FEW-SHOT USER INTENT DETECTION AND RESPONSE SELECTION FOR CONVERSATIONAL DIALOGUE SYSTEM USING DEEP LEARNING

WEI YUAN

A THESIS SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTERS OF COMPUTER SCIENCE

GRADUATE PROGRAM IN ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO

MAY 2023

Abstract

Conversational dialogue systems (CDSs), also known as conversational agents, have made significant development in recent years, driven by advances in natural language processing, machine learning, and artificial intelligence techniques. As a result, CDSs have been implemented across various industries, including education, e-commerce, and customer service in messaging apps, websites, and mobile apps to engage with users through natural language. The primary objective of chatbots is to facilitate communication with people and make numerous repetitious tasks easier for humans. This thesis investigates the application of deep learning methodologies in enterprise CDSs to enhance interpretability, fostering user trust in decision-making processes. The contributions of this thesis include proposing example and description-driven approaches that focus on the semantic similarities between the user input and the intent examples or descriptions in a topological tree for few-shot intent detection in enterprise CDSs. Moreover, this thesis presents a novel Topic-Aware Response Selection (TARS) model to retrieve the most suitable and coherent response from a set of candidates based on contextual information for users in persona-based CDSs.

Acknowledgements

I would like to express my sincere gratitude to my advisor, Professor Aijun An, for her invaluable guidance, unwavering encouragement, and understanding throughout my Master's studies. Her exceptional support has been instrumental in shaping my academic journey. Additionally, I would like to express my sincere gratitude to the committee chair, Professor Uyen Trang Nguyen, and committee member, Jimmy Huang, for their invaluable guidance, comments, and support in revising this thesis.

I would like to express my sincere gratitude to my collaborator, Martin Dimkovski, for his invaluable contribution to our research work and for providing me with the opportunity to intern at Intact. Additionally, I am profoundly grateful for Dr. Zongyang Ma's exceptional skills in successfully overcoming various challenges and ensuring the smooth progress of the project.

I would like to extend my heartfelt appreciation to my parents for their unwavering love, support, continuous encouragement, and profound understanding throughout every stage of my academic journey.

I want to extend my heartfelt thanks to my girlfriend for her endless patience, understanding, and encouragement. Her support and belief in me have been a constant source of motivation for doing the research through my master's studies.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

This chapter serves as an introduction to the thesis, primarily focusing on the background and motivation behind conversational dialogue systems, along with the research objectives. Furthermore, several significant contributions are outlined. Lastly, this chapter presents the thesis organization and provides a list of published and under-review works resulting from this research.

1.1 Background and Motivation

Conversational dialogue systems (CDSs) are software applications developed using natural language processing and artificial intelligence techniques. They are employed in messaging apps, websites, or mobile apps to converse or discuss with the user in natural language. In addition, the primary objective of chatbots is to communicate with people and make numerous repetitious tasks easier for humans. Chatbots are being employed in a variety of industries, including education, e-commerce, and customer service are some of the essential applications where chatbots are being employed.

According to [2], CDSs can be broadly classified into three categories: task-oriented bots, question-answering bots, and chit-chat bots. Among these, task-oriented dialogue systems currently play a significant role in the chatbot domain. These systems assist users in accomplishing specific tasks across different domains, such as booking restaurants, hotels, and flights, providing tourist information, or automating customer support [3]. They achieve this by interpreting users' utterances and accurately predicting their intent. Popular task-oriented dialogue platforms like RASA, Google DialogFlow, and Amazon Alexa are widely adopted across industries. The development of task-oriented dialogue systems has also sparked increased interest in few-shot intent detection tasks. On the other hand, chit-chat CDSs, which prioritize users' personal interests, are also integral components of dialogue systems. By incorporating personas into chit-chat bots, these systems can recognize and acknowledge users' emotions, thereby enhancing user satisfaction. However, accurately selecting the most appropriate response based on users' utterances remains a challenging task in dialogue systems.

1.1.1 Few-Shot Intent Detection

Few-shot intent detection is a significant problem in task-oriented conversational dialogue systems (CDSs), as it involves classifying users' utterances into the correct intent with limited information. Commercial scenarios often face a scarcity of annotated conversation data due to the high cost associated with labeling. Therefore, efficiently utilizing the limited labeled conversational data becomes crucial. In essence, the primary challenge of few-shot intent detection lies in achieving practical intent detection within commercial systems when faced with low-data situations, wherein only a few examples are available for each intent [3]. Furthermore, intent classifiers frequently encounter the need to process multi-domain data,

which poses another critical challenge during the few-shot intent detection task.

1.1.2 Response Selection

Response selection is another fundamental task in persona-based chit-chat CDSs, where a dialogue system communicates with a user based on the chit-chat agent's persona. The goal of the response selection task in dialogue systems is to select the appropriate response r from a pool of candidate responses, given the dialogue context c and persona p. It is challenging for persona-based chit-chat systems to accurately and effectively return responses consistent with the dialogue context and the agent's persona. For retrieval-based chit-chat systems, it is typically necessary to retrieve different topic-aware responses according to the dialogue context with various personas.

1.2 Thesis Objectives

This thesis aims to enhance the performance of CDSs by utilizing deep learning methods, which will be achieved through two primary objectives. The first objective is to develop enterprise CDSs that enhance interpretability, thereby promoting trust in their decision-making process. The second objective is to refine the response selection task in persona-based CDSs by identifying the most appropriate response from a set of candidates, considering the dialogue context and the agent's persona. The following section provides a detailed outline of these two primary objectives.

