Distribution Agreement

Signature:

Zihao Wang Date

Contextual Embedding Representations for Retrieval-Based and Generation-Based Dialogue Systems

By

Zihao Wang
Doctor of Philosophy

Computer Science and Informatics

Jinho Choi, Ph.D.
Advisor

Joyce Ho, Ph.D.
Committee Member

Davide Fossati, Ph.D.
Committee Member

Shamim Nemati, Ph.D.
Committee Member

Accepted:

Kimberly Jacob Arriola, PhD, MPH
Dean of the James T. Laney School of Graduate Studies

Date

Contextual Embedding Representations for Retrieval-Based and Generation-Based Dialogue Systems

By

Zihao Wang
B.E., Dalian University of Technology, China, 2012
M.Sc., Carnegie Mellon University, PA, 2013

Advisor: Jinho Choi, Ph.D.

An abstract of
A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Computer Science and Informatics
2023

Abstract

Contextual Embedding Representations for Retrieval-Based and Generation-Based Dialogue Systems
By Zihao Wang

Context is a crucial element for conversational agents to conduct natural and engaging conversations with human users. By being aware of the context, a conversational agent can capture, understand, and utilize relevant information, such as named entity mentions, topics of interest, user intents, and emotional semantics. However, incorporating contextual information into dialogue systems is a challenging task due to the various forms it can take, the need to decide which information is most relevant, and how to organize and integrate it.

To address these challenges, this dissertation proposes exploring and experimenting with different contextual information in the embedding space across different models and tasks. Furthermore, the dissertation develops models that overcome the limitations of state-of-the-art language models in terms of the maximum number of tokens they can encode and their incapacity to fuse arbitrary forms of contextual information. Additionally, diarization methods are explored to resolve speaker ID errors in the transcriptions, which is crucial for training dialogue data.

The proposed models address the challenges of context integration into retrieval-based and generation-based dialogue systems. In retrieval-based systems, a response is selected and returned by ranking all responses from different components. A contextualized conversational ranking model is proposed and evaluated on the MSDialog benchmark conversational corpus, where three types of contextual information are leveraged and incorporated into the ranking model: previous conversation utterances from both speakers, semantically similar response candidates, and domain information associated with each candidate response. The performance of the contextual response ranking model exceeded state-of-the-art models in previous research, showing the potential to incorporate various forms of context into modeling.

In generation-based systems, a generative model generates a response to be returned to the conversing party. A generative model is built on top of the Blenderbot model, overcoming its limitations to integrate two types of contextual information: previous conversation utterances from both conversing parties and heuristically identified stacked questions that tackle repetition and provide topical diversity in dialogue generations. The models are trained on an interview dataset and evaluated on an annotated test set by professional interviewers and students in real conversations. The average satisfaction score from professional interviewers and students is 3.5 out of 5, showing promising future applications.

Additionally, to better understand topics of interest, topical clustering and diversity are investigated by grouping topics and analyzing the topic flow in the interview conversations. Frequent occurrences of some clusters of topics give a clear presentation of what scopes of topics an interview would touch on while maintaining a great selection of unique topics for individuals. Based on the observation, further discus-

sions on the potential incorporation of such characteristics to improve conversational dialogue models are conducted.

Contextual Embedding Representations for Retrieval-Based and Generation-Based Dialogue Systems

By

Zihao Wang
B.E., Dalian University of Technology, China, 2012
M.Sc., Carnegie Mellon University, PA, 2013

Advisor: Jinho Choi, Ph.D.

A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Computer Science and Informatics
2023

Acknowledgments

I would like to express my deepest gratitude to the following individuals for their unwavering support and encouragement throughout my academic journey:

First and foremost, I would like to thank my dissertation advisor, Dr. Jinho Choi, for his invaluable guidance, insights, and patience throughout the entire process. His expertise and commitment to excellence have been instrumental in shaping this dissertation.

I would also like to thank the members of my dissertation committee, Dr. Davide Fossati, Dr. Joyce Ho, and Dr. Shamim Nemati, for their constructive feedback and valuable suggestions. Their diverse perspectives and areas of expertise have greatly enriched this dissertation.

I am grateful to the faculty and staff of Department of Computer Science and Informatics for creating a supportive and intellectually stimulating environment. Their dedication to teaching and research has inspired me to pursue my academic goals with passion and perseverance.

My deepest gratitude goes to my family for their unconditional love and support. Their belief in me has been my source of strength and motivation. I am also grateful to my friends for their encouragement, laughter, and consistent support.

Lastly, I would like to express my appreciation to all the participants who generously gave their time and insights for this study. Without their contributions, this dissertation would not have been possible.

Thank you all for your support, guidance, and encouragement.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the fast advancement of artificial intelligence, research and applications of conversational agents are trending. Some conversational agents, such as Amazon Alexa, Apple Siri, Microsoft Cortana, and Google Assistant, are popular in the real world, most being assemblies of close-domain components. There is more effort in research and collaborations between academia and industries toward open-domain conversations, such as Alexa Prize [1].

I participated in Alexa Prize for three years, with two years as the team lead. Alexa Prize is a competition event held by Amazon Alexa each year for university students to develop dialogue systems to conduct fluent and smooth conversations. In the Alexa Prize, we developed modules in the dialogue system to consider multiple aspects of the context in the conversations, such as name entities, user identities, stats on discussed topics, intents, etc. We made different strategies to incorporate different types of context in the pipeline. For example, we designed proactive follow-up strategies to continue the conversation by proposing deep discussions on the current topic. We also designed mechanisms to ensure the smooth flow of conversations. For example, we established rules for domain-based components to capture users' intent whether or not they are transitioning to a different topic. Many of the designs in our dialogue system

[1] https://developer.amazon.com/alexaprize

were intended to grasp the context embedded in the conversation flow and exploit it to make smooth conversations. The general user feedback on the performance of our dialogue systems inspires the importance of context in the design of dialogue systems.

With extensive research and development of dialogue systems, many perform excellently in certain task-oriented conversations. However, they cannot conduct fluent, consistent, and open-domain conversations. One of the limitations is the lack of effective contextual information incorporation. Current deep-learning language models have shown exceptional advantages in understanding and utilizing ambient context on different tasks. However, they still cannot effectively exploit prolonged conversational history with other contextual features. To tackle the above limitations, my dissertation consists of different models to enable the integration of prolonged and miscellaneous contexts in the embedding space to improve the performance of conversational agents, as well as insights into conversation flow by analysis of different perspectives of conversational datasets.

In general, there are two popular structures of dialogue systems, retrieval-based and generation-based systems. A retrieval-based dialogue system consists of components that generate candidate responses. There is a response ranking function to rank all candidates and select the best one as the chosen response. A generation-based dialogue system is usually referred to as using a deep generation neural network to generate word by word or character by character an utterance in a dialogue. The focus of my dissertation is on the development of models to improve response ranking and utterance generation by effective integration of various contextual information, in addition to the understanding of conversational flow.

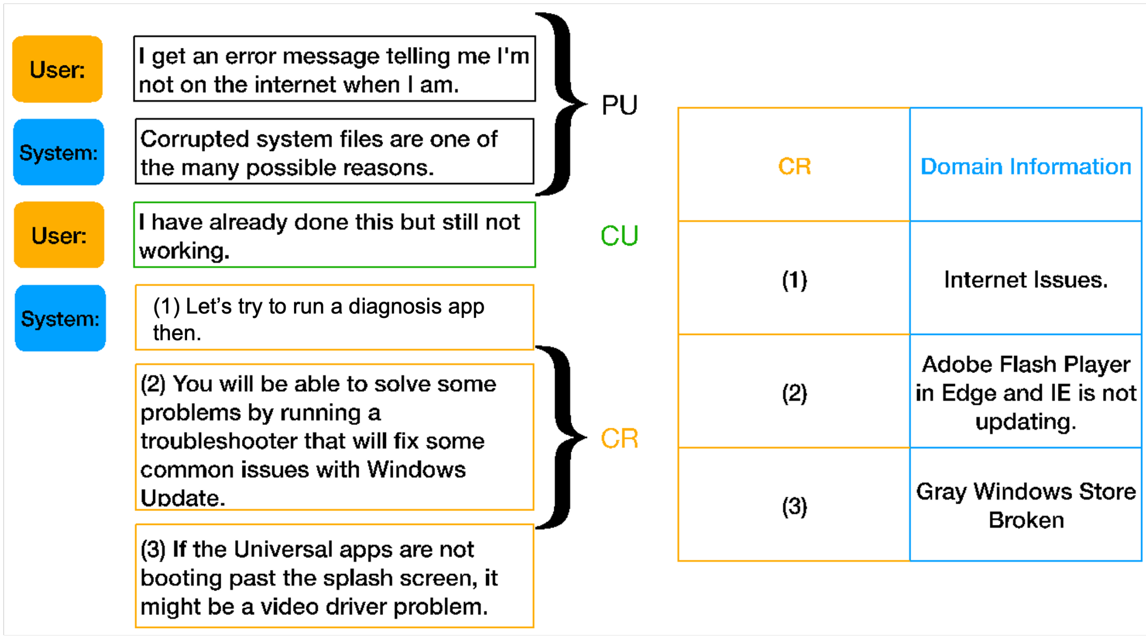

Figure 1.1: MSDialog conversation example. The abbreviations denote the following. CU: current utterance, PU: previous utterances, CR: candidate responses.

1.1 Response Ranking in Dialogue Systems

In retrieval-based dialogue systems, response ranking is essential to select the best candidate response among all generated from different components. The difficulty of response ranking comes from the semantic resemblance of candidates. As shown in Figure 1.1, there are three candidate responses for the current utterance, and candidate (1) is the golden truth, while the other two are negative choices. Semantically, they are all describing computer-related issues, which are similar. Without further context, it is challenging to differentiate between them. In these cases, it is critical to have other surrounding knowledge, such as domain information, as shown in the figure, to make distinguishing decisions.

Extensive research [74, 70, 69] has been conducted on utilizing conversation history with external knowledge to select the most natural response from the candidates as the response. However, in many cases, the candidate responses are semantically similar, even if components generate them in entirely different domains. Most previous work hasn't considered the integration of other contextual information, such as provenances of candidates, other than conversation history into response ranking models in retrieval-based dialogue systems. Instead, I consider it tremendously significant for response ranking models to learn from conversation history and other contextual information jointly.

Practical applications to integrate multiple context sources were explored in the dialogue systems [61, 2] in Alexa Prize, in which candidate responses generated from a dozen components designated for different domains are ranked to have the best one returned to a user. A hybrid weighted and heuristic ranking strategy was designed to consider topic classification, intent classification, conversation transition, special case handling, and personalization. Amazon Alexa users received the ranking strategy well, which is an inspiration and testimonial for tackling the above challenges.

Reported in this dissertation, I establish a multi-channel conversational ranking architecture with variations to take different sources of information in parallel and fuse them together in the embedding space to benefit the model from all available evidence. I perform ablation studies to prove that contextual information and our effort of adapting transformer self-attention architectures in our model significantly improve the ranking performance compared to previous research.

1.2 End-to-end Generative Dialogue Systems

Neural generation-based end-to-end dialogue systems are advancing rapidly. One of its advantages is that it simplifies the complication of managing components in a dialogue system as in retrieval-based systems. Extensive work has been done on developing such neural models for dialogue generations[72, 1, 51]. However, the current state-of-the-art neural dialogue models still lack the capability to conduct conversations with consistency, fluency, commonsense and deeper discussions on specific topics, which in essence, is a consequence of ineffective incorporation of context. To tackle the above limitations of the current generative models, I develop a context-aware architecture with variations and experiment on a real-world interview dataset collected from college admission interviews, which is a mixture of close-domain and open-domain conversations. Our context-aware generative deep learning architecture plays the role of an interviewer, questioning on key perspectives of an interviewee, following up on brought-up topics on the fly, making transitions to the next topics, and proposing proper endings. Many conversations are conducted yearly with students applying to U.S. colleges from over 100 countries. The interview conversations are unscripted discussions on many perspectives of students' academic and extracurricular activities. We project that applying our dialogue model will benefit thousands of students by practicing interviews cost-effectively and time-efficiently and save time and effort for constantly scheduling interviewers to lead interviews. Our models are evaluated by standard generation metrics and satisfaction levels of interactions by professional interviewers and students. Our best model could conduct fluent conversations for around 30 turns, generating thought-provoking questions and interacting with deeper discussions on specific topics.

1.3 Topic Flow in Conversations

In conversations, diverse topics of interest could be involved. The overall conversation flow could be more clear if the topics involved in the conversation were ordered. However, how to extract topics from utterances, what the topics that lead interview conversations are, what the common and diverse topics are in the conversation, and how to extract topic flow embedded in the conversations are critical research questions. Previous research rarely conducts in-depth investigations into conversations, especially on the topic flow. To understand the characteristics of interview conversations, I cluster key questions into different topics, investigate the popularity of different topics and present the topic flow embedded in conversations. Investigating the intrinsic topics in interview conversations gives insight into further building contextualized conversational models.

1.4 Summary of Contributions

I develop conversational dialogue models to incorporate contextual information in the embedding space to improve the performance of conversational agents. Specifically, for retrieval-based dialogue systems, which could consist of multiple components to generate candidate responses, the best model organizes and fuses the current utterance, conversation history, and the source information of candidates. In addition, the model has a parallel multi-channel architecture, which can be extended to integrate other context information. The model outperforms the current state-of-the-art models by noticeable margins. For generative dialogue systems, I develop a generative architecture on top of the state-of-the-art Blenderbot. The model overcomes its limitations of the length of input and the lack of capability to integrate multiple sources of contextual information in the embedding space. This architecture can also be extended to integrate other sources of contextual information with arbitrary text lengths. The model is evaluated by 15 professional interviewers and students and achieves an average of 3.5 out of 5 on overall satisfaction. Furthermore, in-depth organizing and analysis of clusters of topics in the conversations present a better understanding of topic flow in interview conversations, which leads to a new model proposal that could generate the next topic of interest and the next turn of utterance at the same time, while boosting the performance of each other.

