mathematics

MDPI

Article

Personalizing Hybrid-Based Dialogue Agents

Yuri Matveev *, Olesia Makhnytkina, Pavel Posokhov, Anton Matveev and Stepan Skrylnikov

Information Technologies and Programming Faculty, ITMO University, 197101 Saint Petersburg, Russia
* Correspondence: yunmatveev@itmo.ru

Abstract: In this paper, we present a continuation of our work on the personification of dialogue agents. We expand upon the previously demonstrated models—the ranking and generative models—and propose new hybrid models. Because there is no single definitive way to build a hybrid model, we explore various architectures where the components adopt different roles, sequentially and in parallel. Applying the perplexity and BLEU performance metrics, we discover that the Retrieve and Refine and KG model—a modification of the Retrieve and Refine model where the ranking and generative components work in parallel and compete based on the proximity of the candidate found by the ranking model with a knowledge-grounded generation block—achieves the best performance, with values of 1.64 for perplexity and 0.231 for BLEU scores.

Keywords: personalized dialogue systems; hybrid models; retrieve models; refine models

MSC: 68T50

Citation: Matveev, Y.; Makhnytkina, O.; Posokhov, P.; Matveev, A.; Skrylnikov S. Personalizing Hybrid-Based Dialogue Agents. *Mathematics* **2022**, *10*, 4657. https://doi.org/10.3390/math10244657

Academic Editors: Heui Seok Lim, Sanghyuk Lee, Yeongwook Yang and Imatitikua Aiyanyo

Received: 10 November 2022
Accepted: 6 December 2022
Published: 8 December 2022

1. Introduction

The automatic generation of responses indistinguishable from human responses has been a long-term goal of artificial intelligence since the development of the Turing test [1]. While automatic recognition of emotion and sentiment in text is a well-researched problem [2,3], endowing a machine with a personality is currently viewed as critical for improving the quality of dialogue. Providing a chatbot with individual characteristics alleviates personality inconsistency and the lack of explicit long-term memory, which may result in different answers to the same question and a tendency for non-specific answers such as "I don't know". Personalizing a chatbot is a difficult but important task to ensure more realistic and natural communication.

Mairesse and Walker showed that language style can be an indicator of personality [4], and also that spoken language can be generated based on a particular personality [5].

There are approaches to specifying a person in implicit and explicit forms. When the persona is set implicitly, a large volume of user replies is required to reproduce the style of conversation of a particular speaker [6,7].

Another approach involves defining a person explicitly, that is, generating knowledge fragments attributed to a person. Exploring the inherent attributes of dialogues explicitly is one method of improving the diversity of dialogues and ensure their consistency. The topic and the personality are most widely studied among the various attributes. Qian et al. [8] define a personality as a set of profile keys and values and propose a model that consists of three key components: a profile detector that determines whether information about a person should be used, a bidirectional decoder that generates a response, and a position detector that predicts the correct word position at which the profile value can be replaced during decoder training. Such a model can be trained on general dialogue data (without specifying the identity of the interlocutor), and information about the person is included to generate responses that match the profile. Some of the most common attributes of a person are name, gender, age, weight, location, and zodiac sign.

Currently, the most common approach is to use both an explicit description of a person and dialogues with this person. For training of personalized dialogue agents, Zhang et al. [9] introduced an English-language corpus of dialogues, including descriptions of persons and conversations between them. Each of the conversations takes place between two participants who enact artificially modeled personas described by three to five sentences, such as "I like to ski", "I am an artist", "I eat sardines for breakfast daily", etc. At present, the personification of conversational agents is being actively researched and developed for English [6,9–18] and Chinese [19,20], for which there are several datasets available containing information about the person and dialogues. For example, in English, there exists PERSONA-CHAT [9] and ConvAI2 [21]. In Chinese, there are Pchatbot [22] and the Personality Assignment Dataset [8]. For the Russian language, there is Toloka Persona Chat Rus dataset [23], but there are virtually no studies with this dataset. Details regarding various available dataset are listed in Table 1.

Previously, in our work on dialogue agents [24], we noted the importance of personalization not just as a standalone problem but as a vital component of dialogue-based information systems in general. Recently, we proposed a baseline method for search and generative models for the personification of dialogue agents in Russian [25]. In some of our previous works [26], we explored hybrid models for text generation; in this paper, we propose a novel approach that extends both generative and search models in a hybrid manner.

In this work, we first review common designs of models for personalization of dialogue agents, the general ideas behind the algorithms, advantages and disadvantages, and training datasets. Then, we provide a detailed description of the dataset we employ for training our models. After that, we take a closer look at the existing models, their architectures, and specialty. Particularly, we focus on the Retrieve and Refine model, for which we demonstrate a modification with knowledge-grounded generation block, a hybrid approach that appears to demonstrate the best performance. Next, we discuss our approach to performance evaluation and note the details that are important to consider when analyzing the results. Finally, we provide the performance evaluation results and discuss ways in which the results may be interpreted for making decisions about choosing the right model for a task.

Table 1. Comparison of available datasets for building personalized dialogue agents.

Dataset	#Dialogues	Language	Source	Personalized Info
PERSONA-CHAT [9]	10,981	English	Crowdsourcing	Persona descriptions up to five sentences
ConvAI2 [21]	4529	English	Crowdsourcing	Persona descriptions up to five sentences
DuLeMon [19]	27,501	Chinese	Crowdsourcing	Persona descriptions up to five sentences
Personality Assignment Dataset [8]	9,697,651	Chinese	Weibo	Key-value pair profile
PchatbotW [22]	139,448,339	Chinese	Weibo	User ID & Timestamp
PchatbotL [22]	59,427,457	Chinese	Judicial Forums	User ID & Timestamp
Toloka Persona Chat Rus [23]	10,000	Russian	Crowdsourcing	Persona description fixed number of five sentences

2. Related Works

2.1. Dialogue Agents for English

Zhang et al. [9] were the first to propose employing neural networks with memory for building personalized dialogue systems. They investigated the main types of dialogue

models: ranking models with memory and generative models with memory. The implementations of the approaches are sufficiently distinctive, but the general principle of building the system is shared. A person's profile is constructed from collections of facts about this person in text form. The model extracts facts relevant to the current query from the profile and, by combining these facts with the query, calculates a prediction. In a manual performance evaluation, the ranking models are observed to produce higher-quality results. Presumably, ranking models have the capacity to select more specific and interesting responses; however, being limited by the set of candidates from which to select, they may produce erroneous responses that may not fit the current context. On the other hand, generative models are more versatile but tend to predict short, general sentences that are neither informative nor interesting. Exploring approaches to understanding and solving these problems is an important task in the field [18].

Currently, almost all research on the personification of conversational agents utilizes the Persona-Chat dataset and similar ones, and improvements rely on the modification of neural network architectures for building models for generating questions and answers. Existing research considers different approaches for predicting the next utterance: ranking models and generative models. Ranking models produce the next utterance by evaluating utterances in the training set as candidate responses. Generative models obtain new sentences by assessing a dialogue history (and, optionally, a persona) and then generating a response word by word. For generative models, the most common approaches rely on recurrent neural networks and vector representations of words based on the GloVe language model [9].