1.2.1 Objective 1: Few-shot Intent Detection

The task of intent detection plays a crucial role in constructing intelligent dialogue systems equipped with natural language understanding capabilities. The prevalent approach in predicting intents within dialogue systems involves the utilization of one or multiple text classifiers, each of which necessitates extensive labeled training data obtained from subject matter experts or real user examples prior to deployment. From an industry standpoint, acquiring such data proves to be expensive, and effectively allocating limited resources for utterance detection becomes indispensable. To tackle the issue of limited labeled data, we explore the application of few-shot learning for intent detection. Our objective is to develop a method that accurately and efficiently detects user intentions based on their dialogue utterances while effectively utilizing limited resources.

1.2.2 Objective 2: Response Selection

Developing a chitchat dialogue system with persona is challenging. Persona-oriented dialogue systems attempts to understand users' feelings and experiences and select appropriate responses from their perspectives. In such systems, maintaining coherence between response and context is vital, as inconsistent responses can impact the user experience during interactions. To enhance the coherence of persona-based chitchat systems, topic modeling can offer additional knowledge at the topic level to assess the semantic relevance between response and context (or persona) and filter out inappropriate candidate responses. From the topic level perspective, by introducing the explicit topical semantics, in addition to the use of word embeddings might offer better measurement of semantic similarity between context or persona and response. Our goal is to design a deep neural model that explores and leverages topic modeling in persona-based dialogue systems to select persona and topic-aware responses.

1.3 Summary of Contributions

In this thesis, we make significant contributions to the integration of deep learning techniques with conversational dialogue systems (CDSs). We specifically focus on two key issues in CDSs: few-shot intent detection for task-oriented CDSs and response selection tasks for chitchat CDSs. The main highlights of our work can be summarized as follows:

1. We designed a domain-specific topological tree to represent different intents with limited training data. We then trained a language model using contrastive Sentence BERT to compute sentence similarity between user utterances and training data examples. By selecting the nearest neighbor and finding the leaf node of the domain-specific topological tree, we predicted the final intent. In an era where Artificial Intelligence often acts as a black box, understanding how to challenge these black boxes and use external knowledge to provide more solid explainability is important from both academic and business perspectives.

2. We provided evidence suggesting that language models can meet and exceed state-of-the-art results in resolving conversation intents with limited resources, without having to train classifiers. Some off-the-shelf language models can even work reasonably well without fine-tuning for immediate use. Furthermore, we demonstrated the interpretability and malleability benefits of using an optional intent tree, making the approach more feasible.

3. We proposed the Topic-Aware Response Selection (TARS) model, a neural network model that considers the topics of context, persona, and response for the response selection task in a persona-based chitchat system. TARS explores similarities between context and response, as well as between persona and response, at both the word-level

and the topic level. Additionally, multi-layer CNNs aggregate matching features, which are then fed into an MLP architecture to measure the matching score of context-response pairs under a specific persona.

4. The contributions of the response selection task include: i) being the first to consider topics in responses, dialog context, and the persona of the responding agent for selecting topic-aware responses that are consistent with the dialog context and the persona of the responding agent in a persona-based chitchat system; ii) designing a novel deep learning-based response selection model, TARS, and proposing an innovative similarity matching architecture for the model; iii) developing a novel Topic-Aware Attentive Module that makes word-level text representations attend to the topical semantics of the context, persona, or response; iv) demonstrating that the enriched representations of the context, persona, and response in TARS are beneficial, and our model outperforms state-of-the-art models for response selection.

1.4 Thesis Layout

The structure of the thesis is organized as follows:

- **Chapter 2** provides a comprehensive review of the relevant literature on CDSs, which is essential for achieving the research objectives of the thesis.
- **Chapter 3** presents a contrastive learning model that utilizes few-shot learning to detect user utterances in an enterprise CDS, thus enhancing interpretability and building trust in the model's decisions.
- **Chapter 4** proposes the Topic-Aware Response Selection (TARS) model that retrieves

the most coherent responses aligned with both the conversation agent's persona and the user's persona in the chitchat dialogue system.

- **Chapter 5** provides a summary of the conclusions and contributions of this thesis and suggests potential directions for future research.

1.5 Published and Under-review Works

The work presented in this thesis has resulted in the following accepted and under-review publications:

1. Contrastive Fine-tuning on Few-Shot Intent Detection With Topological Intent Tree
   Wei Yuan*, Martin Dimkovski*, and Aijun An. Companion Proceedings of the ACM Web Conference 2023 (WWW'23 Companion)

2. Topic-Aware Response Selection in the Personalized Dialogue System
   Wei Yuan*, Zongyang Ma*, Aijun An, and Jimmy Xiangji Huang. (Under Review)

Chapter 2

Background

This chapter provides an introduction to the background of conversational dialogue systems, with a specific focus on two main aspects: types of dialogue systems and system architectures. The objective is to establish a solid foundational understanding of conversational dialogue systems.

2.1 Introduction

In recent years, there has been considerable attention given to dialogue systems, also referred to as conversational agents, due to their potential to revolutionize human-computer interaction. By enabling users to interact with computers using natural language, these systems make it possible to perform a range of tasks, including information retrieval, customer support, and personal assistance. This section provides a comprehensive literature review aiming to present an overview of the current state of research in dialogue systems. It focuses on key concepts, methodologies, and recent developments. Additionally, we discussed the significant challenges encountered in the design and deployment of these systems.