1.5 Dissertation Structures

In the following chapters of my dissertation, I describe context-based models, their associated datasets, relevant data processing procedures, and evaluation procedures and metrics. Specifically, in Chapter 2, I give more detailed background and related literature to contextualize our contributions. Chapter 3 describes the ranking strategy applied in the Alexa Prize and different variations of our conversational ranking architecture to integrate different contextual information, in addition to experiment setups, evaluation, limitations, and conclusion. Chapter 4 describes our conversational generation architecture with variations to integrate different context information. Furthermore, during data processing, due to the noisy nature of the transcriptions, I develop text-based speaker diarization models to reassign speaker IDs to sentences in the transcriptions to clean up the dataset. I also describe experiment setups, evaluation metrics, limitations, and conclusions. Chapter 5 analyzes the interview dataset and presents insights into topical clusters and diversity which are reflections of the conversation flow. Chapter 6 gives a discussion on the investigation of the role and integration of contextual information, especially topic flow representations, in dialogue systems.

Chapter 2

Background

In this chapter, I introduce my work in Alexa Prize and conclusions on the importance of context in dialogue system design. In addition, I conduct a brief literature review to place my research in context. In the literature review, first, I give a brief introduction of dialogue system. Second, I give a review on the learning to rank which natural leads to the response ranking task in dialogue systems. Then, I describe previous research on context integration for response ranking models. After that, generation-based system and state of the art generation models are reviewed.

2.1 Alexa Prize

During Alexa Prize, a tremendous amount of effort for introducing context information into the flow of conversations. The dialogue systems we designed are retrieval based systems. I describe mainly my work that involve the integration of context.

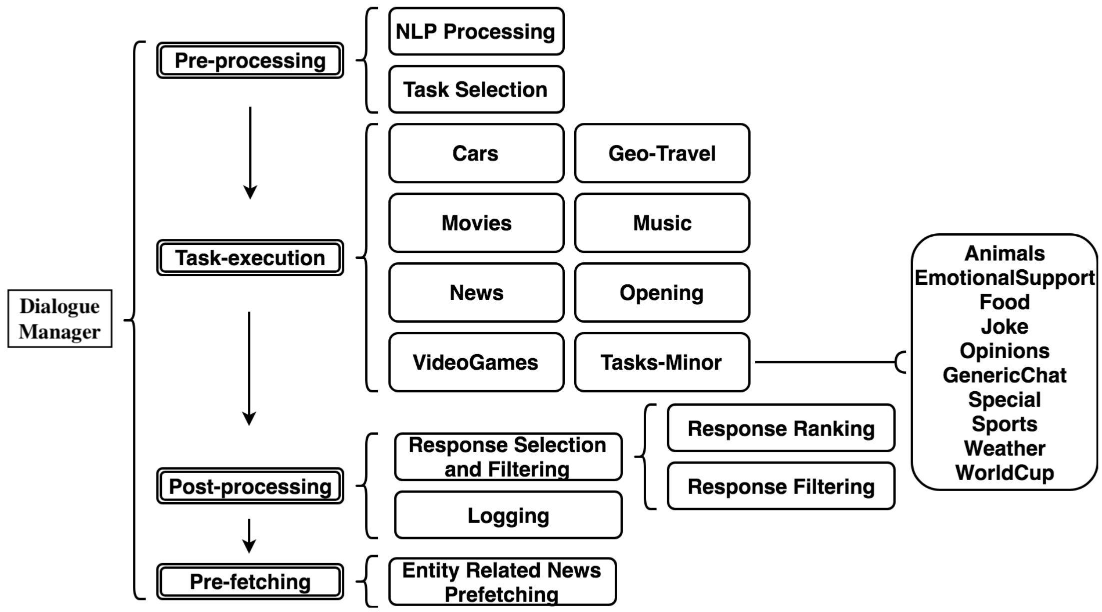

Figure 2.1: Dialogue Manager Framework

2.1.1 Dialogue Manager Framework Implementation and Design

The dialogue manger in the Alexa Prize processes utterances to generate responses using four main execution steps, which are Pre-processing, Task execution, Post-processing and background information pre-fetching as outined in Figure 2.1. The work flow of the dialogue manager is process that allows the the dialogue system to encode, understand and exchange the conversation context to output a response to a user. Specifically, the pre-processing module has a dialogue logger that stores all dialogue state related information in variables, required for tracking the progress of conversations. There are three types of variables, state variables, cache variables, and context variables. State variables are predefined variables to indicate different states of a conversation. Based on different information retrieval components and their capabilities of extending discussions, different states are correspondingly planned. State variables are checked and updated per turn of conversation. Cache variables store detailed and categorized information of a topic and can be retrieved later for deeper topic discussion. Context variables store all the utterances and responses in sequence. Context variables can also be retrieved, mainly for the use of context understanding, such as co-reference resolution function to resolve ambiguous pronouns. The dialogue logger is accessible to all domain-based components and help the components to make in-component decisions to generate context-related candidate responses.

Scenario	Decision
Instruction Request	Special State Component
Pause and Hesitation	Special State Component
SelfHarm	Special State Component
Stop	Special State Component
Opening and Greeting	Conversation Opening Component
Repetition	Special State Component
Information Request	Domain Components

Table 2.1: Conversation States and Component Assignment for Retrieving Responses.

In addition, the features extracted by NLP processes are maintained in the conversation context. Based on context information, the dialogue manager then selects components to take in context and retrieve responses. The are 6 scenarios, as shown in Table 2.1, for the dialogue manager to make different decisions. For example, if the user intent is Repetition, only the Special State Component will be selected to repeat the last response. If the user intent is Information Request, all domain components would have an equal chance to retrieve responses. Domain components retrieve responses in parallel threads sharing context information.

2.1.2 Contextualized Proactivity

There are mechanisms in the Alexa Prize I involved designing and implementing to increase the proactivity and engagement to improve the satisfaction of users. First, a user behavior cache is maintained and later stored in the conversation. Statistics is collected on topics of interest, topics of rejection, returning information, etc. Second, another dictionary is maintained to collect statistics on topics on different time periods of the day. By implement these mechanics, the system can make proactive recommendation during transitions to other topics and increases the success rate to continue the conversation.

	Rating	# of turns
Before Proactivity	2.96	9.66
after proactivity	3.22	10.9

Table 2.2: Results comparison before and after implementation of contextualize proactivity.

As shown in Table 2.2, after the implementation of the proactivity as context, the overall satisfaction of users and the number of turns in the conversation have both increased.

The dialogue system we built in the Alexa Prize clearly demonstrates the importance of context in the design of dialogue systems, which leads to my research focus as reported in this dissertation.

2.2 Dialogue Systems

Dialogue systems refer to computer programs that interact with humans using natural languages. A dialogue system commonly consists of multiple components, such as input interface, natural language understanding, dialogue manager, response generation, etc.

Dialogue systems date back to 1960s, when rule-based dialogue systems such as ELIZA [62] and PARRY [13] started to be active in the research area. Later on, miscellaneous task-oriented dialogue systems started to be implemented both in research and industry to assist on different tasks. Extensive research has been conducted on task-oriented dialogue systems [48, 8, 33, 30, 9]. The above systems require extensive effort for customization for different tasks and can not be adapted for open-ended conversations.

With the availability of large datasets of conversations, deep learning based models start to prevail with their flexible adaptations to different domains by finetuning on designated datasets [60, 10, 33]. Especially in recent advancement, transformer-based models have been performing well with respect to understanding ambient context [59, 46, 31, 73].

2.3 Response Ranking

I now briefly review related work in response ranking in dialogue systems. First, I review general learning to rank approaches, which I adapt to the conversational setting. Second, I summarize the most recent response ranking models as a transition to our model. And then, I review topic modeling and classification in dialogues as it is important and relevant to response ranking in dialogue systems. Last, I review ranking tasks integrating external knowledge.

2.3.1 Learning to rank

Learning to rank approaches have applications in various fields, such as information retrieval and natural language processing. BM25 [49] and its variants have been widely received as reliable baseline methods. Later, supervised machine learning was adapted to ranking tasks, such as the SVM ranking model proposed by [53]. As neural networks started arising, Ranknet[11] and LambdaMart[64] were a series of improvements based on gradient descent methods. However, these algorithms highly rely on the richness of extracted features, while feature selection methods often compromise semantic meanings.

2.3.2 Neural response ranking models

The upsurge of Word2Vec [39] and the development of neural network models facilitated learning-to-rank performance, and they are quickly adapted to dialogue response ranking tasks. Variations of Convolutional Neural Networks (CNN) [68], Recurrent Neural Networks (RNN) [37], and the combination of the two [12] have been explored to push the frontline forward. Most recently, the sequential matching network (SMN) [63], deep matching network with external knowledge [69], deep attention model [47] and the intent-aware model [70] have achieved state of the art respectively.

2.3.3 Topic modeling and classification in dialogues

Topic modeling and classification are critically important to understand users' topics of interest in a conversation, and it is critical to a dialogue system to acquire candidates from knowledge sources based on topic modeling and classification. Jokinen et al. [26] defined topic trees to use topical information for conversational robustness. Latent Dirichlet Allocation (LDA) was applied by [71] to detect topics in conversational systems. However, when applied to dialogues, unsupervised models can only infer topics from lexical statistics, which are not always consistent with conversation context. Supervised methods such as the supervised LDA by [38] and a Deep Average Network-based model [21] further improved topic understanding in either text or dialogues. Most recently, [3] proposed an entity-aware topic classification model to facilitate the understanding of topics with entities. After all, the above models missed the link between topics and conversation context.

2.3.4 Ranking with integration of external knowledge

Integration of external knowledge has a long history in document ranking and retrieval tasks [7, 14, 16]. Various sources of knowledge are utilized to improve the performance of ranking. Hovy et al. [23] uses well-constructed WordNet and QA typology to improve performance on a Question-Answerign system. Wikipedia was used as external knowledge to improve the document clustering tasks by [24]. Other research also incorporates entities for document ranking [66].

In this paper, I utilize both knowledge source information associated to candidate responses, and conversation history to perform the response ranking task in a multi-turn conversation.

2.4 Dialogue Generation

Dialogue systems can be categorized into closed- and open-domain systems [25]. Closed-domain systems require efficient access to domain knowledge [34] and serve specific professions such as education [15], healthcare [19, 4], or customer service [6, 43]. Open-domain systems converse across multiple domains with natural transitions [1] and conduct interactions in a broader horizon [2, 61, 20]. In this section, I give a brief update on the dialogue generation models as well as the current applications in real-world situations.

2.4.1 Dialogue Generation Models

Dialogue generation models start to populate when sequence to sequence models are used to generate text [58]. Currently, the frontier of the dialogue generation models is based on the transformer architecture [59] and pretrained language models [36, 17]. Blenderbot was developed [50] to conduct chit-chat conversations with training data from a mixture of Blended Skilled Talk and Switchboard. The model achieved great performance with respect to the smoothness of the conversation flow. Summary-based [27] dialogue systems [67] stores the long-term context by summarizing conversational history. However, there is are thresholds of the length of utterances they can take as well as the flexibility of fusing multiple forms of context into dialogue models. In the dissertation, I develop an architecture that incorporate arbitrary forms and lengths of context into dialogue generation and achieved high performance evaluated by professional users.

2.4.2 Current Applications of Generative Dialogue Systems For Admission Interviews

For admission interviews, the conversation is often a mixture of closed (job-related questions) and open domains (general aspects of the applicant) dialogues, which makes it more challenging to build an end-to-end system. Several dialogue systems have been developed to communicate with humans for information exchange or elicitation [20, 32, 29] across multiple domains [52, 28, 44]. Conversational agents for interviews have been experimented with for law enforcement [40], healthcare [42], job application [65], and psychology [55], among which most are proof of concept. A few interviewbots have been developed on commercial platforms such as Google Dialogflow and IBM Watson Assistant, with the limitation of pre-scripted interviews; thus, they cannot proactively follow up to the user contents. However, the current applications either rely heavily on the templates, which makes conversations robotic, or can not conduct deep discussions on topics in open-domain conversations.

Chapter 3

Contextual Response Ranking

In this Chapter, I first review on the response ranking strategy in the Irisbot mentioned in chapter 1, and then I describe our contextualized and deep learning-based response ranking models.

3.1 Response Ranking Strategy In Irisbot

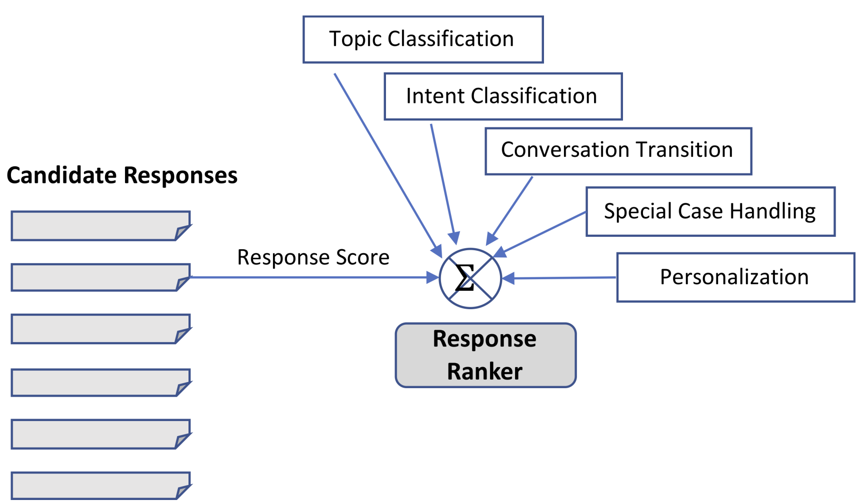

Figure 3.1: IrisBot Contextual Response Ranker with a hybrid weighted and heuristic ranking strategy.