Kulikov et al. [27] investigated an attention-based neural autoregressive sequence model where the encoder is a bidirectional LSTM, the decoder is an LSTM, and GloVe is used for embeddings. The authors reviewed three search strategies, greedy search, beam search, and iterative beam search, each maximizing the probability of selecting the most suitable text. The best results were achieved with ray search and iterative ray search. Yavuz et al. [28] presented the results of a series of experiments with response generation models, where, in addition to the dialogue context, it is assumed that the corresponding unstructured external knowledge is also available in text for the models. Several works have investigated transformer models, specifically neural networks of transformers and transfer learning, where a model is initially trained on a large dataset and further trained on Persona-Chat [29]. Recently, pre-trained models based on autoregressive language model transformers (GPT-2) have become widely popular, with various decoding strategies for generating the response text, such as greedy search, beam search, top-k sampling, etc. [30]. In their work, Yavuz et al. [28] considered the Oracle Seq2Seq + Copy model with the copy mechanism.

Madotto et al. [31] presented a meta-learning setting for personalizing dialogue agents without conditioning the model response to the persona description. The model learns to adapt to new personas by leveraging dialogue samples collected from the same user, unlike in the common approach based on conditioning the response to the persona descriptions. This research shows that a dialogue agent trained with meta-learning achieves more consistency in a dialogue by both automatic measures and human evaluation.

To improve the extraction of personal information, Mazaré et al. [32] proposed to extract it from all user messages according to special linguistic rules to form a persona. After that, this information is used as additional information for concatenation with the vector representation of the context passing through the memory block. The joint representation is used to personalize the ranking search in the database of candidates to improve the quality of the answers of the ranking dialogue model.

Cao et al. [33] suggested several data manipulation techniques that improve the generation performance, including token- and phrase-level persona editing and distillation and augmentation of both persona and dialogue history. It is a model-agnostic data manipulation approach which distills the original data and then augments both the volume and the diversity of the distilled data. Curriculum learning is performed afterwards

utilizing both the augmented and the original data. Results show that this method is an effective improvement of transformer encoder–decoder and GPT2 in terms of performance.

Song et al. [34] proposed a novel BERT-based dialogue model BERT Over BERT: BoB. The model consists of three BERT-based sub-modules: an encoder, a response generation decoder, and a consistency understanding decoder. The encoder works similar to a standard BERT model for context encoding. The response decoder works in an auto-regressive decoder manner. The consistency understanding decoder initializes a good semantic representation for understanding tasks from BERT. It appears that this architecture has a better understanding of persona consistency despite being trained on limited personalized dialogue data.

A transformer model proposed by Zheng et al. [35] can generate responses using conversational data with sparse personality. Additionally, the authors introduced special personal attribute embeddings to improve the modeling of the context of a dialogue by encoding persona profiles along with the history of the dialogue. At the decoding stage, to include the target persona and balance its contribution, an attention routing structure is used inside the decoder to combine features extracted from the target persona and dialogue contexts using predictive weights.

Gu et al. [36] thoroughly explored the impact of utilizing personas that describe both speakers in a dialogue for response selection in retrieval-based chatbots by combining BERT retrieval models in different configurations. Each configuration illustrates a certain method of interaction between personas and contexts or responses. The authors show that the best model for this task is the cross-encoder [37] persona modification.

Kasahara et al. [38] proposed an effective tuning method for dialogue agents based on large-scale pre-trained models using dialogue data based on a single persona. This prompt-tuning method works by freezing a language model and its embedding layer and tuning the newly added embedding layer for persona information. The authors demonstrated that dialogue systems constructed using this prompt-tuning method can respond more naturally in an alignment with a persona and with less computational resources required than fine-tuning.

For researching the correlation between persona and empathy, Das et al. [39] investigated several fusion strategies to model the inter-dependencies of the persona, emotion, and entailment information. They proposed a retrieval bi-encoder model modification with an interaction layer and MLP cross-entropy loss. This model outperforms the retrieval BERT-CRA model and generative TransferTransfo. The authors explained that improvements in the performance is an outcome of usage of emotion, entailment, and concepts, as they play an important role in the problem of response selection. These features help improve the performances of models and provide critical insights into certain aspects of how humans communicate with each other.

2.2. Dialogue Agents for Other Languages

Personalized dialogue agents are actively developing for the Chinese language. PLATO-2 is a model by Bao et al. [40] with three variations: 1.6 billion, 314 million, and 93 million parameters. The model was trained in two stages. First, the model was trained only for one-to-one matching, i.e., only one response is generated for each context. For the second stage, a hidden variable is introduced, which has categorical values each corresponding to a specific latent speech act in the response. The model estimates the distribution of latent acts in the training sample and then generates a response with the chosen latent variable. Both these tasks are performed together in the same model. With that, the model can generate various answers, but it is necessary to choose the most relevant one by ranking this set. Replies are evaluated according to the degree of consistency between them and the provided context of the dialogue.

PLATO-LTM by Xu et al. [19] is a model composed of three modules. The first module extracts information about a person: it receives a phrase or a dialogue fragment and labels it depending on whether it contains any information about the person. This module is

implemented by ERNIE-CNN, which is built from the ERNIE [41] model, for processing text and a CNN for classification [42]. The second module is long-term memory, it stores all facts about the chatbot and the person it talks to; it uses the ERNIE model trained on the DuLeMon [19] dataset. The third module is based on the PLATO-2 model and it generates the response based on the person and the context.

Sugiyama et al. [43] considered the possibility of training a transformer model for conducting a dialogue in Japanese. For training, the authors used four models with 0.35, 0.7, 1.1, and 1.6 billion parameters, all pre-trained on comments from Reddit. For additional training in Japanese, they collected two datasets similar to the PersonaChat and EmpatheticDialogues datasets and their own specific dataset FavoriteThingsChat where the interlocutors shared their preferences with each other.

2.3. Our Previous Work for Russian

Previously, we reported our investigation of different dialog retrieval models architectures [44]. The best performance metrics were achieved by CoBERT, which is a ranking architecture based on BERT encodings with a co-attention mechanism between context, persona, and candidates. There, we proposed a specific fine-tuning algorithm suggesting synchronous learning of query and candidate encoders as a pre-training technique where both are then separately trained after. This algorithm provides better performance on small datasets.

A detailed overview of generative and search models was presented in one of our previous publications, where we describe how we developed a model that combines both approaches to avoid the disadvantages of both. The hybrid approach involves the joint use of generative and ranking models in various combinations [25]. In this paper, we compare various retrieval BERT-based models for dialogue agents implementation with Persona Chat baseline, which include generative and retrieval models for the Russian language. The results show that retrieval models achieve the best performance, both in automatic and expert testing and have simpler and more informative metrics. This paper concludes that the personification of bi-, poly-, and cross-encoders is possible by concatenation of the person and context vector.