2.2 Types of Dialogue Systems

In this section, we will present an overview of the different types of dialogue systems. Dialogue systems can be classified based on various characteristics, such as the internal architecture of the system, the nature of the underlying data, and the methods employed to generate responses.

2.2.1 Rule-based & Data-Driven Dialogue Systems

Dialogue systems can be categorized as rule-based or data-driven systems, depending on the approaches used to process user utterances and generate responses.

Rule-based Dialogue Systems

Rule-based systems, also known as symbolic Artificial Intelligence, are designed using a set of predefined rules and patterns. These rules are manually crafted by domain experts, and the system attempts to match users' utterances with the predefined patterns [4] to generate an appropriate response accordingly. Typically, traditional dialogue systems have predominantly been rule-based [5], making them popular and easy to implement in early industry products. Nevertheless, rule-based dialogue systems heavily rely on predetermined rules and patterns, posing challenges when faced with complex scenarios. Moreover, such systems struggle to handle ambiguous or context-dependent user inputs and exhibit difficulties in comprehending natural language.

Data-Driven Dialogue Systems

In contrast, data-driven systems employ machine learning algorithms to extract patterns from extensive datasets that consist of both human-human and human-machine conversations. Moreover, data-driven approaches utilize advanced machine learning techniques to enhance accuracy and scalability in order to accommodate high user demand and diverse conversation topics. However, the performance of data-driven dialogue systems relies on the quality and diversity of the training data.

2.2.2 Retrieval & Generative-based Dialogue Systems

The CDSs can also be categorized based on the methods used to generate responses: retrieval-based dialogue systems and generative dialogue systems.

Retrieval-based Dialogue Systems

Retrieval-based dialogue systems are designed to choose the optimal response from a pre-defined set of candidate responses. These systems utilize techniques like pattern matching, information retrieval, machine learning, or deep learning to rank and select the most suitable response. Although retrieval-based dialogue systems offer responses that are human-generated, consistent, and grammatically correct, they encounter difficulties in providing satisfactory answers to user utterances beyond the scope of their training data.

Generative-based Dialogue Systems

In contrast, generative-based dialogue systems do not depend on a predetermined set of responses. These systems commonly employ machine learning and deep learning techniques trained on extensive corpora, such as sequence-to-sequence models or transformer-based

architectures, to generate responses that are contextually suitable to users' utterances. Such systems have the capacity to generate a wider range of responses that are context-specific, thereby handling open-domain conversational situations. Nevertheless, generative-based dialogue systems have the potential to generate grammatically incorrect or semantically incoherent responses.

2.2.3 Task-oriented & Open Domain Dialogue Systems

Task-oriented Dialogue Systems

Task-oriented dialogue systems, also known as goal-oriented or task-driven systems, are designed to assist users in accomplishing their specific tasks efficiently, minimizing unnecessary conversation. These systems focus on understanding users' intentions and providing relevant information or assistance. For instance, customer service chatbots can provide potential solutions, facilitate refund processes, and address frequently asked questions. Meanwhile, virtual personal assistants aid in scheduling appointments, managing to-do lists, and setting reminders. However, these systems are specifically designed for particular tasks and may encounter difficulties in managing complex or open-ended conversations beyond their defined domain.

Open Domain Dialogue Systems

Open-domain dialogue systems are designed to engage users in natural, human-like conversations that do not have specific goals or tasks. The primary objective of these systems is to entertain, inform, or offer companionship to users, without being limited to particular tasks or domains [6]. While these systems have the capability to generate diverse and creative responses, providing a wide range of information and engaging conversation, they may also

produce inaccurate or nonsensical answers and lack the ability to fully comprehend context. Consequently, this limitation can result in potential misinformation or confusion.

2.2.4 Single-turn & Multi-turn Dialogue Systems

Moreover, CDSs can be categorized according to their ability to engage in conversations of varying complexity, namely single-turn and multi-turn dialogue systems.

Single-turn Dialogue Systems

Single-turn dialogue systems focus on generating responses solely based on users' utterances, while disregarding the conversation history. In this scenario, each interaction between the user and the system can be considered as an independent utterance-response pair. Additionally, single-turn dialogue systems can efficiently provide reasonable responses without considering the conversation history. However, single-turn dialogue systems lack the contextual information and continuity that considering the conversation history provides, which may result in less personalized and coherent interactions.

Multi-turn Dialogue Systems

In contrast, multi-turn dialogue systems utilize the conversation history to generate context-aware, coherent, and personalized responses. These systems employ advanced natural language understanding techniques and memory mechanisms to track the evolving dialogue state, thereby enabling appropriate responses to user utterances. However, these systems may introduce delays in generating responses due to the incorporation of conversation history, resulting in slower interactions for generating personalized responses.

2.2.5 Conclusion

In this section, we have presented a comprehensive overview of the various types of dialogue systems, including their distinguishing characteristics and classification criteria. We examined rule-based and data-driven systems, highlighting their differences in handling user utterances and generating responses. Additionally, we explored the distinctions between retrieval-based and generative dialogue systems and their advantages and limitations. Furthermore, we investigated task-oriented and open-domain dialogue systems, illustrating the differences in their objectives and applications. Finally, we described single-turn and multi-turn dialogue systems, emphasizing the importance of conversation depth and context in generating coherent and contextually appropriate responses.