IrisBot makes decisions on the final response returning to an Amazon Alexa user by the Response Ranker. Instead of only relying on the topic and intent classifier, the response ranker takes into account a variety of response and context features and handles additional domain-specific cases and states, as illustrated in Figure 3.1. Specifically, for special cases, especially continuing on the same topic as before, recovering from an error condition, or responding to profanity, the ranker uses a pre-defined strategy. For all other cases, a weighted combination of topic and intent classification scores is combined with the response scores, and other contextual information is encoded as features for the ranker. The weighted scoring linear combination model was tuned based on domain knowledge and development data but can be easily extended to use additional features and more sophisticated learning-to-rank models, trained over the data from the conversations in the Semi-finals.

The most important special cases handled by the ranking model include:

- Component Follow-up: in order to maintain conversation flow, the response ranker prioritizes responses from the most recently active domain component, which may have asked the person a question on a previous conversation turn, and is now expecting to provide a follow-up response. For example, if a person is speaking to the bot about Movies, and asked the customer about their favorite genre, the response from Movies about the customer's genre utterance will be prioritized. However, if the customer response is on another topic, the normal weighted ranking flow will be used.

- Special state handling: in case of ASR confusion or other error such as component response timeout, the ranking model attempts to select the next most appropriate response, followed by a topic suggestion.

- Dealing with rejection: topic suggestion: a common case is that a person rejects a topic of conversation proposed by the system. In that case, the ranking strategy will recommend another topic. This recommendation can be random, deterministic (following a pre-defined conversation "script", or personalized:

prioritizing topics predicted to be interesting to the customer.

From the Response Ranker in Irisbot in the Alexa Prize, I realized that it is important to take into account all available sources of context, which is proven to be very helpful especially in a real-world application. The inspirations and testimonials from Alexa Prize lead to my research in contextual models for dialogue systems.

3.2 Contextualized Response Ranking Models

In this section, we, in sequence, define the problem setting (Section 3.2.1), give an overview of the proposed approach (Section 3.3.1), explain the framework architecture in detail, and present two specific settings for ablation studies (Section 3.3.2).

3.2.1 Task Formulation

I now formulate the response ranking more formally. Given a dialogue D, at turn t, there is a conversation history $\{u^0, u^1,...,u^t\}$, and a given set of response candidates $\{c_0, c_1, ..., c_j, ..., c_k\}$ with their associated domain information $\{p_0, p_1, ..., p_j, ..., p_k\}$, from which they are curated. The instantiated task is to leverage conversation history and candidates' domain information to make ranking decisions to user utterances.

3.3 Approach and Implementation

3.3.1 Approach Overview

I approach the conversational response ranking problem with a bi-channel end-to-end pipeline, to fuse contextual information both from conversation history and candidate responses. First, conversation history interacts with candidate responses and their domain information turn by turn, respectively, to build up interaction representations

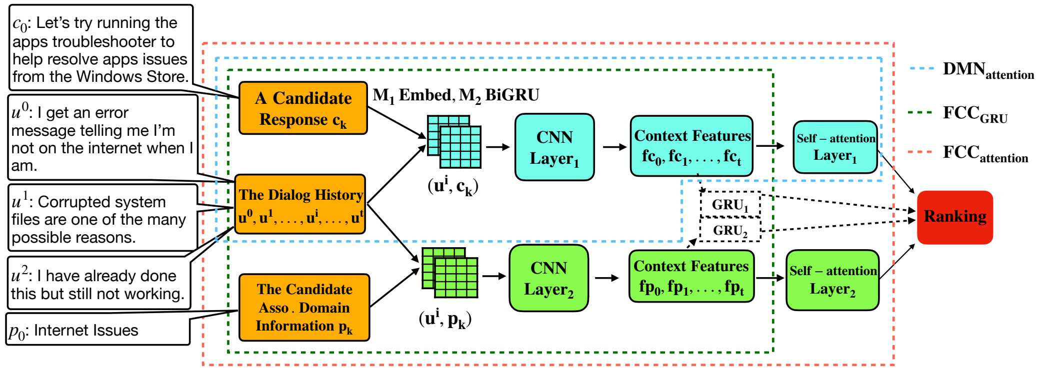

Figure 3.2: The architectures of the response ranking with domain information and GRU layers (FCC_{GRU}), and with domain information and attention layers ($FCC_{attention}$). Symbols denote as follows. c_0: the 0th candidate the current utterance; u^i: the ith utterance in the dialogue, $i \in [0, 1, 2]$; p_0: the domain information associated to candidate c_0.

in each channel. And then, self-attention is applied to each channel to model conversation dependencies. Finally, the output from both channels are concatenated for ranking.

3.3.2 Model Architecture

This section first introduces representation modules of the framework, including interaction matrix representation, textual feature representation, and latent ranking representation. I then describe the specific implementation integrating domain information from candidate provenance besides conversation history, taking advantage of both contextual information sources. Following that, I describe self-attention layers in the conversation history. Furthermore, I designate two other framework settings for ablation studies. Finally, I elaborate on the generalization of our model as a flexible framework.

The initial interactions between the two channels adopt the basic structure of the DMN model by [69] for the following reasons. First, the interaction matrix in DMN has its advantage over other text-matching representations [45]. Second, this representation consists of both embedding and hidden state features, which have performed well in the previous state-of-the-art ranking models [70]. Third, the use of CNN to capture high-level n-gram textual features has been proven to be effective. Last, the GRU module can model sequential relationships. Our proposed framework with ablation studies has improved over the DMN models and is a fair comparison to their performance.

Interaction matrix representation. At conversation turn j, u^j, c_k, or p_k is represented by a sequence of word embeddings E_u^j, E_c^k or E_p^k, and fed into a shared BiGRU to get hidden states, H_u^j, H_c^k, and H_p^k respectively. The embedding interaction matrix between an utterance and a candidate response is calculated by $M_{u_e}^{c_e} = E_u^j \cdot (E_c^k)^T$. The hidden state interaction matrix is calculated by $M_{u_h}^{c_h} = H_u^j \cdot (H_c^k)^T$. The same procedure is actualized to have $M_{u_e}^{p_e}$ and $M_{u_h}^{p_h}$ between an utterance and domain information of a candidate response.

Textual feature representation. The interaction matrix representation is fed into a CNN layer, obtaining $c_j^{u,*}$ (* denotes either candidate response c or topic information p), the n-gram textual feature representation for each turn in the conversation.

Latent conversation history representation. I use a GRU or a self-attention layer for modeling conversation history. However, the self-attention layer is more potent in various tasks [59]. This module $DMN_{attention}$ is applied to the comprehensive conversation history features $C_u^* = [c_0^{u,*}, c_1^{u,*}, ..., c_t^{u,*}]$. The hidden states $R_u^* = [r_0^{u,*}, r_1^{u,*}, ..., r_t^{u,*}]$ from the module are concatenated for ranking.

Model architectures. Here, I fit the representation modules in the $DMN_{attention}$ model setting, the proposed framework with domain information and GRU layers FCC_{GRU}, and that with domain information and attention layers ($FCC_{attention}$). All the models are shown in Fig 3.2 with different legends.

- The $DMN_{attention}$ model is developed to explore how the self-attention layer affects the ranking performance. This model takes candidate responses and dialogue history as input to obtain interaction matrices $M_{u_e}^{c_e}$ and $M_{u_h}^{c_h}$. The CNN layer takes in interaction matrices and outputs a textual feature representation C_u^c. A self-attention layer is applied to C_u^c to acquire a latent conversation history representation.

- The FCC_{GRU} model is developed to explore how domain information affects ranking performance. It takes candidate responses, their corresponding domain information, and dialogue history to create interaction matrices $M_{u_e}^{c_e}$, $M_{u_h}^{c_h}$, $M_{u_e}^{p_e}$, and $M_{u_h}^{p_h}$. The CNN layers take in interaction matrices and output textual feature representations C_u^c and C_u^p. The GRU layers take textual feature representations and output latent conversation history representation R_u^c and R_u^p.

- The $FCC_{attention}$, model follows the same flow as the FCC_{GRU} model, but instead of GRU layers, two self-attention layers are applied to obtain latent conversation history representations.

- The ranking layer takes in R_u^c for the $DMN_{attention}$ model, and $concat(R_u^c, R_u^p)$ for the FCC_{GRU} and the $FCC_{attention}$ models, and outputs a ranking score for each candidate response.

Framework Generalization. The domain information and previous utterances can be replaced, and the parallel structure of the framework can further be expanded to channel in, other contextual features, such as outsourced external knowledge, as an integral part of the end-to-end neural ranking pipeline, to enhance the contextual enrichment.

Framework Summary. In summary, I present our new framework for conversational response ranking, *FCC*, which introduces the following new ideas compared to prior work: 1. an introduction of candidate provenance as a new channel to add to conversation history, generating a compact yet comprehensive representation of a dialogue; 2. an implementation of self-attention layers to improve the modeling of multi-turn dependency; 3. our channelized framework easily being expanded to integrate other contextual features in parallel to further enhance contextual enrichment.

3.4 Experiments

In this section, I describe experiments in three parts. First, I describe the benchmark MSDialog dataset in Section 3.4.1. Next, I describe experimental procedures in Section 3.4.2, which include three experiments: 1. A study on the performance of $FCC_{attention}$; 2. An ablation study comparing a self-attention layer and a GRU layer to model multi-turn dependency; 3. An ablation study on the effect of domain information of candidates on ranking performance. Last, I summarize experimental results comparing with the state-of-the-art baselines in Section 3.4.3.

Table 3.1: Comparison of different models over MSDialog. Numbers in bold font mean the result is better compared with the best baseline $IART$ models. $*$ means statistically significant difference over the best baseline $IART_{Bilinear}$ with $p < 0.05$ measured by the Student's t-test. † means statistically significant difference over FCC_{GRU} model with $p < 0.05$ measured by the Student's t-test. § means statistically significant difference over $DMN\text{-}PRF$ with $p < 0.05$ measured by the Student's t-test.

Data	MSDialog			
Metrics	R10@1	R10@2	R10@5	MAP
DMN-KD [69]	0.4908	0.7089	0.9304	0.6728
DMN-PRF [69]	0.5021	0.7122	0.9356	0.6792
DAM [74]	0.7012	0.8527	0.9715	0.8150
$IART_{Dot}$ [70]	0.7234	0.8650	0.9772	0.8300
$IART_{Outerproduct}$ [70]	0.7212	0.8664	0.9749	0.8289
$IART_{Bilinear}$ [70]	0.7317	0.8752	0.9792	0.8364
$DMN_{attention}$	0.5544§	0.7579§	0.9507§	0.7180§
FCC_{GRU} (our framework)	**0.770***	**0.8780**	0.9717	**0.8548***
$FCC_{attention}$ (our framework)	**0.7879*†**	**0.8992*†**	**0.9810†**	**0.8697*†**

3.4.1 Dataset

The MSDialog conversational dataset is collected from the Microsoft products online forum, which discusses issues in a miscellaneous assortment of domains. It includes more than 35,000 conversations and more than 337,000 utterances. I use the subset MSDialog-ResponseRank dataset processed by [47]. In the MSDialogue dataset, candidate responses are extracted from conversations discussing various issues. These issues are summarised in the "title" fields in the dataset, which is a fair comparison to domain information of specific components in retrieval-based dialogue systems. Therefore, I take "title" fields as our domain information for candidates and this information is reasonably straightforward and easy to get in a dialogue system.

I use Matchzoo[1] as the data preprocessing tool. Each ranking list, which has one true response and nine candidate responses, is converted to a pair-wise ranking setting. Each true response will be ranked against each candidate response.

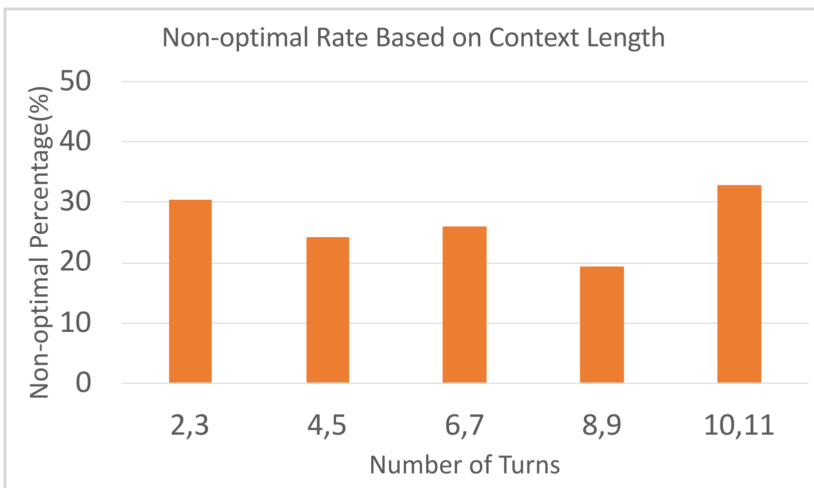

Figure 3.3: The Non-optimal Rate over Conversation History Length

3.4.2 Experimental Setup

There are over 173k samples in the training set, and 37k and 35k in the validation and testing sets. I implement the models using Pytorch [2]. For CNN layers, I use two convolution and max-pooling sub-layers with the number of filters (16, 16), convolu-

[1]https://github.com/NTMC-Community/MatchZoo
[2]https://pytorch.org/

tional kernels (3, 3), max-pooling kernels (2, 2) and strides (1, 1). The self-attention layer has two heads and two encoder blocks. I train the models on the ranking corpus to gain pre-trained embeddings with dimension 200 with the Word2Vec tool [39]. The maximum number of turns in a dialogue is 10. The maximum sequence length for utterance and candidate response is 90 and 30 for the domain information sequence. The batch size is 50. The modes are tuned by the validation dataset.

3.4.3 Model evaluation

In this section, I first report the performance of $FCC_{attention}$, comparing with the state-of-the-art baselines in response ranking and response selection fields. And then, I show the results of ablation studies on the impact of domain information and self-attention layers. Experiment results are reported in Table 3.1.

