiv:1905.04071.
[27] B. Liu, T. Yu, I. R. Lane, O. J. Mengshoel, Customized nonlinear bandits for online response selection in neural conversation models, in: AAAI, 2018.
[28] Z. Chen, H. Liu, H. Xu, S. Moon, H. Zhou, B. Liu, Nuanced: Natural utterance annotation for nuanced conversation with estimated distributions, arXiv preprint arXiv:2010.12758 (2020).
[29] B. Liu, G. Tür, D. Hakkani-Tür, P. Shah, L. Heck, Dialogue learning with human teaching and feedback in end-to-end trainable task-oriented dialogue systems, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 2060–2069. URL: https://aclanthology.org/N18-1187. doi:10.18653/v1/N18-1187.
[30] E. Hosseini-Asl, B. McCann, C.-S. Wu, S. Yavuz, R. Socher, A simple language model for task-oriented dialogue, in: Advances in Neural Information Processing Systems, volume 33, 2020, pp. 20179–20191.
[31] E. Dinan, V. Logacheva, V. Malykh, A. H. Miller, K. Shuster, J. Urbanek, D. Kiela, A. D. Szlam, I. Serban, R. Lowe, S. Prabhumoye, A. W. Black, A. I. Rudnicky, J. Williams, J. Pineau, M. S. Burtsev, J. Weston, The second conversational intelligence challenge (ConvAI2), ArXiv abs/1902.00098 (2019).
[32] W. Xu, F. Charles, C. Hargood, F. Tian, W. Tang, Influence of personality-based features for dialogue generation in computational narratives, in: ECAI, 2020.
[33] T. Wolf, V. Sanh, J. Chaumond, C. Delangue, Transfertransfo: A transfer learning approach for neural network based conversational agents, 2019. arXiv:1901.08149.
[34] R. Zhang, J. Guo, L. Chen, Y. Fan, X. Cheng, A review on question generation from natural language text, ACM Transactions on Information Systems (TOIS) 40 (2021) 1–43.
[35] B. Settles, G. T. LaFlair, M. Hagiwara, Machine learning–driven language assessment, Transactions of the Association for computational Linguistics 8 (2020) 247–263.
[36] J. Li, A. H. Miller, S. Chopra, M. Ranzato, J. Weston, Learning through dialogue interactions by asking questions, in: International Conference on Learning Representations, 2020.
[37] M. G. Araneta, G. Eryigit, A. König, J.-U. Lee, A. Luis, V. Lyding, L. Nicolas, C. Rodosthenous, F. Sangati, Substituto-a synchronous educational language game for simultaneous teaching and crowdsourcing, in: 9th Workshop on Natural Language Processing for Computer Assisted Language Learning (NLP4CALL 2020), 2020, pp. 1–9.
[38] E. da Cruz Dalcol, M. Poesio, Polygloss-a conversational agent for language practice, Natural Language Processing for Computer Assisted Language Learning (NLP4CALL 2020) (2020) 21.
[39] A. Holtzman, J. Buys, L. Du, M. Forbes, Y. Choi, The curious case of neural text degeneration, in: International Conference on Learning Representations, 2020. URL: https://openreview.net/forum?id=rygGQyrFvH.
[40] O. J. Mengshoel, Understanding the role of noise in stochastic local search: Analysis and experiments, Artificial Intelligence 172 (2008) 955–990. URL: https://www.sciencedirect.com/science/article/pii/S0004370208000040. doi:https://doi.org/10.1016/j.artint.2007.09.010.
[41] R. Lowe, M. Noseworthy, I. V. Serban, N. Angelard-Gontier, Y. Bengio, J. Pineau, Towards

an automatic turing test: Learning to evaluate dialogue responses, CoRR abs/1708.07149 (2017). URL: http://arxiv.org/abs/1708.07149. arXiv:1708.07149.

[42] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, A. M. Rush, Transformers: State-of-the-art natural language processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online, 2020, pp. 38–45. URL: https://www.aclweb.org/anthology/2020.emnlp-demos.6.

[43] J. Pfeiffer, A. Rücklé, C. Poth, A. Kamath, I. Vulić, S. Ruder, K. Cho, I. Gurevych, Adapterhub: A framework for adapting transformers, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 2020, pp. 46–54.

[44] A. H. Miller, W. Feng, A. Fisch, J. Lu, D. Batra, A. Bordes, D. Parikh, J. Weston, ParlAI: A dialog research software platform, arXiv preprint arXiv:1705.06476 (2017).

[45] C. Liu, R. Lowe, I. V. Serban, M. Noseworthy, L. Charlin, J. Pineau, How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation, CoRR abs/1603.08023 (2016). URL: http://arxiv.org/abs/1603.08023. arXiv:1603.08023.

[46] A. Ghandeharioun, J. H. Shen, N. Jaques, C. Ferguson, N. Jones, A. Lapedriza, R. Picard, Approximating interactive human evaluation with self-play for open-domain dialog systems, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 32, 2019. URL: https://proceedings.neurips.cc/paper/2019/file/fc9812127bf09c7bd29ad6723c683fb5-Paper.pdf.

[47] S. Bodnar, The instructional effectiveness of automatically generated exercises for learning French grammatical gender: preliminary results, in: Swedish Language Technology Conference and NLP4CALL, 2022, pp. 10–22.