2.3 Architectures of Dialogue Systems

Dialogue system architectures and approaches encompass the structural design and methods employed for developing conversational agents. These systems can be broadly classified into two main categories: modular and end-to-end. This section will provide an overview of these architectures and the primary approaches used within each category.

2.3.1 Modular Dialogue Systems Architectures

This section illustrates the modular architecture of the dialogue system and describes the main components. The Modular dialogue systems decompose the conversational process into distinct components [7], each responsible for a specific aspect of the interaction. Common modules include natural language understanding (NLU), dialogue management (DM), and natural language generation (NLG).

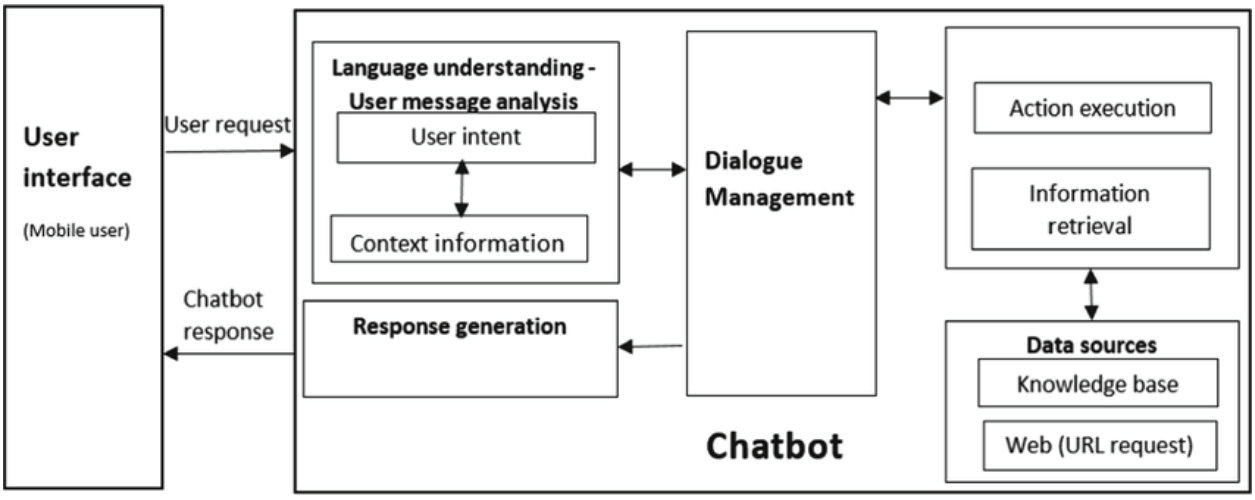

Figure 2.1: General Architectures of Dialogue Systems

User Interface

Initially, the dialogue system activates upon receiving a user request through text or voice input tools, including Facebook, Siri, Google Assistant, Amazon Alexa, WhatsApp, WeChat, or Skype.

User Message Analysis

User message analysis captures the user's request from the user interface controller, analyzing it to determine the user's intent and extracting entities using pattern-matching or other machine learning and deep learning techniques. Furthermore, the user's input sentence can be stored as plain text, thus preserving its grammatical and syntactical information, and subsequently processed utilizing natural language processing (NLP) techniques [8].

Dialog Management

Dialogue management is designed to manage and update the context of the conversation. It identifies and retains the user's intent until the end of the dialogue chat session. It also asks follow-up questions after recognizing the intent [9]. After intent classification in the dialogue management component, the system retrieves information from the back-end and generates appropriate responses for the user.

Response Generation

The response generation component generates responses using one or more of three models: rule-based, retrieval-based, and generative-based.

- **Rule-based** approach selects a response from a set of rules rather than generating new text responses. Knowledge Base structured with conversational patterns is used in rule-based models [10]. However, rule-based response generation is limited to handling specific domain situations.
- **Retrieval-based** approach is more adaptable since it chooses the appropriate response by checking and analyzing accessible resources [11].
- **Generative-based** approach employs natural language generation (NLG) strategies to generate natural language responses to the user, considering recent and previous user input sentences [12].

2.3.2 End-to-End Dialogue Systems

End-to-end dialogue systems leverage data-driven techniques, usually relying on deep neural networks, to learn the entire conversational process without explicitly separating it into

separate modules. These systems are typically trained on large datasets of conversation data.

Sequence-to-Sequence Models

Sequence-to-sequence (Seq2Seq) models [13] use encoder-decoder architectures to generate system responses to user utterances. These models have demonstrated potential in producing diverse, contextually appropriate responses before the BERT model comes out. However, those seq2seq models lack inconsistency and interpretability.

Memory Networks

Memory Networks [14] incorporate external memory to store and retrieve information during conversation. The memory network based-models can facilitate more context-aware and dynamic interactions, used to address some of the limitations of sequence-to-sequence models.

A working memory model (WMM2Seq)[15] introduced a task-oriented system that employs a memory network and it consists of three memory modules: two long-term memory modules storing dialogue history and the knowledge base respectively, and a working memory module that memorizes two distributions and controls the final word prediction. Latterly, [16] trained a task-oriented dialogue system with a “Two-teacher-one-student" framework, aimed at enhancing knowledge retrieval and response quality. In term application of Memory networks for open-domain dialogue systems, [17] proposed a knowledge-grounded chit-chat system, which used a memory network o store query-response pairs, and the generator produced the response conditioned on both the input query and memory pairs.