3. Dataset

Toloka Persona Chat Rus is a dataset compiled in the Neural Networks and Deep Learning Laboratory at the Moscow Institute of Physics and Technology by modeling a certain person with their own personal characteristics during the dialogues by each of the participants in the study. The Toloka Persona Chat Rus dataset is packaged as two files: profile.tsv, containing lines with a person's characteristics separated by tabulations; and dialogues.tsv, containing more than 10,000 dialogues in Russian between the participants of the study and presented in a file in the format "persona1profile\$t$persona2profile\$t$dialogue", where "persona1profile" and "persona2profile" are the characteristics of the persons modeled in the conversation, and "dialogue" is the history of the dialogue, presented in HTML markup.

Each personality description is expressed in five sentences, for example "I draw" or "I have a snake". There are 1505 different profiles in total. Data collection took place in two stages. During the first stage, with the help of Toloka users, profiles were collected containing information about a person, their hobbies, profession, family, and life events. After that, those that were better suited for a dialogue were selected. At the second stage, the participants were asked to play a role of a person described by one of these profiles and communicate with each other. The purpose of each dialogue was to learn more about each other and talk about themselves. The produced dialogues were manually checked by other participants. Here is an example of one record in this dataset with added transliterations and translations for purposes of this paper:

Person A

Я медсестра
(Ja medsestra)
(I am a nurse)
Люблю ухаживать за больными
(Ljublju uhazhivat' za bol'nymi)
(I love taking care of patients)
Много читаю
(Mnogo chitaju)
(I read a lot)
Люблю путешествовать
(Ljublju puteshestvovat')
(I love to travel)
Увлекаюсь цветоводством
(Uvlekajus' cvetovodstvom)
(My hobby is gardening)

Person B

Я управляющая
(Ja upravljajushhaja)
(I am a manager)
Есть семья
(Est' sem'ja)
(I have a family)
Люблю животных
(Ljublju zhivotnyh)
(I love animals)
Много читаю
(Mnogo chitaju)
(I read a lot)
Отдыхаю на даче
(Otdyhaju na dache)
(I take rest at my country house)

Dialogue

A: Привет, как дела?
(Privet, kak dela?)
(Hello, how are you?)
B: Привет
(Privet)
(Hi)
B: Тебя как зовут
(Tebja kak zovut)
(What is your name?)
B: Меня Оля
(Menja Olja)
(I am Olya)
A: Чем занимаешься, я сейчас отдыхаю с семьёй, а ты
(Chem zanimaesh'sja, ja sejchas otdyhaju s sem'joj, a ty)
(What are you doing, I'm on holiday with my family, what about you?)
A: Виктория
(Viktorija)
(Victoria)
B: Я работаю в больнице. Я медсестра. Сейчас моя смена. Пока передышка, могу общаться

(Ja rabotaju v bol'nice. Ja medsestra. Sejchas moja smena. Poka peredyshka, mogu obshhat'sja)

(I work in a hospital. I am a nurse. Now is my shift. Having a break now, so I can chat)

B: Семья то большая? Шумно у вас?

(Sem'ja to bol'shaja? Shumno u vas?)

(Is the family big? Is it noisy?)

A: Знаешь, после утомительной работы я работаю управляющей отеля очень хочется на природу и почитать, что нибудь

(Znaesh', posle utomitel'noj raboty ja rabotaju upravljajushhej otelja ochen' hochetsja na prirodu i pochitat', chto nibud')

(You know, after a tedious job, I work as a hotel manager, I really want to go to nature and read something)

B: Я тоже читать люблю ... и цветы разводить мне тоже нравится

(Ja tozhe chitat' ljublju ... i cvety razvodit' mne tozhe nravitsja)

(I also like to read ... and I also like to grow flowers)

B: Что читаете

(Chto chitaete)

(What are you reading?)

A: Семья очень большая 18 человек и куча любимых животных

(Sem'ja ochen' bol'shaja 18 chelovek i kucha ljubimyh zhivotnyh)

(Very large family of 18 people and a lot of pets)

A: А читаю Фауст Гёте, уже в десятый раз ... у меня здесь одна книга

(A chitaju Faust Gjote, uzhe v desjatyj raz ... u menja zdes' odna kniga)

(I'm reading Goethe's Faust for the tenth time ... I only have one book here)

4. Methods

In this paper, we propose several new hybrid models for personalized conversational agents based on retrieval and generative models. Retrieval models function by ranking possible outcomes and selecting the best one according to a certain metric, in our case—finding among the set of possible answers the one most relevant to the provided context of a dialogue. The dataset is a collection of N_D dialogues $D = \{d_i | i = 1 \dots N_D\}$ where d_i is a single dialogue represented by a collection of phrases $d_i = \{d_{ij} | j = 1 \dots N_{d_i}\}$. We call the subset of phrases $\{d_{i1}, \dots, d_{i(N_{d_i}-1)}\}$ a context of the dialogue d_i and our goal is to predict the phrase $d_{iN_{d_i}}$. For predicting $d_{iN_{d_i}}$, first we obtain the vector embeddings of the context $q_i = \{q_{i1}, \dots, q_{i512}\} = Encode_{BERT_q}(\{d_{i1}, \dots, d_{i(N_{d_i}-1)}\})$ where $BERT_q$ indicates a pre-trained BERT model for contexts and $Encode_{BERT_q}$ is the encoding function of this pre-trained model that converts dialogue contexts into vector embeddings, Figure 1 illustrates this operation.

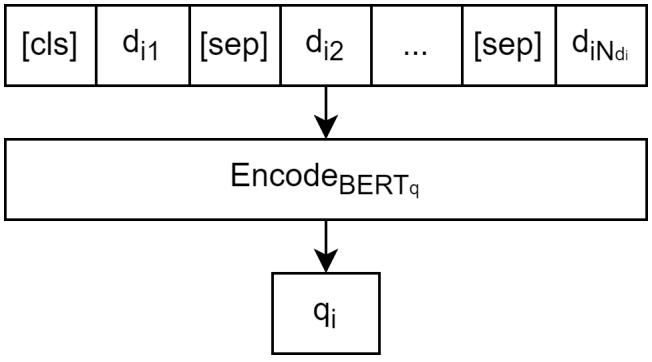

Figure 1. The scheme of the $Encode_{BERT_q}$ mapping.

For formatting the data to the acceptable by $Encode_{BERT_q}$ layout, we include an aggregating special token $[cls]$ and separating special tokens $[sep]$.

Candidates c_p for predicting $d_{iN_{d_i}}$ are vector embeddings of all phrases from all dialogues except the first phrase of each dialogue, because it is missing context. To find the best result, all candidates are collected via $c_p = \{c_{p1}, \ldots, c_{p512}\} = Encode_{BERT_c}(\{d_{kj} | k = 1 \ldots N_D, j = 2 \ldots N_{d_k}\})$, and each candidate is evaluated against the context as:

$$Sim(q_i, c_p) = \frac{q_i \times c_p}{\|q_i\| \times \|c_p\|} \tag{1}$$

All candidates are then sorted by their Sim value and the highest is selected as the output of the model:

$$Answer_{Retrieval} = c_b, \ where \ b = argmax(\{Sim(q_i, c_p)\}) \tag{2}$$

Refine models produce the answer token-by-token based on the context. Each phrase of the context $\{d_{i1}, \ldots, d_{i(N_{d_i}-1)}\}$ is tokenized via BPE [45] producing a set of context tokens $Q = \{t_1, \ldots, t_J\}$. The goal is to generate a set of tokens representing the target phrase $d_{iN_{d_i}} = \{t_{J+1}, \ldots, t_{J+maxlen}\}$, where $maxlen$ is chosen based on available computational resources for training the model.