Main results. I evaluate $FCC_{attention}$, on $R10@1$, $R10@2$, $R10@5$, and MAP. The results show that $FCC_{attention}$, has an improvement on all four metrics over the state of the art $IART$ models, especially on R10@1, R10@2, and MAP, which all have significance p-value < 0.05. The performance on recall@1 has the most significant 7.7% improvement, which is most important since a dialogue system usually picks the best candidate to return to a user.

The ablation study on domain information. To study the impact of domain information compared with the DMN models, I evaluate FCC_{GRU} on the same metrics. The results show that with an extra channel to integrate domain information from candidates to the DMN architecture, the ranking performance improves significantly, with margins between 2.2% to 38.9% corresponding to different metrics. The ranking performance not only surpasses the DMN models but has significant improvements on recall@1 and MAP over $IART$ models, with margins of 5.0% and 2.2%. This comparison confirms the positive effect of domain information from the candidates. The domain information provides

The ablation study on self-attention layers. I conduct an ablation study on the effect of self-attention layers over conversation history. The $DMN_{attention}$ model has improved over the DMN-PRF model with margins ranging from 1.6% to 10.4%. The $FCC_{attention}$ model surpasses the performance of the FCC_{GRU} model with improvement ranging from 1.0% to 2.4%. From the results, it is clear that the self-attention layer impacts positively the ranking performance.

Furthermore, I analyze the non-optimal rate (percentage of cases in which the true response is not ranked first), to explore the effectiveness of the self-attention layer conditioning on conversation history length. It is demonstrated that the non-optimal rate drops from about 30% to 20% as the length of conversation history increases until a sudden surge in conversations with 10 and 11 (maximum length) turns. It is reasonable to conjecture that when the conversation only has a few turns, such as 2 or 3, the model is not fed with enough contextual information to make an optimal decision. While in the opposite case, the model isn't sophisticated enough to isolate effective information from over-long conversations. The self-attention layers are most effective in conversations with 4 to 9 turns.

3.5 Discussion and Conclusion

The flexible framework (*FCC*) is capable of incorporating miscellaneous contextual resources for response ranking in multi-turn dialogue systems. To validate the framework, I implement embedding domain information of candidates with self-attention layers to improve the relevance modeling between utterances and candidate responses.

Specifically, the domain information adds a second source to interact with utterances, a mechanism to either confirm or alleviate the semantic matching just between conversation history and candidates. One of the examples as a demonstration here:

–Utterance: ...message telling me I am not on the internet while I am ...

–Candidate 1: You will be ...running a troubleshooter... to fix some common issues with Window Update. (Domain Info: Adobe Flash Player in Edge and IE is not updating from vulnerability.)

–Candidate 2: Let's try running ... troubleshooter to help resolve app issues from the Windows Store. (Domain Info: Internet Issues.)

The trained $DMN_{attention}$ model ranked Candidate 1 first, without knowing the domain information. However, FCC models successfully ranked Candidate 2 first since the domain knowledge directly points to the intention of the user. This example clearly supports our claim that domain knowledge from the source of candidates enhances the effectiveness of a response ranking model.

The overall result supports our claim as well, by outperforming existing state-of-the-art models, with ablation studies to show that both domain information of candidates and self-attention layers lead to critical increments in the performance respectively and conjunctively.

Chapter 4

Response Generation in Dialogue Systems – InterviewBot

4.1 Interview Dataset

Audio recordings of 7,361 interviews are automatically transcribed with speaker identification by the online tool RevAI[1], where 440 are manually corrected on speaker ID assignment for finetuning and evaluation of our models (Table 4.1). Each recording contains an average of a $\approx$ 15-min long dialogue between an interviewer and an interviewee. The interviews were conducted by 67 professionals in 2018 - 2022. The largest age group of interviewees is 18-years-old with 59.3%, followed by 17-years-old with 29.4%. The male-to-female ratio is 1.2:1. The major country of origin is China with 81.4% followed by Belgium with 10.5%, alongside 37 other countries.

All recordings are transcribed into text and speakers are identified automatically. For speech recognition, three tools from Amazon[2], Google[3], and RevAI are assessed on 5 recordings for speaker diarization, achieving the F1-scores of 66.3%, 50.1%, and

[1] `https://www.rev.ai`
[2] `https://aws.amazon.com/transcribe`
[3] `https://cloud.google.com/speech-to-text`

72.7%, respectively.[4]

	D	U	S1	S2
TRN	140	43.8	39.3	64.0
DEV	150	45.0	36.2	60.3
TST	150	44.3	37.8	61.3
RAW	6,921	40.4	41.5	67.6

Table 4.1: Distributions of our data. D: num of dialogues, U: avg-num of utterances per dialogue, S1/S2: avg-num of tokens per utterance by interviewer/interviewee. TRN/DEV/TST: training/development/evaluation (annotated) sets. RAW: unannotated set (auto-transcribed).

4.2 Speaker Diarization

Speaker diarization is a task of segmenting an audio stream into utterances according to the speaker identity, and considered critical in automatic transcription [5]. Conversation data with diarization errors can lead to major failure of building robust dialogue models. Particularly, the alternation of conversation parties may be disturbed by speaker ID errors in the training. Our most accurate transcriber, RevAI, still gives 27.3% errors for speaker diarization (Section 4.1). The main reason is that audios from the interviewer (S1) and the interviewee (S2) are recorded in one channel, so that they are saved in a single waveform, while no clear pauses exist between S1 & S2's speeches or their speeches often overlap. The following example illustrates when the speech of S2 (underlined) is not recognized as a separate utterance:

S1: Hi , it 's nice to meet you . Nice to meet you. Um , can you tell me what is a topic that um , you cannot stop talking about?

[4]The same metric as in Table 4.4 is used for this evaluation.

Thus, speaker diarization models are developed to re-align utterances from conversation parties to provide clean data to our dialogue model (Sec. 4.3). Figure 4.1 depicts the distributions of different types of diarization errors found in 100 dialogues. Most errors are caused by filler words and arbitrary concatenation (joining multiple utterances as one with no apparent patterns, not caused by filler words).

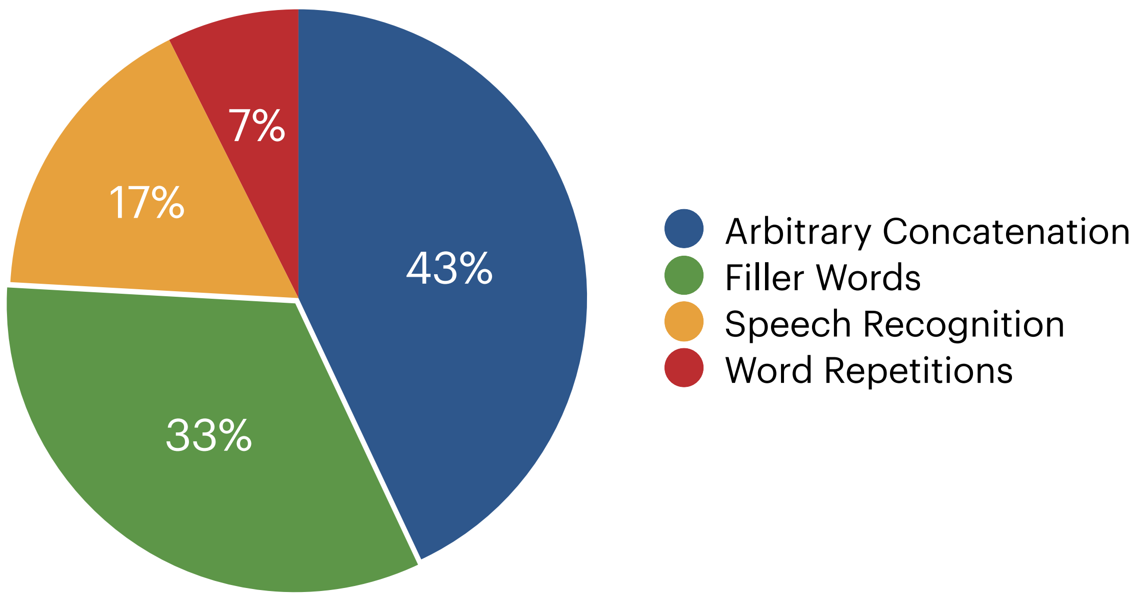

Figure 4.1: Distributions of the diarization error types.

4.2.1 Manual Annotation

440 dialogues are sampled, in which every token is annotated either `1` if it is one of the last two tokens of an utterance before the speaker is switched, and `0` otherwise. For the above example, the $8-9^{th}$ tokens are the last two tokens of the utterance before it switches to `S2` and so are the $13-14^{th}$ tokens before switching to `S1`; thus, they are annotated `1`:[5]

Hi	,	it	's	nice	to	meet	you	.	Nice	to	meet	you	.
0	0	0	0	0	0	0	1	1	0	0	0	1	1

[5] I also annotated only the last token as `1`, or annotated all words from `S0` as `0` and from `S1` as `1`, which yielded worse results in terms of the end performance.

Doccano is used as the annotation tool [41], and ELIT is used for the tokenization [22]. To measure the inter-annotator agreement, 10 dialogues are double-annotated that show the high kappa score of 84.4%.

4.2.2 Pseudo Annotation

Because our annotated data are relatively small, a larger dataset is pseudo-created for this task using 2,400 dialogues in the Switchboard [57] and 6,808 dialogues in the BlendedSkillTalk [56] datasets (thus, a total of 9,208 dialogues). These two datasets are chosen because their dialogues sound more speech-originated than others, having an adequate amount of filler words. Among the 4 types of diarization errors (Figure 4.1), the ones caused by filler words (33%) can be simulated on dialogues that do not contain such errors using statistical heuristics[6].

#	**Dist**	*okay*	*yeah*	*right*	*um*	*so*	*uh*	*well*	*like*	*oh*
2	40.4	46.7	16.0	8.0	8.5	8.0	4.4	4.7	0.2	3.1
3	35.9	33.3	29.8	3.9	8.5	11.1	6.2	2.3	0.6	4.1
4	8.6	33.7	24.5	5.6	9.7	11.2	5.1	3.6	1.0	5.1
5	7.3	28.9	30.7	6.6	4.2	15.1	6.6	4.8	0.6	2.4

Table 4.2: Distributions of filler words w.r.t. diarization errors. Dist: percentage of dialogues containing # number of utterances with errors caused by the filler words. *filler_word*: percentage of the filler word appearing in the corresponding dialogue group.

The errors associated with filler words are pseudo-inserted to dialogues from the two datasets by finding an utterance either beginning or ending with a filler word, and concatenating it with an utterance prior or next to it. Global search is made to the entire dialogues for finding such utterances to mimic the distributions in Table 4.2 such that about 40.4% of the dialogues in the pseudo-created data would contain 2 utterances with diarization errors, where 46.7% of them are caused by the filler

[6]Filler words are inferred by the outputs of the part-of-speech tagger and the dependency parser in ELIT.

word *okay*, and so on. It is possible that more than two utterances get joined; in our case, up to 8 utterances are concatenated. Table 4.2.3 includes the statistics of our pseudo-created dataset for transfer learning.

4.2.3 Joint Model

Figure 4.2 shows an overview of our speaker diarization model. Let $U_i = \{w_i^\circ, w_{i1}, .., w_{in}\}$ be the ith utterance to be handled, where w_i° is the special token representing U_i and w_{ij} is the jth token in U_i.

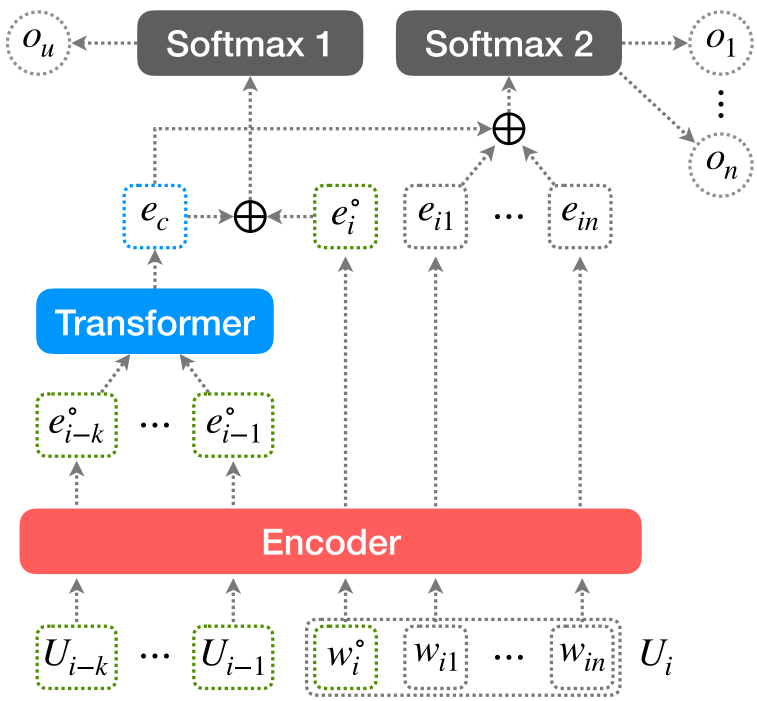

Figure 4.2: The overview of our diarization model.

U_i is fed into the encoder $\mathcal{E}$ that generates the embeddings $\{e_i^\circ, e_{i1}, .., e_{in}\}$. The previous utterances $\{U_{i-k}, .., U_{i-1}\}$ are also fed into $\mathcal{E}$ that generates $\{e_{i-k}^\circ, .., e_{i-1}^\circ\}$ (in our case, $k = 5$ that is the context window). For utterance-level weighting, these embeddings are fed into a transformer layer, which creates the context embedding e_c. Finally, $e_c \oplus e_i^\circ$ is fed into a softmax layer that outputs o_u to make a binary decision of whether or not U_i includes any error. Jointly, each $e_c \oplus e_{ij}$ is fed into another softmax that outputs o_j to decide whether or not w_{ij} is one of the last two tokens of an utterance. The embeddings are initialized by the pre-trained language model Roberta. The default AdamW optimizer and learning rate are used due to negligible differences in model performance through experiments.