Attention

Attention mechanisms are also used in end-to-end dialogue systems. [18] leveraged a two-level attention for neural response generation in the dialogue system. Firstly, word-level attention weights are computed, and then sentence-level attention is used to re-scale the weights. Also, [19] proposed a Vocabulary Pyramid Network (VPN) architecture for response generation. This model used an attention-based recurrent architecture with a multi-level encoder-decoder, where the encoder maps raw words, low-level clusters, and high-level clusters into embedded representations, and the decoder uses attention mechanisms to leverage the embedded representations to generate responses.

Transformer

In recent years, transformer-based pre-trained models have become increasingly popular in dialogue systems. For instance, ChatGPT uses the advanced GPT-4.0 model [20], a transformer-based pre-trained model, to understand and generate more coherent, accurate, and contextually relevant responses to users.

Conclusion

In this section, we have provided an overview of the general architectures of dialogue systems, specifically on both modular and end-to-end dialogue systems. Modular dialogue systems offer a structured approach, breaking down the conversational process into distinct components such as natural language understanding, dialogue management, and natural language generation. On the other hand, the end-to-end systems employ data-driven techniques and deep learning models to learn the entire conversational process without explicitly dividing it into separate modules. Additionally, the recent advancements in transformer-based pre-trained models,

attention mechanisms, and memory networks have further improved the performance and capabilities of the end-to-end dialogue systems.

In conclusion, the continuous advancements in the field of dialogue systems hold great promise for the future of human-computer interaction. The development and refinement of both modular and end-to-end architectures, as well as the integration of various techniques and models, will facilitate the creation of more intelligent, context-aware, and engaging conversational agents that can better understand and respond to users' needs.

Chapter 3

Contrastive Fine-tuning on Few-Shot Intent Detection with Topological Intent Tree

This chapter discuss the few-shot intent detection approach which has been deployed in production at Intact Financial Corporation. Our method is example and description-driven, focusing on the semantic similarities between user input and the intent examples or descriptions in a topological tree. This approach does not rely on a generalized classification model. The primary goal was to develop a system capable of accurately identifying intents with as few as one or a few examples per intent. We used large language models (LLMs) to generate embedding representations for the intent resolver, which powers a chatbot at Intact. Since its deployment over a year ago, this chatbot has successfully handled more than 53,000 conversations. The main advantage of our approach is its ability to adapt and learn with minimal training data, making it more flexible and efficient than traditional intent detection

systems. This has proven to be particularly useful in the deployment of our chatbot, which has demonstrated strong performance in real-world applications.

3.1 Introduction

Task-oriented dialogue systems enable users to interact with systems through natural language to accomplish specific tasks or acquire information within a particular domain. Such dialogue systems can be divided into various components, including natural language understanding (NLU), dialogue state tracking (DST), dialogue policy (Policy), and natural language generation (NLG) [21]. Intent detection is a crucial element of the natural language understanding component, and its objective is to accurately map user utterances to the corresponding intents, which then trigger predefined actions or generate appropriate responses

In recent years, the dominant enterprise method for intent detection in dialogue systems is to use a trained text classifiers $f(u)$, which map the user utterance u into an intent in a closed set of K intents $\{I_1, I_2, ...I_K\}$ [22]. Typically, such a classifier requires a substantial amount of training data prior to deployment. This requires collecting sufficient labeled training data from subject matter experts or real user inputs in the form of (u, I), representing user utterances and intents.

The limitations of this dominant method can be illustrated by imagining a new deployment where the intents $\{I_1, I_2, ...I_K\}$ are known with minimal initial information, such as a brief intent description of the intent or a few examples per intent. While this might be adequate for a human to map new user utterances to the intents, a classifier $f(u) \rightarrow \{I_1, I_2, ...I_K\}$ would generally require more labeled data to effectively train the model. Moreover, if a new intent I_{K+1} must be incorporated, the system has to be stoped, and $f(u)$ needs to be retrained to

accommodate the new intent, resulting in $I_1, I_2, \ldots, I_K, I_{K+1}$.

Motivated by the recent achievements in few-shot intent detection tasks, an alternative approach employs a semantic similarity function $Sim(u, I)$ that can assess the similarity between user utterances u and the descriptions or training examples of each intent. Subsequently, the most similar intent can be utilized to determine the user's intent. Pre-trained large language models (LLMs) can be employed to embed user utterances, intent descriptions, and examples in order to compute the similarity without an extensive amount of domain-specific training data. As more data become available, the LLMs can be fine-tuned, and the similarity function can be further optimized. Additionally, modifying or adding intents can be executed online without causing system downtime.

3.2 Related Works

3.2.1 Few-shot Intent Detection

The objective of few-shot intent detection is to evaluate the extent to which practical outcomes in the industry align with recent research findings. We examine trends in few-shot learning and the use of embedding vector spaces as alternatives to intent classifiers. The subsequent section presents a performance comparison between related models and our own.

A collaboration between Carnegie Mellon University and Amazon Alexa AI [23] introduced a BERT-based model trained and fine-tuned on a large-scale, open-domain, multi-turn dialogue corpus, serving as a baseline for the benchmark task-oriented dialogue, DialoGLUE. Moreover, this work proposed task-adaptive training, which fine-tunes BERT to obtain a language model, **CONVBERT**, with over a few hundred million dialogue data, and passes the encoded representation through a linear layer for intent prediction.