Given P obtained by training the language mode as the probability distribution for t_i conditional to the preceding sequence of tokens $\{t_1, \ldots, t_{i-1}\}$, a refine model produces the sequence $\{t_{J+1}, \ldots, t_{J+maxlen}\}$ for which the cumulative probability is the highest:

$$Answer_{generative} = max\left(\prod_{i=J}^{J+maxlen} P(t_i | t_{j=0 \ldots i})\right) \tag{3}$$

The Retrieve and Refine model is represented by an ensemble of ranking and generative models (Figure 2) and includes the information about a person as a set of facts $G = \{g_1, \ldots, g_{N_G}\}$ where N_G is the number of facts about the person. To include the information about a person, its vector embeddings are concatenated with the context $q_i = \{q_{i1}, \ldots, q_{i512}\} = Encode_{BERT_q}(\{g_1, \ldots, g_{N_G}, d_{i1}, \ldots, d_{i(N_{d_i}-1)}\})$.

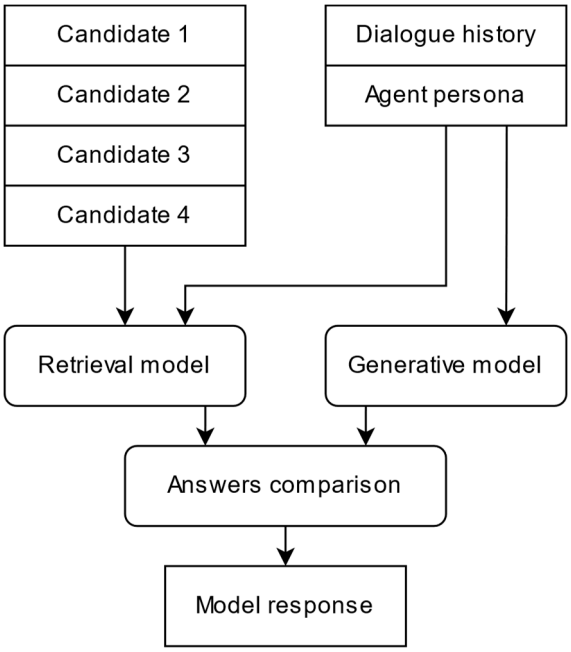

Figure 2. Retrieve and Refine hybrid model.

This design of the hybrid model suggests the production of an agent's response in parallel by both architectures of dialogue systems. After receiving the response from both models, they are compared, and the best response is selected. In the framework of this work, the proximity of the candidate found by the ranking model—which can be regarded as the degree of confidence of the model in the answer—is used as a selection criterion with a threshold TH:

$$Answer_{Retrieve\&Refine} = \begin{cases} Answer_{Retrieval}, \ if \ Sim(q_i, Answer_{Retrieval}) \geqslant TH \\ Answer_{Generative}, \ if \ Sim(q_i, Answer_{Retrieval}) < TH \end{cases} \quad (4)$$

For optimization purposes, the system first performs ranking, and then, if the answer does not exceed the dotprod similarity function threshold, the generative model produces the answer. The threshold is set according to the distribution of the responses such that every second response at the training stage exceeds the threshold value, and further, it is adjusted empirically. This strategy appears to produce the most relevant answers from the database of candidates, and if they are missing, it generates new answers, which makes the dialogue agent more flexible without limiting it to the domain presented in the database of candidates.

The Retrieval and Personifier model, similarly to the Retrieve and Refine model, is represented by an ensemble of ranking and generative models and includes the information about a person, but it performs a preliminary ranking of candidates via the personalized retrieval block described above and then the candidate with the highest rank is concatenated with the person vector and sent to the input of the generative block (Figure 3).

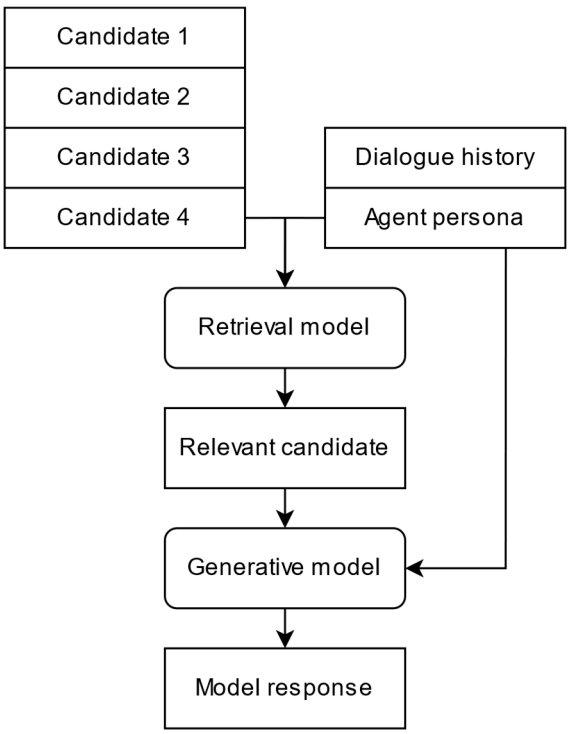

Figure 3. Retrieval and Personifier hybrid model.

Given the tokens $\{t_1, \ldots, t_J\}$ of the best candidate (with the highest value of *Sim*), they are concatenated with the facts about the person $\{t_1, \ldots, t_J, g_1, \ldots, g_{N_G}\}$. The goal of the

model is to produce a sequence of tokens $d_{iN_{d_i}} = \{t_{J+N_G+1}, \ldots, t_{J+N_G+maxlen}\}$ for which the cumulative probability is the highest:

$$Answer_{Retrieval\&Personifier} = max\left(\prod_{i=J+N_G}^{J+N_G+maxlen} P(t_i|t_{j=0\ldots i})\right) \quad (5)$$

The generative component aims for a stylistic personification of the dialogue agent's speech. The retrieval block is the same, and the candidates obtained from it at the training stage are augmented without preserving the personal style, while the source text becomes the target variable of the generative model.

The Generate and Retrieve model architecture (Figure 4) is also represented by an ensemble of ranking and generative models and includes the information about a person. It works by generating multiple possible answers via the generative block and selecting k answers with the Beam search [46] algorithms that allows us to find candidates with the highest cumulative probability for the set of tokens representing a candidate, and then selecting the best one with the retrieval block. The primary advantage of this solution is that there is no need to store a database of candidates in long-term memory.