	Switchboard				**BlendedSkillTalk (BST)**				**Interview Data (Before)**				**(After)**		
	D	**U**	**S1**	**S2**	**D**	**U**	**S1**	**S2**	**D**	**U**	**S1**	**S2**	**U**	**S1**	**S2**
`TRN`	1,115	42.2	31.6	31.3	4,819	6.2	25.0	25.2	140	42.2	37.4	73.1	43.8	39.3	64.0
`DEV`	21	16.6	49.4	43.7	1,009	6.1	25.8	25.3	150	44.1	34.9	67.3	45.0	36.2	60.3
`TST`	19	32.7	32.9	32.9	980	6.2	26.2	26.2	150	44.2	34.2	69.0	44.3	37.8	61.3
`RAW`	N/A				N/A				6,921	39.6	38.2	75.1	40.4	41.5	67.6

Table 4.3: Distributions of the pseudo-created datasets (Switchboard, BST) and our interview data (before and after diarization). D: number of dialogues, U: avg-number of utterances, S1/S2: avg-number of tokens per utterance by `S1`/`S2`. `TRN`/`DEV`/`TST`: training/development/evaluation (annotated) sets. `RAW`: unannotated set. Note that I follow the same splits suggested by the original papers of the Switchboard and BST datasets for comparability.

4.2.4 Experiments

For the encoder, the RoBERTa large model is used [35][7]. Table 4.2.3 shows the distributions of the pseudo-created data (Section 4.2.2), as well as our interview data (Section 4.1) before and after the diarization where errors in the train/dev/test sets are manually annotated (Section 4.2.1) and errors in the raw set are automatically corrected by the joint model (Section 4.2.3). After diarization, `S2`'s utterances with diarization errors get split such that the average length of `S2`'s utterances decreases while the average length of dialogues slightly increases. Meanwhile, some parts of `S2`'s utterances, incorrectly separated from `S1`'s utterances by the transcriber, are recovered back to `S1`; thus, the average length of `S1`'s utterances increases.

Table 4.4 shows results of three models: the *baseline* model taking U_i and producing $O_w = \{o_1, .., o_n\}$, the *context* model taking $\mathcal{U}_c = \{U_{i-k}, .., U_i\}$ and producing O_w, as

[7]Several transformer encoders including BERT [18] were evaluated and RoBERTa yielded the best results.

well as the *joint* model taking $\mathcal{U}_c$ and producing o_u and O_w (Figure 4.2). The baseline model does not create e_c, so e_{i*} are directly fed to Softmax 2. Also, the baseline and context models do not use e_i°, so only Softmax 2 is used to produce the outputs. For evaluation, the F1-scores of the label `1` on the last two tokens are used. All models are developed three times and their average scores and standard deviations are reported.

	Baseline	**Context**	**Joint**
Ours only	92.9±0.4	92.9±0.3	92.9±0.2
Transferred	93.2±0.3	93.4±0.3	**93.6±0.3**

Table 4.4: Diarization model performance. Ours: trained on `TRN` of the Interview Data (After) in Table 4.2.3. Transferred: trained first on the `TRN` mixture of Switchboard and BST, then finetuned on `TRN` of the Interview Data.

When trained on only our data, all models perform similarly. The joint model slightly outperforms the others when transfer learning is applied. Although the improvement is marginal, the joint model has a benefit of identifying utterances with diarization errors, showing the F1 score of 93.6% for this task, while the transferred models generally show much higher performance on the other datasets than the non-transferred models. Thus, the joint transferred model is used to auto-correct all dialogues in `RAW`.

4.3 Dialogue Generation

Figure 4.3 depicts an overview of our dialogue generation model. Since inputs to the encoder $\mathcal{E}$ and the decoder $\mathcal{D}$ are limited by the total number of tokens that the pre-trained language model accepts, *sliding window* (Sec. 4.3.1) and *context attention* (Sec. 4.3.2) are proposed to handle long utterances and contexts in the previous utterances, respectively. In addition, *question storing* is used to remember user-oriented topics brought up during the interview (Sec. 4.3.3). The input to $\mathcal{E}$ and output of $\mathcal{D}$ include the speaker ID `S1` or `S2` as the first token followed by an utterance from the interviewer or interviewee, respectively. Hyperparameters are finetuned by cross validations.

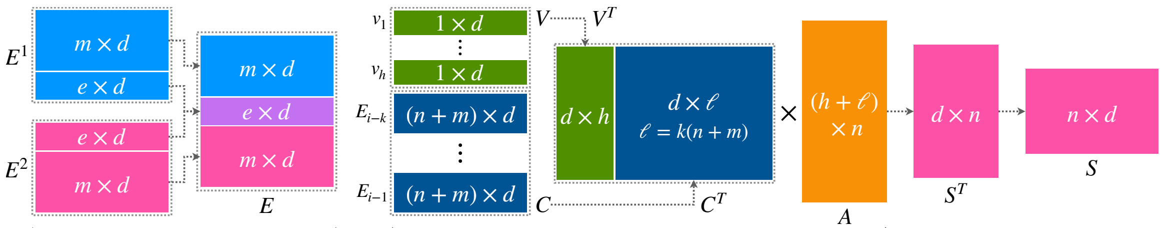

Figure 4.3: The overview of our dialogue generation model.

4.3.1 Sliding Window

Let $n = m + e$ be the max-number of tokens that $\mathcal{E}$ and $\mathcal{D}$ accept ($e < m < n$). Every utterance U whose length is greater than n is split into U^1 and U^2 as follows (w_i is the ith token in U):

$$U^1 = \{w_1, \ldots, w_m, w_{m+1}, \ldots, w_n\}$$
$$U^2 = \{w_{m+1}, \ldots, w_n, w_{n+1}, \ldots, w_{n+m}\}$$

In our case, $n = 128$, $m = 100$, and $e = 28$ such that $n + m = 228$ is sufficiently long enough to handle most utterances based on our stats. $\mathcal{E}$ takes U^1 and U^2 then produces $E^1 = \{e^1_1, \ldots, e^1_n\}$ and $E^2 = \{e^2_{m+1}, \ldots, e^2_{n+m}\}$ where $e^*_i \in \mathbb{R}^{1\times d}$ is the embedding of w_i. Finally, the embedding matrix $E \in \mathbb{R}^{(n+m)\times d}$ of U is created by stacking all of the following embeddings:

$$\{e^1_1, \ldots, \frac{1}{2}\sum_{i=1}^{2}(e^i_{m+1}), \ldots, \frac{1}{2}\sum_{i=1}^{2}(e^i_n), \ldots, e^2_{n+m}\}$$

For utterances whose lengths are less than or equal to n, zero-padding is used to transform $\mathcal{E}$'s output from $\mathbb{R}^{n\times d}$ to $\mathbb{R}^{(n+m)\times d}$.

4.3.2 Context Attention

Let U_i be the $i\acute{t}h$ utterance to be generated as output. Let $C \in \mathbb{R}^{\ell \times d}$ be the context matrix stacking the embedding matrices of the previous utterances $\{E_{i-k}, .., E_{i-1}\}$, where k is the number of previous utterances to be considered and $\ell = k(n+m)$. The transpose of C is multiplied by the attention matrix $A \in \mathbb{R}^{\ell \times n}$ such that $C^T \cdot A \rightarrow S^T \in \mathbb{R}^{d \times n}$. Thus, $S \in \mathbb{R}^{n \times d}$ represents the context summary of $U_{i-k}, .., U_{i-1}$, which is fed into the decoder $\mathcal{D}$.

4.3.3 Question Storing

Even with the context attention, the model still has no memory of contexts prior to U_{i-k}, leading it to repeat the same topics that it has already initiated. To overcome this issue, question storage is introduced to remember key questions derived by the interviewer. Every interview in our data came with 8-16 questions by the interviewer, who used those questions during the interview and thought they led to assessing crucial aspects of the interviewee. Our final model considers these "key leading questions" and dynamically stores them as the dialogue progresses.

Let $Q = \{q_1, .., q_h\}$ be the leading question set. During training, $\mathcal{D}$ learns to generate `Q` instead of `S1` as the first token of the interviewer's utterance that contains any $q_i \in Q$. In addition, it generates `B/E` if the interviewer begins/ends the current dialogue with that utterance (Table 4.6). Any utterance starting with `Q` is encoded by $\mathcal{E}$ that creates the utterance embedding $v_i \in \mathbb{R}^{1 \times d}$. These embeddings get stacked as the interview goes to create the question matrix $V \in \mathbb{R}^{h \times d}$. If $|Q| < h$, then zero-padding is used to create V (in our case, $h = 16$). Finally, V is stacked with the context matrix C (Sec. 4.3.2), and $(V \oplus C)^T \in \mathbb{R}^{d \times (h+\ell)}$ is multiplied by the attention matrix $A \in \mathbb{R}^{(h+\ell) \times n}$ to create the transpose of the context summary matrix $S \in \mathbb{R}^{n \times d}$.

4.4 Experiments

For our experiments, the encoder and the decoder in BlenderBot 1.0 [51] are used.[8] Three models are developed as follows:

- BB: *Blender Baseline* that takes only up to 128 tokens in U_{i-1} as context.
- SW: BB + *Sliding Window* (Section 4.3.1), taking all tokens in U_{i-1} as context.
- CT: SW + *Context Attention* (Section 4.3.2) + *Question Storing* (Section 4.3.3), taking all tokens in $\{U_{i-k}, .., U_{i-1}\}$ as context.

All models are first trained on `raw` and finetuned on `TRN` in Table 4.1). To assess real-life performance, 10 interviews are conducted per model, where each interview consists of exactly 30 turns. Qualitative analysis is performed on the top-3 most frequently occurring errors as follows:

- Repetitions: how often it repeats topics already covered in the previous utterances.
- Early Ending (EE): implies ending the interview without covering sufficient topics.
- Off Topic (OT): how often it makes utterances that are not relevant to the current topic.

Table 4.5 shows the error analysis results. The repetition rates are significantly reduced as the model gets more advanced. Compared to the baseline, the CT model conducts 3.5 times longer conversations before it attempts to end the interview while generating twice as fewer off-topic utterances, which is very promising. Examples of these error types are provided in Appendix A.3.

[8] There have been updated versions of BlenderBot introduced [67, 54]. However, I chose the first version for our experiments because I found it to be as effective yet much more efficient than the newer versions.

Model	Repetitions (R)	Early Ending (EE)	Off Topic (OT)
BB	30.0	13.3	20.0
SW	16.7	23.3	26.7
CT	**6.7**	**46.7**	**10.0**

Table 4.5: The error analysis of all generation models. R: avg-% of repeated topics, EE: avg-% of the interview conducted before the model attempts to end (higher is better), OT: avg-% of off-topic utterances.

Flag	Utterance
B	Hello, and thank you so much for coming today. uh, Let us start by having you just to state your full legal name and also the school that you are currently attending?
S2	Sure, my name is David, and I go to Shanghai high school in China.
Q	Great. um, yeah, let's talk about leadership experiences. Can you give me some specific examples?
S2	Sure, when I was a junior, I was elected to be the Vice President in the Student Union and I organized many activities to help students to enrich their life on campus.
S1	Okay, great. Thank you so much for sharing that. so, um, tell me a little bit about your leadership style. What do you think you bring to the student union?
S2	Sure, I am very open-minded, and with some of my colleagues, we together established a lot of new programs on top of our existing ones. Many of those programs became very popular.
S1	Ok, great. and so, um, what, uh, what would you say has been the most challenging aspect of being a leader?
	...
Q	Great. uh, um, hmm, let' talk about your courses. What courses did you take in high school and which courses do you like most?
S2	Sure, I took Math, English, PE, Chemistry, Art, and so on. I like Math most.
Q	Okay. so, um, tell me a little bit about your, your interest in mathematics.
	...
E	Okay, great. well, thank you very much for sharing that with me. um, and thank you for taking the time out of your day to interview with us.

Table 4.6: An interview dialogue conducted by our best model (CT in Section 4.4). `S1/S2`: interviewer/interviewee (chatbot/human), `B/E`: beginning/ending utterance (chatbot), `Q`: topical question (chatbot).

4.4.1 Static Evaluation

Following previous work, static evaluation is performed on the CT model, where the input is every batch of k-utterances and prior topics per interview, and its output is compared to the corresponding human response in `TST` (Table 4.1). The average Bleu score is 0.08 and cosine similarity is 0.19, which are low. However, such static evaluation assesses each output independently and obstructs dialogue fluency by artificially inserting human utterances to the model, and thus, does not reveal its capability in conducting long contextualized interviews.

4.4.2 Real-time Evaluation

The CT model is deployed to an online text-based platform in a public cloud. For real-time evaluation, 5 professional interviewers and 10 students are invited to have conversations with our InterviewBot and give ratings from 1 to 5 to indicate their overall satisfactions. The average dialogue duration is 256 seconds. Almost half of the evaluators are satisfied (Scores 4 and 5) and another 40% indicate positive attitude on the coverage of topics and discussions (Score 3), implying that it performs reasonably well for this realistic setting (Table 4.7). The detailed score distribution from professional interviewers and students is shown in the appendix A.4. Overall, with the average score of 3.5, the InterviewBot has shown great potential in applying to practical applications.

Rating Score	**5**	**4**	**3**	**2**	**1**	$\sum$
Evaluator Count	3	4	6	1	1	15

Table 4.7: The rating distribution of the InterviewBot conversations for real-time evaluation. 5: very satisfied, 4: satisfied, 3: neutral, 2: unsatisfied, 1: very unsatisfied.

4.5 Limitations

While the evaluation of the interviewbot has shown its promising application, I would like to summarize its limitations for future improvement. First, the current interviewbot still has the early-ending issue in conversations, in which the ending utterances would be generated after a non-sufficient number of turns. In this case, the conversation would not be able to cover discussions on critical aspects of applicants. Second, although the interviewbot has certain capabilities to follow up on certain topics brought up during a conversation, it is expected to perform deeper discussions on more details. Third, the bot cannot handle name entities, such as people's names, in conversations, although, in most of the conversations, the interviewbot chose not to mention name entities at all. Last, in some conversations, the interviewbot still generates repeated or random ordering of words and punctuations.