Researchers from PolyAI, a leading developer of conversational platforms, assessed sentence embedding [3] and introduced an efficient dual encoder, pre-trained on 727 million (input-response) pairs from the Reddit conversational corpus, to combine dual sentence encoders **USE** [24] and **ConveRT** [25]. The language model was pre-trained on a conversation response selection task and later employed an MLP for intent detection.

In a collaboration between Salesforce and academia, [26] avoided training a classifier model, instead inferring intents using a few-shot nearest-neighbor approach. They employed a BERT-based pairwise encoder to train a binary classifier that outputs a similarity score between a user utterance and a training example, pre-training it with over one million annotated examples for natural language inference (NLI).

In [27], the authors demonstrated an effective few-shot intent detection scheme through contrasting pre-training and fine-tuning on six public datasets covering various user intents.

In [28], the team behind the DialoGLUE benchmark employed a BERT-based encoder without intent classifiers, but their model required the addition of a new "observer" token to their BERT architecture. They extended this work and proposed the **CONVBERT+Obser+Ex** model to improve utterance representation by introducing observers (i.e., tokens not attended to, an alternative to the CLS token used in BERT) and to enhance prediction through example-driven training (where an utterance is classified by measuring similarity to a set of examples corresponding to each intent class). The method was evaluated in both complete data and few-shot training settings.

In [29], PolyAI's researchers devised a straightforward and efficient similarity-based intent detection method, **CONVFIT**, by transforming pre-trained language models into universal sentence encoders and specializing the sentence encoder through contrastive learning. This method involved fine-tuning on task-specific sentence encoders, such as BERT or RoBERTa,

and additional training with 1

Furthermore, [27] proposed **CPFT**, which employs contrastive learning in self-supervised pre-training and supervised fine-tuning steps to achieve state-of-the-art performance. Conversely, authors [26] proposed a data augmentation schema to pre-train a model on a natural language inference (NLI) dataset and then perform transfer learning to predict classes of utterances.

In summary, a trend has emerged in optimizing Transformer models and determining intents by applying distance functions to compute similarity directly with training examples, subsequently identifying the most suitable intent as output. However, the existing literature does not sufficiently explain the extent to which language models have improved over the past year, enabling their use off-the-shelf through simple similarity functions with little or no fine-tuning.

3.2.2 Sentence BERT

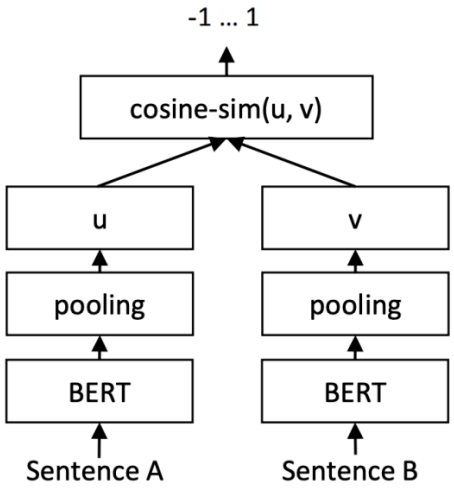

Figure 3.1: Bi-encoder architecture at inference, which used for compute cosine-similarity [1]

Bi-Encoder

To calculate the cosine similarity between two sentences A and B, the Bi-Encoder first inputs each sentence separately into the BERT model and adds a pooling layer to compute the mean of all output vectors using mean squared loss, as illustrated in Figure 3.1. This process generates meaningful semantic sentence embeddings u and v. The cosine similarity between the two sentences is then represented as d-dimensional sentence embeddings. Consequently, the encoding process for sentences x and y can be denoted as:

$$\mathbf{u} = \text{Bi} - \text{encoder}(x) \in \mathbb{R}^d$$

$$\mathbf{v} = \text{Bi} - \text{encoder}(y) \in \mathbb{R}^d$$

$$D(\mathbf{u}, \mathbf{v}) = 1 - \text{cosine-sim}(\mathbf{u}, \mathbf{v}) = 1 - \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\|\|\mathbf{v}\|}$$

In our methodology, we employ a bi-encoder to train and fine-tune the labeled corpus obtained from the cross-encoder, calculate the cosine similarity between the user utterance and each training example within a given intent, and subsequently choose the intent with the highest similarity as the output. Let the user utterance be denoted as $\mathbf{u}$ and the training examples as $\mathbf{x}_{i,n,M}$; then

$$I(\mathbf{u}) = \text{Intent}\left(\arg\min D\left(\mathbf{u}, \mathbf{x}_{i,n,M}\right)\right)$$

Cross-Encoder

In contrast to the bi-encoder, the cross-encoder [1] does not produce semantically meaningful sentence embeddings. However, it can generate cosine similarity between two sentences with greater accuracy. The cross-encoder inputs a distinct sentence pair $[[\text{CLS}], \mathbf{u}, [\text{SEP}], \mathbf{v}, [\text{SEP}]]$,

in comparison to the bi-encoder.