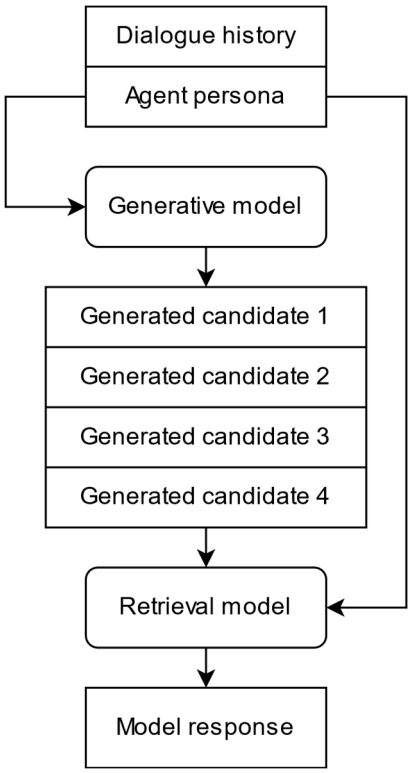

Figure 4. Generate and Retrieve hybrid model.

The Retrieve and Refine and KG model (Figure 5) is a modification of the Retrieve and Refine model that includes the knowledge-grounded generation block [47]. This hybrid approach is based on generating responses based on context and relevant knowledge from a supplementary knowledge base.

Provided a set of context tokens $\{q_1, \ldots, q_{N_q}\}$ and knowledge fragments $\{k_1, \ldots, k_{N_k}\}$ relevant to this context, they are concatenated and used as an input for the model with the

goal of producing a sequence of tokens $d_{iN_{d_i}} = \{t_{N_q+N_k+1}, \ldots, t_{N_q+N_k+maxlen}\}$ for which the cumulative probability is the highest:

$$Answer_{Retrieve\&Refine\&KG} = max\left(\prod_{i=N_q+N_k}^{N_q+N_k+maxlen} P(t_i|t_{j=0\ldots i})\right) \quad (6)$$

The architecture we propose consists of two models: a ranking search model for relevant knowledge, represented by the bi-encoder [37] architecture using BERT as an encoder, and a response generation model based on GPT-2 with a language modeling layer.

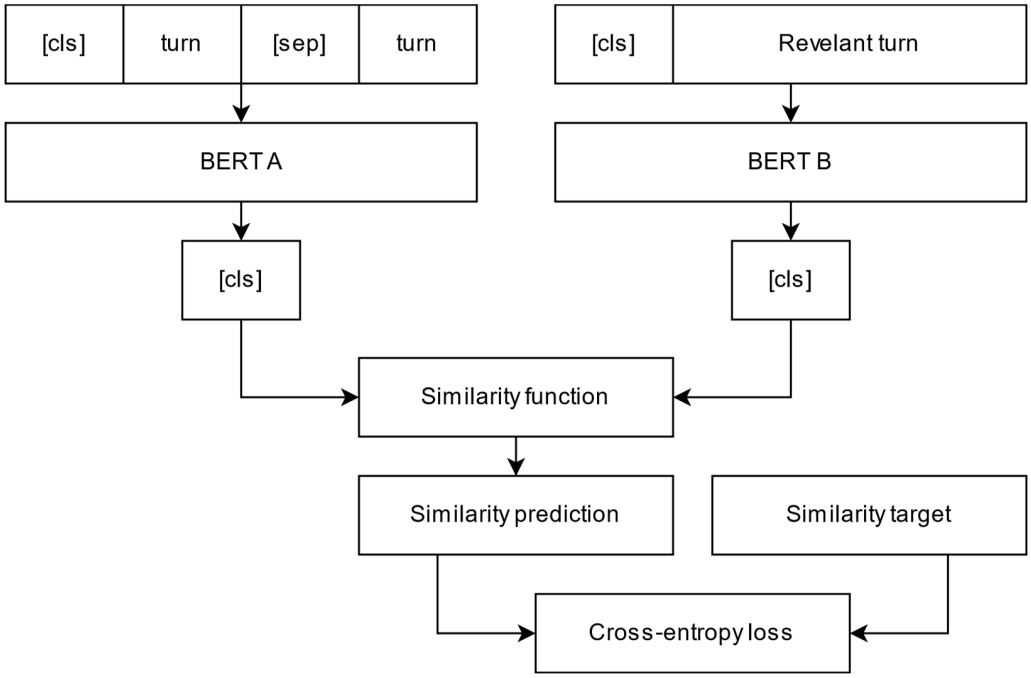

Figure 5. Retrieve and Refine and KG hybrid model.

In this work, we investigate the task of ranking the context: the history of the dialogue and personal knowledge relevant to it. The database of candidates here is a set of facts about the personality of the speaker. For validation, we use the stochastic ranking method in which a set of ranking candidates is randomly generated for some arbitrary set of contexts. This method allows us to achieve high-quality metrics for our model more efficiently and with less computational cost. For greater generalization capability for the results, we divide not only the training sample but also the dialogues from which they are produced; we use nine thousand dialogues for training and one thousand for testing.

The Toloka Persona Chat dataset is not annotated for the knowledge-grounded generation task, and for our work we previously identified relevant knowledge using the keyword method for training data; however, such markup cannot be used as ground-truth labels, so the test part of the data was annotated manually. The self-supervised approach predicates a correct approximation of the search function based on the training data, with a correction for each iteration based on the validation metrics.

First of all, let us review the principle and the structure of the knowledge search model. The training of such models is conducted by maximizing the similarity of the sequence representations obtained by the encoders. The similarity maximization problem can be transformed into a cross-entropy error minimization problem, where the similarity coefficients are passed as predictions, and the candidate relevance class is viewed as the target variables. Experiments show that the most efficient method to represent a sequence is to use the *cls*-token at the beginning of the sequence, the embedding of which aggregates

information about the entire sequence. Sequence encoders may be Siamese, i.e., context and candidate encoding can both be performed with a single instance of the BERT model. However, such architecture has a pitfall of substituting the task of predicting responses with the task of finding the most semantically close candidate because identical candidates will have the highest similarity coefficients. With that, it is more efficient to use separate encoders; however, in this case, we must understand that the amount of training data for each individual BERT will be reduced in comparison to the first approach, which can be critical when training on small volumes of data. With that in mind, we propose to use the Siamese preliminary training, after which the encoders are trained separately.

To improve the outcomes of training, we apply negative sampling. With that, the model not only maximizes the similarity of the relevant vectors but also minimizes it for the rest. This approach to training produces a higher-quality vector space in which vectors are more neatly separable. Further, we can improve the results with batch negatives sampling. With this approach, n context examples and n corresponding pairs of candidates that are relevant to them are processed in a batch. In this case, for each context i, the candidate i will be a positive example, and the remaining $n - 1$ candidates are negative with the similarity minimized during training.

When using in-batch negatives sampling, it is critical to ensure the relevant candidates are not presented as negative samples, as it greatly reduces the effectiveness of training. Due to the nature of the Toloka Persona Chat dataset, where each context can correspond to several relevant knowledge fragments at the same time and where different contexts can correspond to the same knowledge fragments, the issue can only be alleviated by picking only one training sample from each dialogue paired with one of the corresponding relevant knowledge fragments with an epoch of training.