4.6 Conclusion

Our InterviewBot is a model-based dialogue system equipped with contextual awareness and topic sensitivity that conducts college admission interviews. Questions covering diverse topics and discussions in extended follow-ups are carried along the conversations, which have been assessed by professional interviewers and student volunteers. The average satisfaction score of 3.5 projects prevailing deployment of the InterviewBot for thousands of college applicants, especially for international students.

With promising future applications, however, the current version of the InterviewBot has two major limitations. First, the early ending in Table 4.5 still happens, where an ending utterance gets generated after an insufficient amount of turns, in which case, the interview may not cover critical aspects of theapplicant. Second, the bot makes good follow-ups to various topics; however, it needs to derive deeper discussions with more details.

Chapter 5

Topical Investigation in Interview Conversations

In this chapter, I process and analyze the interview dataset, extracting utterances that reflect the topics of interest within a few turns, clustering topics into groups that give an insightful presentation of frequent and diverse topics in conversations, and connecting topics to align with the conversation flow.

Interview ID	0AZ4vAaZP6EhgHoWNZa6ZJewp7GhX5QU2unCefUo
Annotation Question	[If you could start high school over], how would you approach it differently?
Question Timestamp	00:00:19.
Original Utterance	Great. Uh, while trending, uh, what will sorry, one, one. Why don't you, I mean, um, great. Uh, so, um, you're a high school senior now, right? Which means that you've had a couple years of high school under your belt already, but, uh, looking back at all your experiences, if you could restart your high school, how would you approach your, your high school differently?
Original Utt Timestamp	00:00:39

Table 5.1: A comparison between the annotated question and the original utterance. There are differences in the timestamps and expressions.

5.1 Interview Topic Data Processing

In the interview dataset, 180,000 key questions in 7,360 conversations are annotated with timestamps by InitialView. However, there are multiple issues upon observation.

Key	Value
start	**Speaker 0:** Hello, and thank you so much for coming today. Uh, why don't we start by having you just to state your full legal name and also the school that you're currently attending? **Speaker 1:** Uh, my <inaudible> launcher me and you can call me Queenie and my high school is she does on number one high school.
If you could start high school over, how would you approach it differently?	**Speaker 0:** Great. Uh, while trending, uh, what will sorry, one, one. Why don't you, I mean, um, great. Uh, so, um, you're a high school senior now, right? Which means that you've had a couple years of high school under your belt already, but, uh, looking back at all your experiences, if you could restart your high school, how would you approach your, your high school differently? **Speaker 1:** Um, maybe I would choose to enter a international high school. ...
What are some things you've done that you can be proud of in the future	**Speaker 0:** Okay. Yeah. Um, so in, in your school now, um, what, what are some things that you've done that you think you can look back and be really proud of in, in the future? **Speaker 1:** Uh, it's uh, my school, uh, it's just that practice so many practice and having a burden of study that makes me more patient and more hardworking. ...
...	...

Table 5.2: Final match of annotated question to the original conversation.

(1) Many are edited with extra information or summarized to complete semantics. (2) The timestamps are offset and even incorrect due to edits of questions and erros. An example of the annotated questions compared to original utterances is shown in Table 5.1. To match the annotation to the original utterances in the conversation, a procedure involving time slot matching, data normalization, fuzzy matching, and SentenceBurt-based embedding cosine similarity was established, followed by a reordering procedure. The reordering procedure helps correct the errors in utterances matching due to utterances similarities as well as speaker ID errors in utterance align-

ment. An example of the final match in the conversations is shown in Table 5.2. In the figure, the opening utterances are placed in a segment with the key ”start” followed by all the annotated questions leading each segment of utterances in the conversation. These leading questions are spoken by interviewers, bringing up inquiries of certain perspectives of applicants. The utterances in the segments are mostly discussions of the key points from leading questions accordingly. In addition, the leading questions reflect the overall topics of interest and how the conversation was developed.

5.2 Interview Data Topic Clustering

Looking at the leading question of each segment, it is clear to observe that questions can be categorized into different topic clusters, as demonstrated in Table 5.3. To initially cluster the questions into different topical categories, I tried multiple algorithms such as k-means-based algorithms, Latent Dirichlet allocation (LDA), language model-based similarities, keyword-based KeyBert, etc. Specifically, LDA and pre-trained language models, respectively, cannot handle sophisticated expressions in prolonged utterances. KeyBert does a decent job, but the extra effort must be spent on the phrase level to map actual utterances and extracted phrases, which introduces errors in the process. Eventually, I used BERtopic to perform the topic clustering task initially. BERTopic[1] is a topic modeling algorithm that leverages transformer-based embedding representation and TF-IDF to group sentences into interpretable clusters. By BERTopic, all questions are split into 2,026 clusters, an example of which is shown in Table 5.4. The cluster titles in the table are manually summarized based on utterances in corresponding segments respectively. Some clusters could further be merged together automatically based on semantics. SentenceBert was used to calculate centroid embedding cosine similarities among clusters and used 0.9 as a threshold to make merges. After the automatic merge, the number of clusters is

[1]https://maartengr.github.io/BERTopic/index.html

Topic Cluster	**utterances**
Topic 1	have you participated in other meaningful activities did you participate in any other activities have you participated in other competitions are there any other activities you participate in ...
Topic 2	when did you graduate did you graduate already when did you graduate from high school did you graduate high school in december ...
Topic 3	was there anything surprising you did this past summer describe anything surprising you did this past summer what was something surprising you did this past summer did anything surprising happen to you this past summer ...
Topic 4	what subjects interest you what other subjects are you hoping to explore what other subjects do you want to explore what subject do you want to pursue ...
Topic 5	what was a highlight of 2019 what was your 2020 highlight was there a highlight what was your highlight moment ...
Topic 6	are you interested in math and physics why are you interested in physics and math why do physics and math interest you are you interested in math and physics ...
...	...

Table 5.3: BERTopic initial clustering result demonstration.

Topics	Utterances
Research Opportunity	what opportunities have you had for research? how did you find this research opportunity? what other research opportunities have you had? how did you get your research opportunity? ...
Critical Thinking	can you teach someone critical thinking? how has critical thinking helped you? how would you define critical thinking? how have you developed critical thinking? ...
Journalism	do you have experience with journalism? do you have journalism experience? when did you first become interested in journalism? are you interested in journalism? ...
...	...

Table 5.4: BERTopic initial clustering result demonstration.

reduced to 1,206. Manual merges were necessarily performed to finalize each cluster semantically. Table 5.5 shows several examples of the final clusters. In the table, each cluster represents a relatively general topic. The popularity of each topic is different in conversations. Table 5.6 shows the top 10 most frequent and unique conversation topics. The statistics in the table show that conversations always tend to cover high school and college academics and activities, which makes sense in college interviews. Other popular topics include extracurricular activities and some academic subjects.

5.3 Topic Flow in Interview Conversations

Identifying the topic flow in conversations through question annotation and clustering is a crucial step in analyzing interview data. Through this process, valuable insights into the topics of interest in interviews are gained regarding how they relate. The examples presented in Table 5.7 provide a clear visualization of how topics are

Topics	Utterances
College Academics/Activities	what subject comes easily to you? which subject excites you most? what problems or programs do you want to work on in college? what are your plans for college? ...
High School Academics	what classes are you taking in high school? what classes did you enjoy taking at your school? what are your higher level courses? what is a favorite course of yours? ...
Highlights	what was a highlight of 2019? what was your 2020 highlight? what was the trip highlight? what was your highlight moment? ...
...	...

Table 5.5: BERTopic final clustering result demonstration.

Topic Cluster	Unique Frequency	Topic Cluster	Unique Frequency
College Academics/Activities	782	High School Academics	575
Computer Science	466	Extracurricular	370
Competition	336	Research	287
Economics	239	Music/Art	227
Psychology	223	Projects	188

Table 5.6: Unique topic frequency in conversations.

connected, allowing us to see the conversation progression from one topic to another.

Moreover, by examining the statistics presented in Table 5.6, we can observe that certain topics appear more frequently than others, indicating their importance to constructing interview conversations. For instance, College Academics/Activities, High School Academics, Computer Science, and Extracurricular activities are some of the most popular topics discussed in interviews, suggesting that these topics are the core of the interview conversations.

Interview ID	Topic Flow
mnfRs5fNQqB MYqknbA-JWAyvRAva rWAreGHabG-MEf	National Holiday -> Practical Application -> Chemistry -> College Academics/Activities -> Chemistry -> Sports -> Academic Project -> Weekend Plan -> Professor Connection -> Responsibility -> Team Contribution -> Miscellaneous -> More to Share -> Chinese New Year Plan -> Sports -> Sports -> End
tJsIIguOM9 k7eouzxTYAR2U QWwh1v8n4uK wNqYg1	Miscellaneous -> College Academics/Activities -> Miscellaneous -> Writing -> Astrophysics -> Astrophysics -> Extracurricular -> Computer Games -> Computer Science -> Team Leading Challenge -> Extracurricular -> News -> College Academics/Activities -> More to Share
1lbNg2XYQBj 3gLTWKJYHp TTWinZTlY9N U7Oboo7k	High School Academics -> Psychology -> Experience Influence -> Leisure Time -> Favorite Place -> Sports -> Extracurricular -> More to Share -> More to Share -> End
vKRX3ubfvp 3bszjZfPy6S FHUKyUfT6N mXMAs7ZXY	School Project -> Charity Work -> High School Academics -> Miscellaneous -> Research -> College Academics/Activities -> High School Academics -> Psychology -> Biology -> College Academics/Activities -> Extracurricular -> Extracurricular -> Music/Art -> Music/Art -> More to Share -> Leadership -> Leadership -> End
FfWGYvyZDb L4UB9rJdSOsb EFYFaZrqZ S6S3nuMCt	College Academics/Activities -> College Academics/Activities -> Computer Science -> Engineering -> School Project -> Practical Application -> Environmental Science -> Computer Science -> Mathematics -> Miscellaneous -> School Project -> More to Share -> End

Table 5.7: Topic Flow in Conversations.

Understanding the conversation topic flow can be immensely valuable in designing generative models that accurately capture the themes and patterns that emerge in interviews. With the incorporation of this information into model designs, it is promising that responses that are more relevant and reflective of the topics of interest can be generated.

Chapter 6

Summary, Discussion, and Future Work

In this chapter, I review the major research objectives that guided my research, the findings, and their potential applications to improve the performance of dialogue systems further.

6.1 Summary, Discussion and Future Work

My research focuses on the effective incorporation of contextual information for dialogue systems. The main research questions that drive my exploration for answers are as follows:

- RQ1: What contextual information could be incorporated into dialogue models?
- RQ2: How can contextual information be effectively incorporated into dialogue models?
- RQ3: What can be effective representations of a conversation flow?

- RQ4: How to extract and utilize the conversation flow into conversational modeling?

To answer these questions, a parametrized model was first built to integrate multiple sources of context and select a response from candidates. Although this model was simple, it improved the satisfaction of the dialogue system in the Alexa Prize. It gives insights into how critical contextual information is in building conversational systems.

Previous state-of-the-art research in retrieval-based dialogue systems built models that only utilize conversation history as contextual representation while overlooking the fact that candidates have associated contextual information. Although this context comes from candidates, it reflects the interaction topic. In my research, a parallel architecture model was built to integrate that particular context. The experiment results showed that integrating this contextual representation significantly improved performance. In addition, it brought up the insight that topic-related information is beneficial to dialogue systems. Furthermore, the parallel model architecture is easily expandable if other context sources must be leveraged into dialogue systems.

Contextual representation in generative models was then studied. State-of-the-art Blenderbot had limitations and difficulty generating consistent, diverse, and enriching interview conversations after finetuning only on conversational history. So first, a general window sliding technique is applied to overcome the input length limitation of Blenderbot and its capability of fusing arbitrary sources of context. Then the key questions are utilized as topical representations to improve the consistency and diversity of the conversation as described in Section 4.3.3. Experiment results showed that the topic-related questions helped improve the overall quality of conversation generation.

From the studies, it is demonstrated that the architectures that are developed can be generalized in both retrieval-based and generative models, respectively. Moreover, the most critical perspective is the insight that topic-related context plays an important role in dialogue generation.

However, there are still limitations in the developed models. Although the contextual representations in the above models are related to topics of interest, they are not abstracted and used individually, missing connections with each other.

Further studies were conducted to study the representation of conversation flow by topic-related context. In Chapter 5, not only were topics clustered to show popularity and diversity in conversations but also connected to represent conversation flow. The multiple tables give invaluable insight that topics are critical elements leading conversation flows.

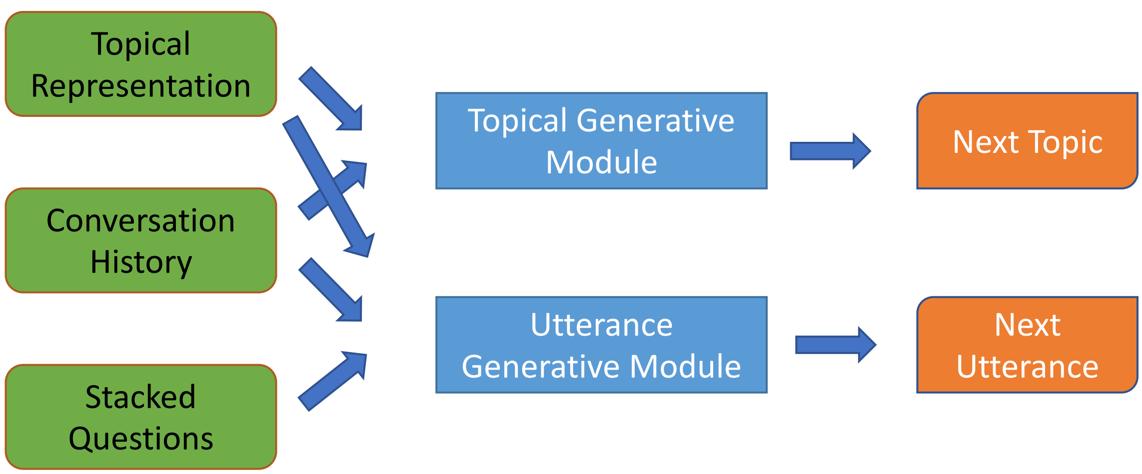

Figure 6.1: Proposition of generative model architecture.