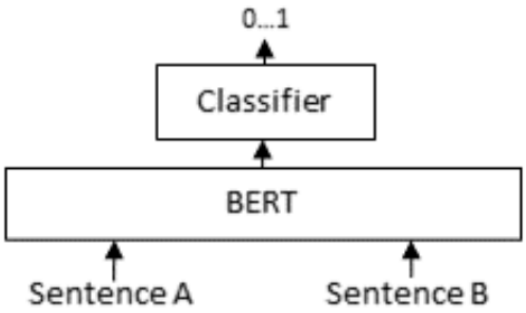

Figure 3.2: Cross-encoder architecture [1]

3.2.3 Contrastive Learning

In [30], contrastive learning was first introduced, and since then, it has been extensively applied in Computer Vision [31]. Recently, an increasing number of research studies in natural language processing have employed contrastive learning. In [32, 33], models incorporating contrastive loss were proposed to achieve superior embedding-based sentence representations. For many-to-many machine translation, [34] developed the mRASP2 model with contrastive loss, minimizing the discrepancy in representations across different languages. In [35], a contrastive framework was introduced for document summarization.

3.2.4 Incrementally Learning for New Intent

In addition to managing new intents, the few-shot text classification model, **HYBRID**, was introduced in [36]. This model incrementally learns new classes in a round-by-round manner, eliminating the need for retraining on previously encountered examples.

3.3 Proposed Methodology

3.3.1 Intent Tree

The system uses a custom dialogue state machine which can prompt the user for clarification if the required information is missing. The state machine guides the dialogue along possible paths in an intent topological tree structure, where each leaf node is a final or more specific intent and each non-leaf node represents a superset of all final intents beneath it. Figure 4.1 shows an extract from an intent tree.

The text content of a leaf node is a set of actual utterance examples associated with that final intent in the dataset. For example, *"How to reset or change my Office 365 password"* is an example of leaf node *"Reset password"*. The text content of a non-leaf node is additional information describing that superset and is created by knowledge base administrators. For instance, the internal node *"login problems"* contains the description of *"can't connect with password"*. It summarises the general content of all its descendants' final intents, and it might or might not contain any information used in the final intents examples.

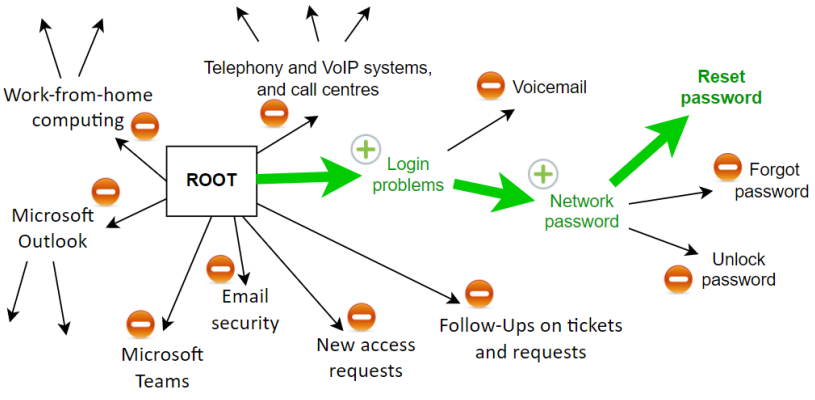

Figure 3.3: Part of the intent tree related to the "reset password"

For each description or example in a tree node, we pre-compute their embeddings using an

LLM, and store them in the system, which speeds up the intent detection process since we do not need to compute these embeddings in the inference step. In addition, our system allows any number of training examples linked to a final intent leaf node and multiple descriptions contained by an internal description node. All nodes in production have less than 8 examples, with the average being 2.6, i.e., the system is few-shot at most and often only one-shot in terms of the number of training examples needed at each leaf.

3.3.2 Branching Decision Making

Given an LLM with an embedding dimensionality d, a user input u, and M branching options represented by the text strings $b_1, b_2, ...b_M$, the dialogue engine does the following: (i) calls the LLM to embed the text u into an $LLM(u) \in \mathbb{R}^d$ vector; (ii) it then retrieves the $LLM(b_1), LLM(b_2), ...LLM(b_M)$ vectors with the dimension of d that are pre-computed and updated when the knowledge base is edited; (iii) makes a branching choice as

$$\arg\max_{b_x} Sim(LLM(u), LLM(b_x))$$

where the Sim represents a similarity function (cosine similarity in our dialogue system). In summary, each branching decision is a nearest-neighbor calculation over the embedded vectors. Repeating this process at each successive branching point shows a path in the intent tree ending with a leaf node.

The user is not necessarily aware of all the branching events, i.e., the dialogue turns. If the user input contains enough information for a confident decision at every internal node along a path to a leaf node, the dialogue engine returns the final action or answer in one turn. If the engine is not confident at any internal node, it queries the user for additional

information before retrying to find a path to a leaf node.

Our design uses the intent tree for interpretability. It allows domain experts to organize knowledge by organizing answers under any visually meaningful hierarchy. The tree also offers interpretability to end-users. If the dialogue engine fails in a final answer, the user can see the assumptions made by the chatbot, i.e., the branching decisions along the tree path. The engine allows the user to backtrack and instruct the bot to correct assumptions before retrying.

Our system uses SBERT[1] [1] as the LLM. The first chatbot version was deployed in production without fine-tuning SBERT because the publicly available general-purpose SBERT worked well enough for the first attempt. As more domain-specific training data become available, SBERT was fine-tuned for better performance. In addition, we can continually expand the intent tree and learn new intents without re-training.