In the original Bi-Encoder architecture, a scalar product of the encoder representations works as a similarity function. However, the dot product is not restricted, which complicates the selection of the relevance threshold and impairs the optimization of model weights during training. Here, we propose to use the scaled value of the cosine similarity of vectors. This function is a normalized scalar product with values within $[-1, 1]$. To increase the efficiency of the optimization algorithm, this coefficient is shifted towards positive values and scaled tenfold: $(cossim + 1) \times 10$.

5. Results

The evaluation results presented in this section are obtained by selecting 9018 dialogues (180,000 generated answers) for training and 995 dialogues (10,000 generated answers) for validation, and there is no intersection between the dialogues and the generated answers in the training data. The retrieval and generative models are trained with the setup listed in Table 2 and all sequences have fixed length with tail-padding and cropping of the beginning.

Table 2. Parameters for training the retrieval and generative models.

Model	Batch	Maxlen	Optimizer
Retrieval	86	context: 128 candidate: 64	AdamW, 30 epoch, warmup: 1000
Generative	32	256	AdamW, 3 epoch, warmup: 5000

To compare the dialogue models developed in this study, we employ the comparison metrics perplexity and BLEU. BLEU score [48] is calculated based on w_t: the number of n-gramms in the candidate and the maximum (referred as m_{max} between the number of n-gramms in the candidate which are also present in the reference answer of the model and the number of n-gramms in the reference answer:

$$BLEU(n) = \frac{m_{max}}{w_t} \tag{7}$$

We calculate perplexity as the exponential of the cross-entropy:

$$PP(W) = 2^{H(w)} = 2^{\frac{1}{N} log_2 P(w_1, w_2, \ldots, w_N)}, \quad (8)$$

where N is size of the dictionary of the language model.

The mean reciprocal rank is calculated as the multiplicative inverse of the rank of the first correct answer:

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} \quad (9)$$

The recall of a retrieval model is calculated as:

$$RetrievalRecall = \frac{RelevantCandidate \cap RetrievedCandidate}{RelevantCandidate} \quad (10)$$

The relevant metrics for all models are presented in Table 3 and Table 4. For the Retrieval and Personifier model, when analyzing the responses of this agent, it is evident that they are slightly semantically different from the responses of a simple personified ranking model while the generation unit principally attempts to modify the surface-level stylistic features inherent to this person. In particular, by reproducing the syntactic or morphological errors such as the replacement of the ellipsis sign "..." with ".." or dots at the end of the sentence with a line break "\n", etc. At this moment, it is not possible to objectively assess the quality of this model due to the lack of the metrics reflecting stylistic conformity.

Rigorous training of the Generate and Retrieve model requires significant computational resources; however, for training this hybrid model, we can utilize separate training where we use pre-trained generative and ranking models without further joint fine tuning. This approach will inevitably reduce the quality of the model (Table 4) but will still highlight its distinctive features at this stage. In particular, applying an additional ranking module after the generator block makes it possible to extract more meaningful candidates from the language model, but makes the generation less stable, which results in grammatically inaccurate predictions. The Retrieve and Refine model at the current stage of our research appears to be the most optimal hybrid model architecture. The decrease in quality observed in Table 4 is chiefly due to the lack of flexibility of the metrics and the lack of a unified system for an objective and comprehensive assessment of the agent's responses. To properly evaluate the performance of our models, we also include the performance metrics of the DialoGPT [10] model trained on the same dataset we use for training our models in Table 4.

For the modification of the Retrieve and Refine architecture, Retrieve and Refine and KG, Table 3 shows the results of comparing models with different similarity functions for the task of ranking context–response pairs because such a selection of samples can be considered as a baseline to a greater extent and the results of such testing are more objective. At the same time, the quality of the ranking algorithm does not depend on the type of problem being solved, and applying the model to the problem of ranking searches of knowledge produces similar results. Because *dotprod* and *cossim* produce different error values, the hyperparameter optimization of such models should be separate. Observably, none of the presented model configurations with *dotprod* outperform *cossim*.

Table 3. The results of comparing models with different similarity functions for the task of ranking context–response pairs.

Learning Rate	R1@86	mrr@86	Cross-Entropy
dotprod, one bert encoder			
1×10^{-5}	0.29	0.47	3.39
2×10^{-5}	0.30	0.48	3.10
3×10^{-5}	0.31	0.49	2.96
4×10^{-5}	0.32	0.49	2.96
5×10^{-5}	0.33	0.50	2.85
6×10^{-5}	0.33	0.49	2.82
7×10^{-5}	0.32	0.49	2.83
8×10^{-5}	0.33	0.49	2.84
9×10^{-5}	0.33	0.49	2.86
$10 \times$ cossim, two bert encoders			
5×10^{-5}	0.38	0.53	2.33
$10 \times$ cossim, one t5 encoder			
5×10^{-5}	0.42	0.56	2.15

Moving on to reviewing the principles behind the operation of the ranking model, the response generation, in this case, is a language modeling problem. The model is trained to predict a sequence consisting of a concatenated context, relevant knowledge, and a context-appropriate response. Each replica in the dialogue begins with the corresponding special token [user] or [model] depending on whom it belongs to, and the relevant knowledge fragments that starts with the [GMK] token. This way, during training, the model learns the conversational prediction pattern and aims to extract information from additional knowledge to improve the model response. Knowledge-grounded generation achieves better performance metric values when compared to the traditional generation of answers without relying on knowledge. The test results are presented in Table 4.

Table 4. Comparison of the performance metrics for the tested models.

Model	Perplexity	BLEU
DialoGPT	2.15	0.231
Retrieval and Personifier	3.96	0.019
Generate and Retrieve	3.89	0.020
Retrieve and Refine	2.99	0.116
Retrieve and Refine and KG	1.64	0.231

The inference of such models involves the ranking of relevant knowledge fragments by the search model and the concatenation of the context and the relevant knowledge that has passed the threshold. The combined text is then used as an input for the language model which predicts its continuation. The model can be operated in several modes. The first model involves supplementing the input sequence with the [model] token, indicating the point where the model should start generating its response. The second mode involves placing the [GMK] token at the end, in which case the model will generate additional knowledge about itself and will use it for generating an answer. In this case, it is necessary to preserve the new knowledge in the knowledge base of the model to maintain its consistency. The third mode, hybrid mode, proposes not to supplement the sequence, in which case the model will independently decide whether to supplement the answer with new knowledge, generate an answer, or expand and specify the last introduced knowledge. However, the latter two options require forced generation of the [model] token if it does not happen automatically.