With the Results in Chapter 5, both retrieval-based and generative dialogue systems would benefit from topic flow representations. Topic flow representations can help in capturing the overall theme of the conversation, allowing retrieval-based systems to select more relevant responses. On the other hand, topic flow representations can assist generative systems in maintaining coherence and continuity in the conversation, ensuring that the generated responses are contextually appropriate and coherent. Extending the work in Section 2.4, I would like to prospect future work to improve generative dialogue model performance by utilizing topical context. The current model stacks questions the interviewer (Interviewbot) discussed to avoid repetitions as described in Section 4.3.3. However, it is still not equipped to express the topic flow in conversations explicitly.

There are three directions worth exploring. First, the annotated questions, as the explicit representation of topics, can be connected sequentially to represent the conversation flow. The connection aligns the detailed information in questions with the semantics in adjacent utterances, which could be essential to generate relevant content in the next turn. One potential drawback is that the model may focus too much on the question's details and miss the conversation's broader context. The second proposition would be using clustered topics implicitly. By clustering similar questions, the connections of these clusters in conversations sequentially can identify the underlying structure of the conversation, which would be beneficial to the overall conversation flow. Additionally, this may be more robust to noise in the input data, as it is not overly reliant on specific details. However, this approach may lose some specific information embedded in the utterances. The third method would be combining both approaches in a hybrid way that leverages the strengths of each.

Overall, I can envision that with more effective ways of incorporating enriched contextual information, such as topic information, the conversational dialogue will be generating consistent, natural, and semantically enriching conversations.

6.2 Special Discussion on ChatGPT

The abrupt emergence of ChatGPT is a significant milestone in the development of artificial intelligence and natural language processing.

One of the most significant positive impacts of ChatGPT is its ability to enhance the quality of dialogue systems. ChatGPT has significantly improved the naturalness and coherence of the dialogue system, allowing for more natural and human-like conversations between machines and humans. With its advanced language processing capabilities, ChatGPT can understand complex sentences, idioms, and slang, which makes it easier for humans to communicate with machines.

While it presents new opportunities in the human-computer interactions, new challenges are raised as how to restrain it and adapt it to specific fields with minimal cost. As some research scenarios discussed in this dissertation, they are targeted at specific goals or mixture of multiple goals, but not wide open topics. How can we adapt ChatGPT into these dialogue systems? Can we use the topic flow as investigated in the dissertation to restrict guide ChatGPT to serve the conversation needs specifically.

In conclusion, the emergence of ChatGPT has brought about significant advancements in the field of artificial intelligence and natural language processing. At the same time, it prompts new challenges in these fields as well to further push the frontier forward.

Appendix A

Appendix

A.1 Interviewee Demographics

We summarize the demographics of the interviewees in this section. Table A.1 shows the distribution of the ages of applicants. Most interviewees are between 17 to 19, which is an accurate reflection of the ages of high school students applying to colleges.

The distribution of countries of origin of applicants is shown in Figure A.2. There are 38 countries in total. The majority of applicants come from China. Other major countries are Belgium, Bangladesh, Canada, India and Belarus.

The gender distribution of applicants is shown in Figure A.3. The numbers of male and female applicants are close, with exclusion of applicants not providing gender information.

A.2 Examples of Diarization Errors

The following are examples to illustrate the sources of diarization errors (underlined). In many case, interviewers and interviewees overlap on speeches or thinking out loud with or without filler words, which concatenates the two utterances. A small portion

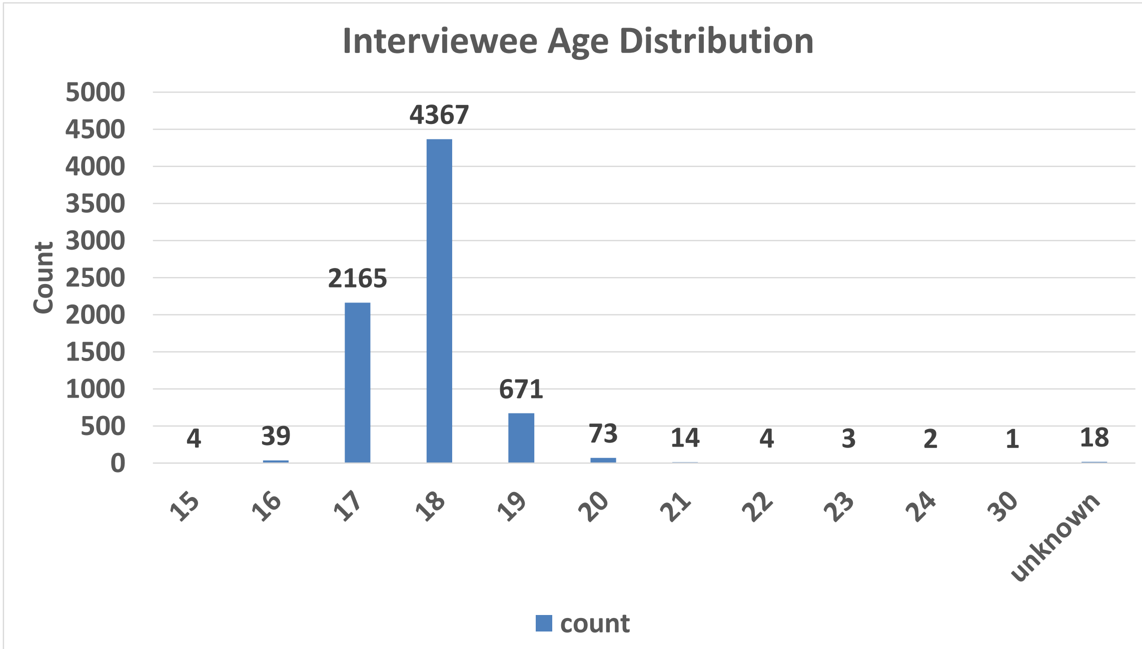

Figure A.1: The interviewee age demographics.

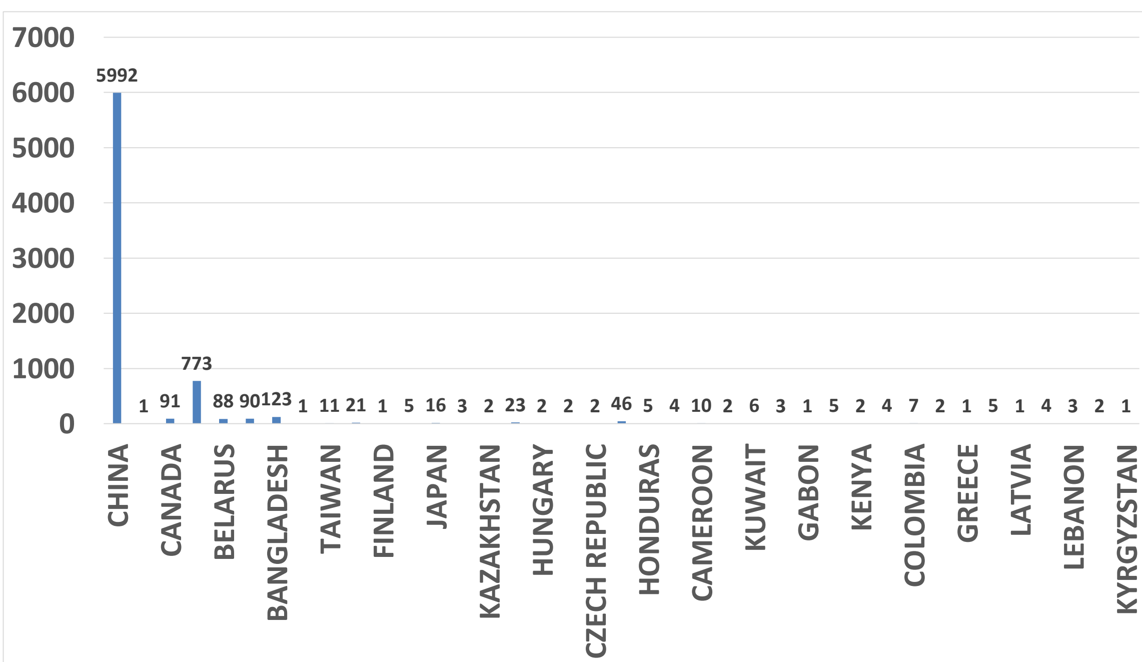

Figure A.2: The interviewee country demographics.

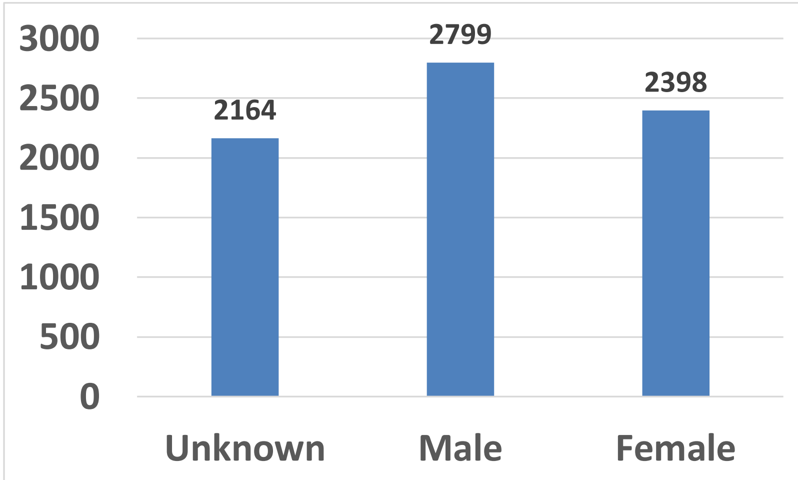

Figure A.3: The interviewee gender demographics.

of diarization errors are from speech recognition and word repetition errors.

- Arbitrary Concatenation
  - What do you think the benefits might be of this kind of technology? If we develop it, I think this technology will eventually replace, um, human delivery.
- Filler Words
  - Oh, no, I'm going to make majoring mathematics. Okay. Okay. Now why, why do you think receiving an education is important?
- Speech Recognition
  - Um, okay. My name is <inaudible>. I'm a senior year student come from Hunger-Free. Which high school are you from?
- Word Repetition
  - I heard it said, so it's kind of like a DIY community community. Are there community activities?

Error Type	Examples
ASR	Um, okay. My name is <inaudible>. I'm a senior year student come from Hunger-Free language school.
WR	I heard it said, so it's kind of like a DIY community community.
FW	Oh, no, I'm going to make majoring mathematics. Okay. Okay. Now why, why do you think receiving an education is important?
AC	What do you think the benefits might be of this kind of technology? If we develop it, I think this technology will eventually replace, um, human delivery.

Table A.1: Error type examples.

We conduct speaker diarization errors by annotating 100 conversations with an average of 39 turns. The type of errors with their stats are shown in Table A.2. The major errors are caused by filler words and arbitrary concatenation. Counts of single errors are counted. Table A.1 listed examples underlined for each type of errors respectively.

Count	Type		
212	Misc	ASR: 147	WR: 65
289	FW		
378	AC		
# of Convs	100		
Avg # of turns	39		

Table A.2: Speaker diarization error analysis in the text transcription from interview recordings. Notions are Misc: Miscellaneous; ASR: Auto Speech Recognition; WR: Word Repetition; FW: Filler Words; AC: Adjacent Concatenation.

From the table, there are different occasions for ASR errors, such as words are not transcribed, words transcribed incorrectly, etc. Word repetitions happen rarely when non-filler words repeat in a sentence. Concatenations of two adjacent utterances take the largest portion of all errors.

A.3 Examples of Generation Limitations

The following are examples for each type of generation errors (underlined). Early ending prevents the Interviewbot from covering critical aspects of applicants and further discussion. The Interviewbot cannot name mentions properly, either skipping over or generating wrong entities in the conversations. Random Generation generates random words and punctuation, and concatenates them incorrectly. Not following up on previous topics hinders the Interviewbot from becoming proactively engaged with interviewees.

- Early Ending
  * InterviewBot: Could you please state your name?
  * Interviewee: Sure. My name is David.
  * ... 4 more turns ...
  * InterviewBot: Thank you and best wishes to your applications.
- Repetition
  * InterviewBot: Now why, why do you think receiving an education is important?
  * ... more turns ...
  * InterviewBot: Why do you think people value education so much?
- Off Topic
  * Interviewee: Sure, I really like Mathemathcs.
  * InterviewBot: What do you do for fun?

A.4 InterviewBot Human Evaluation Distribution

The following table gives the distribution of evaluation scores from invited professional interviewers and students evaluating the InterviewBot.

	5	**4**	**3**	**2**	**1**
Professional Interviewer	1	2	1	1	0
Students	2	2	5	0	1

Table A.3: Human evaluation score distribution from professional interviewers and students.

Bibliography

[1] Daniel Adiwardana, Minh-Thang Luong, David R So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, et al. Towards a human-like open-domain chatbot. arXiv preprint arXiv:2001.09977, 2020.

[2] Ali Ahmadvand, Ingyu Jason Choi, Harshita Sahijwani, Justus Schmidt, Mingyang Sun, Sergey Volokhin, Zihao Wang, and Eugene Agichtein. Emory irisbot: An open-domain conversational bot for personalized information access. Alexa Prize Proceedings, 2018.