6. Discussion

The models we explored earlier—the ranking and generative models—are, by nature, limited to specific problems that they are able to solve, i.e., the ranking models cannot produce results outside the scope of the available candidates and the generative models cannot produce results with specified, necessary content. There are certain NLP problems where this is exactly what is needed to solve the problem because the problem maps precisely to these algorithms, but more often than not, the real-world problems are more complicated or detailed. For such problems, it is still indeed helpful to have methods that can solve one part of the problem, but it is necessary to find a way to add agility to the solution. One method of achieving this is the building of hybrid models: architectures that utilize the more basic models for certain sub-processes but also introduce interactions between the components, expanding the scope of the possibly achievable results. One potential issue in this case is that, depending on how much additional agility is introduced to the system, it is possible to unintentionally substitute the problem being solved with another, slightly different problem. There are two methods to maneuver around this issue. The first method is the use of performance metrics which are precise at discriminating "good" results from "bad" ones; in this case, even if the system happens to solve—maybe even exceptionally well—the wrong problem, the metrics will quickly indicate that. Another method is to extensively manually examine the results. In this work, we demonstrate the ability of the hybrid models to expand the scope of possible results produced when solving the problem of personalization of dialogue agents; however, the lack of precise and commonly adopted performance metrics for this specific problem and the difficulty of manual evaluation of the results appear to inhibit the process of model evaluation, which is critical, particularly, for machine learning problems.

7. Conclusions

In this paper, we presented a study of hybrid models for personification of dialogue agents. We discovered that the Retrieve and Refine and KG model—a modification of the Retrieve and Refine model where the ranking and generative components work in parallel and compete based on the proximity of the candidate found by the ranking model with knowledge-grounded generation block—achieves the best performance with values of 1.64 for perplexity and 0.231 for BLEU scores, surpassing the state-of-the-art DialoGPT model when training all models on the Toloka Persona Chat Rus dataset.

It appears that knowledge-grounded hybrid models gain an advantage by utilizing the extra-linguistic knowledge obtained by a retrieval block for generation of more specific answers. This architecture is not restricted by the parametric memory of the generative model, and it has the capability to produce logical answers in a agile manner not limited by a fixed set of candidate answers.

For future work, we are interested in different methods of evaluating the quality of dialogue agents and extending the knowledge of a person by extracting it from dialogue history. This is important, because the maximum length of the context in the model currently restricts the source of knowledge to short dialogues. By broadening it, we can source the knowledge from longer dialogues, likely achieving more consistent results.

Author Contributions: Conceptualization, Y.M. and O.M.; methodology, Y.M. and O.M.; software, P.P. and A.M.; validation, P.P. and S.S.; formal analysis, Y.M. and O.M.; investigation, Y.M., O.M., P.P. and S.S.; resources, Y.M. and O.M.; data curation, P.P.; writing—original draft preparation, Y.M., O.M., P.P. and S.S.; writing—review and editing, Y.M. and O.M.; supervision, Y.M.; project administration, O.M.; funding acquisition, Y.M. All authors have read and agreed to the published version of the manuscript.

Funding: The research was financially supported by the Russian Science Foundation (project 22-11-00128).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Turing, A.M. Computing machinery and intelligence-AM Turing. *Mind* **1950**, *59*, 433 . [CrossRef]
2. Bogoradnikova, D.; Makhnytkina, O.; Matveev, A.; Zakharova, A.; Akulov, A. Multilingual Sentiment Analysis and Toxicity Detection for Text Messages in Russian. In Proceedings of the 2021 29th Conference of Open Innovations Association (FRUCT), Tampere, Finland, 12–14 May 2021; pp. 55–64. [CrossRef]
3. Makhnytkina, O.; Matveev, A.; Bogoradnikova, D.; Lizunova, I.; Maltseva, A.; Shilkina, N. Detection of Toxic Language in Short Text Messages. In Proceedings of the International Conference on Speech and Computer, St. Petersburg, Russia, 7–9 October 2020; Karpov, A., Potapova, R., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 315–325.
4. Mairesse, F.; Walker, M. Automatic recognition of personality in conversation. In Proceedings of the Human Language Technology Conference of the NAACL, New York, NY, USA, 4–9 June 2006. [CrossRef]
5. Mairesse, F.; Walker, M. PERSONAGE: Personality generation for dialogue. In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic, 23–30 June 2007.
6. Li, J.; Galley, M.; Brockett, C.; Spithourakis, G.P.; Gao, J.; Dolan, B. A persona-based neural conversation model. *arXiv* **2016**, arXiv:1603.06155
7. Kottur, S.; Wang, X.; Carvalho, V.R. Exploring personalized neural conversational models. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017. . [CrossRef]
8. Qian, Q.; Huang, M.; Zhao, H.; Xu, J.; Zhu, X. Assigning personality/profile to a chatting machine for coherent conversation generation. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018. [CrossRef]
9. Zhang, S.; Dinan, E.; Urbanek, J.; Szlam, A.; Kiela, D.; Weston, J. Personalizing dialogue agents: I have a dog, do you have pets too? *arXiv* **2018**, arXiv:1801.07243.
10. Wu, Y.; Ma, X.; Yang, D. Personalized Response Generation via Generative Split Memory Network. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 6–11 June 2021; pp. 1956–1970. [CrossRef]
11. Sordoni, A.; Galley, M.; Auli, M.; Brockett, C.; Ji, Y.; Mitchell, M.; Nie, J.Y.; Gao, J.; Dolan, B. A neural network approach to context-sensitive generation of conversational responses. *arXiv* **2015**, arXiv:1506.06714.
12. Song, H.; Zhang, W.N.; Hu, J.; Liu, T. Generating persona consistent dialogues by exploiting natural language inference. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020. [CrossRef]
13. Liu, Q.; Chen, Y.; Chen, B.; LOU, J.G.; Chen, Z.; Zhou, B.; Zhang, D. You Impress Me: Dialogue Generation via Mutual Persona Perception. *arXiv* **2020**, arXiv:2004.05388.
14. Lin, Z.; Liu, Z.; Winata, G.I.; Cahyawijaya, S.; Madotto, A.; Bang, Y.; Ishii, E.; Fung, P. XPersona: Evaluating Multilingual Personalized Chatbot. *arXiv* **2021**, arXiv:2003.07568.
15. Kim, H.; Kim, B.; Kim, G. Will I sound like me? Improving persona consistency in dialogues through pragmatic self-consciousness. *arXiv* **2020**, arXiv:2004.05816.
16. Bao, S.; He, H.; Wang, F.; Wu, H.; Wang, H. PLATO: Pre-trained Dialogue Generation Model with Discrete Latent Variable. *arXiv* **2020**, arXiv:1910.07931.
17. Madotto, A.; Lin, Z.; Bang, Y.; Fung, P. The Adapter-Bot: All-In-One Controllable Conversational Model. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020. [CrossRef]
18. Roller, S.; Dinan, E.; Goyal, N.; Ju, D.; Williamson, M.; Liu, Y.; Xu, J.; Ott, M.; Shuster, K.; Smith, E.M.; et al. Recipes for building an open-domain chatbot. *arXiv* **2021**, arXiv:2004.13637.
19. Xu, X.; Gou, Z.; Wu, W.; Niu, Z.Y.; Wu, H.; Wang, H.; Wang, S. Long Time No See! Open-Domain Conversation with Long-Term Persona Memory. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, 22–27 May 2022.
20. Zheng, Y.; Chen, G.; Huang, M.; Liu, S.; Zhu, X. Personalized Dialogue Generation with Diversified Traits. *arXiv* **2019**, arXiv:1901.09672.
21. Dinan, E.; Logacheva, V.; Malykh, V.; Miller, A.; Shuster, K.; Urbanek, J.; Kiela, D.; Szlam, A.; Serban, I.; Lowe, R.; et al., The Second Conversational Intelligence Challenge (ConvAI2). In *The NeurIPS'18 Competition*; Springer: Cham, Switzerland, 2020. ._7. [CrossRef]
22. Qian, H.; Li, X.; Zhong, H.; Guo, Y.; Ma, Y.; Zhu, Y.; Liu, Z.; Dou, Z.; Wen, J.R. Pchatbot: A Large-Scale Dataset for Personalized Chatbot. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 11–15 July 2021. [CrossRef]
23. Toloka. Toloka Persona Chat Rus. Available online: https://toloka.ai/datasets (accessed on 9 November 2022).
24. Matveev, A.; Makhnytkina, O.; Matveev, Y.; Svischev, A.; Korobova, P.; Rybin, A.; Akulov, A. Virtual Dialogue Assistant for Remote Exams. *Mathematics* **2021**, *9*, 2229. [CrossRef]
25. Posokhov, P.; Apanasovich, K.; Matveeva, A.; Makhnytkina, O.; Matveev, A. Personalizing Dialogue Agents for Russian: Retrieve and Refine. In Proceedings of the 2022 31st Conference of Open Innovations Association (FRUCT), Helsinki, Finland, 27–29 April 2022; pp. 245–252. [CrossRef]

26. Makhnytkina, O.; Matveev, A.; Svischev, A.; Korobova, P.; Zubok, D.; Mamaev, N.; Tchirkovskii, A. Conversational Question Generation in Russian. In Proceedings of the 2020 27th Conference of Open Innovations Association (FRUCT), Trento, Italy, 7–9 September 2020; pp. 1–8. [CrossRef]
27. Kulikov, I.; Miller, A.H.; Cho, K.; Weston, J. Importance of a Search Strategy in Neural Dialogue Modelling. *arXiv* **2018**, arXiv:1811.00907.
28. Yavuz, S.; Rastogi, A.; Chao, G.L.; Hakkani-Tür, D. DEEPCOPY: Grounded response generation with hierarchical pointer networks. *arXiv* **2019**, arXiv:1908.10731.
29. Wolf, T.; Sanh, V.; Chaumond, J.; Delangue, C. TransferTransfo: A Transfer Learning Approach for Neural Network Based Conversational Agents. *arXiv* **2019**, arXiv:1901.08149.
30. Li, P. An Empirical Investigation of Pre-Trained Transformer Language Models for Open-Domain Dialogue Generation. *arXiv* **2020**, arXiv:2003.04195.
31. Madotto, A.; Lin, Z.; Wu, C.S.; Fung, P. Personalizing dialogue agents via meta-learning. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020. [CrossRef]
32. Mazaré, P.E.; Humeau, S.; Raison, M.; Bordes, A. Training millions of personalized dialogue agents. *arXiv* **2018**, arXiv:1809.01984.
33. Cao, Y.; Bi, W.; Fang, M.; Shi, S.; Tao, D. A Model-Agnostic Data Manipulation Method for Persona-based Dialogue Generation. *arXiv* **2022**, arXiv:2204.09867.
34. Song, H.; Wang, Y.; Zhang, K.; Zhang, W.N.; Liu, T. BoB: BERT over BERT for training persona-based dialogue models from limited personalized data. *arXiv* **2021**, arXiv:2106.06169.
35. Zheng, Y.; Zhang, R.; Mao, X.; Huang, M. A Pre-Training based personalized dialogue generation model with persona-sparse data. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020. [CrossRef]
36. Gu, J.C.; Liu, H.; Ling, Z.H.; Liu, Q.; Chen, Z.; Zhu, X. Partner Matters! An Empirical Study on Fusing Personas for Personalized Response Selection in Retrieval-Based Chatbots. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 11–15 July 2021. [CrossRef]
37. Humeau, S.; Shuster, K.; Lachaux, M.A.; Weston, J. Poly-encoders: Transformer Architectures and Pre-training Strategies for Fast and Accurate Multi-sentence Scoring. *arXiv* **2019**, arXiv:1905.01969.
38. Kasahara, T.; Kawahara, D.; Tung, N.; Li, S.; Shinzato, K.; Sato, T. Building a Personalized Dialogue System with Prompt-Tuning. *arXiv* **2022**, arXiv:2206.05399.
39. Das, S.; Saha, S.; Srihari, R.K. Using Multi-Encoder Fusion Strategies to Improve Personalized Response Selection. *arXiv* **2022**, arXiv:2208.09601.
40. Bao, S.; He, H.; Wang, F.; Wu, H.; Wang, H.; Wu, W.; Guo, Z.; Liu, Z.; Xu, X. PLATO-2: Towards Building an Open-Domain Chatbot via Curriculum Learning. *arXiv* **2020**, arXiv:2006.16779.
41. Sun, Y.; Wang, S.; Li, Y.; Feng, S.; Tian, H.; Wu, H.; Wang, H. ERNIE 2.0: A Continual Pre-training Framework for Language Understanding. *arXiv* **2019**, arXiv:1907.12412.
42. Kim, Y. Convolutional Neural Networks for Sentence Classification. *arXiv* **2014**, arXiv:1408.5882.
43. Sugiyama, H.; Mizukami, M.; Arimoto, T.; Narimatsu, H.; Chiba, Y.; Nakajima, H.; Meguro, T. Empirical Analysis of Training Strategies of Transformer-based Japanese Chit-chat Systems. *arXiv* **2021**, arXiv:2109.05217.
44. Posokhov, P.; Matveeva, A.; Makhnytkina, O.; Matveev, A.; Matveev, Y. Personalizing Retrieval-Based Dialogue Agents. In Proceedings of the International Conference on Speech and Computer, Gurugram/New Delhi, India, 14–16 November 2022; Prasanna, S.R.M., Karpov, A., Samudravijaya, K., Agrawal, S.S., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 554–566.
45. Sennrich, R.; Haddow, B.; Birch, A. Neural Machine Translation of Rare Words with Subword Units. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016.
46. Huang, L.; Zhao, K.; Ma, M. When to finish? Optimal Beam Search for Neural Text Generation (modulo beam size). In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 7–11 September 2017.
47. Zhao, X.; Wu, W.; Xu, C.; Tao, C.; Zhao, D.; Yan, R. Knowledge-grounded dialogue generation with pre-trained language models. *arXiv* **2020**, arXiv:2010.08824.
48. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. BLEU: A Method for Automatic Evaluation of Machine Translation. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Philadelphia PA, USA, 7–12 July 2002; Association for Computational Linguistics: Philadelphia PA, USA, 2002; pp. 311–318. [CrossRef]