[3] Ali Ahmadvand, Harshita Sahijwani, Jason Ingyu Choi, and Eugene Agichtein. Concet: Entity-aware topic classification for open-domain conversational agents. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pages 1371–1380, 2019.

[4] Parham Amiri and Elena Karahanna. Chatbot use cases in the covid-19 public health response. Journal of the American Medical Informatics Association, 29 (5):1000–1010, 2022.

[5] Xavier Anguera, Simon Bozonnet, Nicholas Evans, Corinne Fredouille, Gerald Friedland, and Oriol Vinyals. Speaker diarization: A review of recent research. IEEE Transactions on audio, speech, and language processing, 20(2):356–370, 2012.

[6] Daniel Baier, Alexandra Rese, Maximilian Röglinger, D Baier, A Rese, and M Röglinger. Conversational user interfaces for online shops? a categorization of use cases. In Proceedings of the International Conference on Information Systems, ICIS, 2018.

[7] Adam Berger and John Lafferty. Information retrieval as statistical translation. In ACM SIGIR Forum, volume 51, pages 219–226. ACM New York, NY, USA, 2017.

[8] Dan Bohus and Alexander I Rudnicky. Ravenclaw: Dialog management using hierarchical task decomposition and an expectation agenda. 2003.

[9] Dan Bohus and Alexander I Rudnicky. The ravenclaw dialog management framework: Architecture and systems. Computer Speech & Language, 23(3):332–361, 2009.

[10] Antoine Bordes and Jason Weston. Learning end-to-end goal-oriented dialog. arXiv preprint arXiv:1605.07683, 2016.

[11] Christopher Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Gregory N Hullender. Learning to rank using gradient descent. In Proceedings of the 22nd International Conference on Machine learning (ICML-05), pages 89–96, 2005.

[12] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555, 2014.

[13] Kenneth Mark Colby, Sylvia Weber, and Franklin Dennis Hilf. Artificial paranoia. Artificial Intelligence, 2(1):1–25, 1971.

[14] Gordon V Cormack, Mark D Smucker, and Charles LA Clarke. Efficient and effective spam filtering and re-ranking for large web datasets. Information retrieval, 14(5):441–465, 2011.

[15] Sam Cunningham-Nelson, Wageeh Boles, Luke Trouton, and Emily Margerison. A review of chatbots in education: practical steps forward. In 30th Annual Conference for the Australasian Association for Engineering Education (AAEE 2019): Educators Becoming Agents of Change: Innovate, Integrate, Motivate, pages 299–306. Engineers Australia, 2019.

[16] Jeffrey Dalton, Laura Dietz, and James Allan. Entity query feature expansion using knowledge base links. In Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval, pages 365–374, 2014.

[17] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL `https://www.aclweb.org/anthology/N19-1423`.

[19] Xiangmin Fan, Daren Chao, Zhan Zhang, Dakuo Wang, Xiaohua Li, Feng Tian, et al. Utilization of self-diagnosis health chatbots in real-world settings: case study. Journal of medical Internet research, 23(1):e19928, 2021.

[20] Sarah E Finch, James D Finch, Ali Ahmadvand, Xiangjue Dong, Ruixiang Qi, Harshita Sahijwani, Sergey Volokhin, Zihan Wang, Zihao Wang, Jinho D Choi, et al. Emora: An inquisitive social chatbot who cares for you. arXiv preprint arXiv:2009.04617, 2020.

[21] Fenfei Guo, Angeliki Metallinou, Chandra Khatri, Anirudh Raju, Anu Venkatesh, and Ashwin Ram. Topic-based evaluation for conversational bots, 2018.

[22] Han He, Liyan Xu, and Jinho D. Choi. ELIT: emory language and information toolkit. CoRR, abs/2109.03903, 2021. URL `https://arxiv.org/abs/2109.03903`.

[23] Eduard Hovy, Ulf Hermjakob, Chin-Yew Lin, et al. The use of external knowledge in factoid qa. In TREC, volume 2001, pages 644–52, 2001.

[24] Xiaohua Hu, Xiaodan Zhang, Caimei Lu, Eun K Park, and Xiaohua Zhou. Exploiting wikipedia as external knowledge for document clustering. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 389–396, 2009.

[25] Vladimir Ilievski, Claudiu Musat, Andreea Hossmann, and Michael Baeriswyl. Goal-oriented chatbot dialog management bootstrapping with transfer learning. arXiv preprint arXiv:1802.00500, 2018.

[26] Kristiina Jokinen, Hideki Tanaka, and Akio Yokoo. Context management with topics for spoken dialogue systems. In Proceedings of the 17th international conference on Computational linguistics-Volume 1, pages 631–637. Association for Computational Linguistics, 1998.

[27] Khosrow Kaikhah. Automatic text summarization with neural networks. In 2004

2nd International IEEE Conference on'Intelligent Systems'. Proceedings (IEEE Cat. No. 04EX791), volume 1, pages 40–44. IEEE, 2004.

[28] Bui Thanh Khoa. The impact of chatbots on the relationship between integrated marketing communication and online purchasing behavior in the frontier market. Jurnal The Messenger, 13(1):19–32, 2021.

[29] Soomin Kim, Joonhwan Lee, and Gahgene Gweon. Comparing data from chatbot and web surveys: Effects of platform and conversational style on survey response quality. In Proceedings of the 2019 CHI Conference, pages 1–12, 04 2019. doi: 10.1145/3290605.3300316.

[30] Staffan Larsson and David R Traum. Information state and dialogue management in the trindi dialogue move engine toolkit. Natural language engineering, 6(3-4):323–340, 2000.

[31] J Devlin M Chang K Lee and K Toutanova. Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

[32] Jingyi Li, Michelle X Zhou, Huahai Yang, and Gloria Mark. Confiding in and listening to virtual agents: The effect of personality. In Proceedings of the 22nd International Conference on Intelligent User Interfaces, pages 275–286, 2017.

[33] Jiwei Li, Will Monroe, Tianlin Shi, Alan Ritter, and Dan Jurafsky. Adversarial learning for neural dialogue generation. arXiv preprint arXiv:1701.06547, 2017.

[34] Rongzhong Lian, Min Xie, Fan Wang, Jinhua Peng, and Hua Wu. Learning to select knowledge for response generation in dialog systems. arXiv preprint arXiv:1902.04911, 2019.

[35] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A

robustly optimized BERT pretraining approach. CoRR, abs/1907.11692, 2019. URL http://arxiv.org/abs/1907.11692.

[36] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692, 2019.

[37] Yi Luan, Yangfeng Ji, and Mari Ostendorf. Lstm based conversation models. arXiv preprint arXiv:1603.09457, 2016.

[38] Jon D. Mcauliffe and David M. Blei. Supervised topic models. In J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, editors, Advances in Neural Information Processing Systems 20, pages 121–128. Curran Associates, Inc., 2008. URL http://papers.nips.cc/paper/3328-supervised-topic-models.pdf.

[39] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems, pages 3111–3119, 2013.

[40] Rashid Minhas, Camilla Elphick, and Julia Shaw. Protecting victim and witness statement: examining the effectiveness of a chatbot that uses artificial intelligence and a cognitive interview. AI & SOCIETY, 37(1):265–281, 2022.

[41] Hiroki Nakayama, Takahiro Kubo, Junya Kamura, Yasufumi Taniguchi, and Xu Liang. doccano: Text annotation tool for human, 2018. URL https://github.com/doccano/doccano. Software available from https://github.com/doccano/doccano.

[42] Lin Ni, Chenhao Lu, Niu Liu, and Jiamou Liu. Mandy: Towards a smart primary care chatbot application. In International symposium on knowledge and systems sciences, pages 38–52. Springer, 2017.

[43] Eliza Nichifor, Adrian Trifan, and Elena Mihaela Nechifor. Artificial intelligence in electronic commerce: Basic chatbots and the consumer journey. Amfiteatru Economic, 23(56):87–101, 2021.

[44] Chinedu Wilfred Okonkwo and Abejide Ade-Ibijola. Chatbots applications in education: A systematic review. Computers and Education: Artificial Intelligence, 2:100033, 2021.

[45] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. Text matching as image recognition, 2016.

[46] ME Peters, M Neumann, M Iyyer, M Gardner, C Clark, K Lee, and L Zettlemoyer. Deep contextualized word representations. arxiv 2018. arXiv preprint arXiv:1802.05365, 12, 1802.

[47] Chen Qu, Liu Yang, W. Bruce Croft, Johanne R. Trippas, Yongfeng Zhang, and Minghui Qiu. Analyzing and characterizing user intent in information-seeking conversations. The 41st International ACM SIGIR Conference on Research Development in Information Retrieval, Jun 2018. doi: 10.1145/3209978.3210124. URL `http://dx.doi.org/10.1145/3209978.3210124`.

[48] Charles Rich and Candace L Sidner. Collagen: A collaboration manager for software interface agents. User Modeling and User-Adapted Interaction, 8(3): 315–350, 1998.

[49] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. Okapi at trec-3. Nist Special Publication Sp, 109:109, 1995.

[50] Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M Smith, et al. Recipes for building an open-domain chatbot. arXiv preprint arXiv:2004.13637, 2020.

[51] Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Eric Michael Smith, Y-Lan Boureau, and Jason Weston. Recipes for building an open-domain chatbot. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 300–325, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.24. URL `https://aclanthology.org/2021.eacl-main.24`.

[52] Zeineb Safi, Alaa Abd-Alrazaq, Mohamed Khalifa, Mowafa Househ, et al. Technical aspects of developing chatbots for medical applications: scoping review. Journal of medical Internet research, 22(12):e19127, 2020.

[53] Amnon Shashua and Anat Levin. Ranking with large margin principle: Two approaches. In Advances in neural information processing systems, pages 961–968, 2003.

[54] Kurt Shuster, Jing Xu, Mojtaba Komeili, Da Ju, Eric Michael Smith, Stephen Roller, Megan Ung, Moya Chen, Kushal Arora, Joshua Lane, Morteza Behrooz, William Ngan, Spencer Poff, Naman Goyal, Arthur Szlam, Y-Lan Boureau, Melanie Kambadur, and Jason Weston. Blenderbot 3: a deployed conversational agent that continually learns to responsibly engage, 2022. URL `https://arxiv.org/abs/2208.03188`.

[55] Abubakr Siddig and Andrew Hines. A psychologist chatbot developing experience. In AICS, pages 200–211, 2019.

[56] Eric Michael Smith, Mary Williamson, Kurt Shuster, Jason Weston, and Y-Lan Boureau. Can you put it all together: Evaluating conversational agents' ability to blend skills. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 2021–2030, Online, July 2020. Association for

Computational Linguistics. doi: 10.18653/v1/2020.acl-main.183. URL https://aclanthology.org/2020.acl-main.183.

[57] Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. Dialogue act modeling for automatic tagging and recognition of conversational speech. Computational Linguistics, 26(3):339–374, 2000. URL https://aclanthology.org/J00-3003.

[58] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Advances in neural information processing systems, pages 3104–3112, 2014.

[59] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Advances in neural information processing systems, pages 5998–6008, 2017.

[60] Oriol Vinyals and Quoc Le. A neural conversational model. arXiv preprint arXiv:1506.05869, 2015.

[61] Zihao Wang, Ali Ahmadvand, Jason Ingyu Choi, Payam Karisani, and Eugene Agichtein. Emersonbot: Information-focused conversational ai emory university at the alexa prize 2017 challenge. Proc. Alexa Prize, 2017.

[62] Joseph Weizenbaum. Eliza—a computer program for the study of natural language communication between man and machine. Communications of the ACM, 9(1):36–45, 1966.

[63] Bowen Wu, Baoxun Wang, and Hui Xue. Ranking responses oriented to conversational relevance in chat-bots. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 652–662, 2016.

[64] Qiang Wu, Christopher JC Burges, Krysta M Svore, and Jianfeng Gao. Adapting boosting for information retrieval measures. Information Retrieval, 13(3):254–270, 2010.

[65] Ziang Xiao, Michelle X Zhou, and Wat-Tat Fu. Who should be my teammates: Using a conversational agent to understand individuals and help teaming. In Proceedings of the 24th International Conference on Intelligent User Interfaces, pages 437–447, 2019.

[66] Chenyan Xiong, Zhengzhong Liu, Jamie Callan, and Eduard Hovy. Jointsem: Combining query entity linking and entity based document ranking. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pages 2391–2394, 2017.

[67] Jing Xu, Arthur Szlam, and Jason Weston. Beyond goldfish memory: Long-term open-domain conversation. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 5180–5197, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.356. URL `https://aclanthology.org/2022.acl-long.356`.

[68] Rui Yan, Yiping Song, and Hua Wu. Learning to respond with deep neural networks for retrieval-based human-computer conversation system. In Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, pages 55–64. ACM, 2016.

[69] Liu Yang, Minghui Qiu, Chen Qu, Jiafeng Guo, Yongfeng Zhang, W Bruce Croft, Jun Huang, and Haiqing Chen. Response ranking with deep matching networks and external knowledge in information-seeking conversation systems.

In The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, pages 245–254. ACM, 2018.

[70] Liu Yang, Minghui Qiu, Chen Qu, Cen Chen, Jiafeng Guo, Yongfeng Zhang, W. Bruce Croft, and Haiqing Chen. Iart: Intent-aware response ranking with transformers in information-seeking conversation systems, 2020.

[71] Jui-Feng Yeh, Chen-Hsien Lee, Yi-Shiuan Tan, and Liang-Chih Yu. Topic model allocation of conversational dialogue records by latent dirichlet allocation. In Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific, pages 1–4. IEEE, 2014.

[72] Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. DIALOGPT : Large-scale generative pre-training for conversational response generation. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pages 270–278, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-demos.30. URL https://aclanthology.org/2020.acl-demos.30.

[73] Peixiang Zhong, Chen Zhang, Hao Wang, Yong Liu, and Chunyan Miao. Towards persona-based empathetic conversational models. arXiv preprint arXiv:2004.12316, 2020.

[74] Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. Multi-turn response selection for chatbots with deep attention matching network. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1118–1127, 2018.