3.3.3 LLM Fine-tuning with contrastive learning

Due to limited labelled training examples, we use contrastive learning to fine-tune the language model. The intent tree is used to automatically generate pairs of similar or dissimilar examples/descriptions for contrastive learning. Starting from a leaf node, i.e., a final intent, positive/similar pairs are created by pairing its examples with the superset description of its parent, then with its grandparent, and so on, up until the root. In summary, this creates positive associations from the root toward the leaf node. In terms of negative pairs, they were recursively created by pairing examples of a leaf node with superset descriptions of siblings of its parent, then with the siblings of its grandparent. These negative associations nudge the intent inference along the correct ancestry path. The contrastive loss function used

[1] `https://github.com/UKPLab/sentence-transformers`

in fine-tuning is as follows [37]:

$$L_{CR}\left(x_i, x_j, y\right) = \frac{1}{2} y D(x_i, x_j)^2 + \frac{1}{2}(1-y)\left\{\max\left(0, m - D(x_i, x_j)\right\}^2\right.$$

where x_i and x_j are the LLM embeddings for utterances u_i and u_j, respectively, and y is the pair label, where $y = 1$ indicates u_i and u_j are a positive pair, and $y = 0$ indicates they are negative. D is the cosine distance metric between x_i and x_j (i.e $D(x_i, x_j) = 1 - \cos\_\text{sim}(x_i, x_j)$), and m is a hyperparameter of margin value. The contrastive learning loss function minimizes the embedding distance of positive/similar pairs and maximizes the distance of negative pairs.

3.3.4 Contrastive Learning with Supervised SimCSE

Our approach employs contrastive learning to transform user utterances into semantic understanding, effectively concentrating on learning by drawing semantically similar representations closer and distancing dissimilar ones [38] [32].

Moreover, we employ another contrastive learning method, contrastive learning LLM Supervised SimCSE model [32], a simple framework for learning sentence embeddings. It leverages self-supervised learning and does not require negative examples during training. The SimCSE method creates positive pairs by applying data augmentation techniques, such as dropout masks, to the same input sentence. The training objective can be described as follows:

$$\ell_i = -\log \frac{e^{\text{sim}\left(\mathbf{h}_i, \mathbf{h}_i^+\right)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}\left(\mathbf{h}_i, \mathbf{h}_j^+\right)/\tau}}$$

Where $\mathbf{h}_i$ and $\mathbf{h}_i^+$are representations of training examples $\mathbf{x}_i$ and $\mathbf{x}_i^+$respectively, and training examples $\mathbf{x}_i$ and $\mathbf{x}_i^+$are comes from same intent, such that $\left(\mathbf{x}_i, \mathbf{x}_i^+\right) \sim p_{\text{pos}}$ Following same

setting for SimCSE, the approach continue using standard dropout as minimal form of data augmentation, thus the positive pairs will different in drop masks [32].

Subsequently, we train supervised SimCSE without a pre-training model on a task-specific dataset containing only positive sentence pairs, where a positive sentence pair consists of sentences from the same intent, including every sentence pair. After training and fine-tuning the SimCSE model, contrastive learning enables it to draw training examples of the same intent closer and those of different intents farther apart. We then compute the average sentence embedding for each intent and classify user utterances based on the maximum cosine similarity between the utterance and the average sentence embedding.

3.4 Experiments

3.4.1 Variations of Our Method

We first evaluate our method with publicly available pre-trained SBERT [1] without fine-tuning. Then we experiment with fine-tuning SBERT with domain-specific knowledge as described in Section 3.3.3 to observe the performance changes. We vary some design choices regarding the LLM and the similarity function. In one variation, we calculate the similarity between the user utterance and each example of the final intents under consideration. Then we use the intent of the nearest neighbor as the intent.

Two distinct language models (LLMs), SBERT and SimCSE, were selected for encoding the utterances and obtaining the sentence embeddings in the experiment. Initially, when we began this project, SBERT was considered the state-of-the-art (SOTA) sentence embedding method. Later, SimCSE emerged, introducing a contrastive learning strategy for sentence embedding. SBERT facilitates contrastive learning due to its siamese network structure. We

Dataset	Intents	Examples	Domains
Intact	33	188	1
HWU64	64	25,716	21
CLINC150	150	23,700	10
BANKING77	77	13,083	1

Table 3.1: Statistics of evaluation datasets

compared the performance of the two different LLMs (SBERT and SimCSE) with contrastive learning.

When we use SBERT as LLM, we denote this variation as SBERT+NN. Another variation calculates the average of all the embeddings in a final intent to obtain a single-vector representation of the intent. Then we compare a user input to this average instead of the multiple calculations required with the nearest neighbor method. We will refer to this as SBERT+AveSE. In addition, we replace SBERT as an LLM with SimCSE[2] [39][3], giving us SimCSE+NN and SimCSE+AveSE. SBERT and SimCSE use different datasets and model designs for their pre-trained LLMs.

3.4.2 Datasets

We use the Intact's helpdesk dataset and three public datasets as described in Table 3.1. The public datasets are taken from the DialoGLUE benchmark [23]: CLINC150 [40], BANKING77 [3], and HWU64 [41]. Each dataset has a fixed split for training and testing data. For Intact's dataset, 101 examples are used for testing and 87 examples for training and fine-tuning. For the public datasets, we use the same split as the previous methods. We follow the benchmark guidelines for 5-shot or 10-shot training and use 5 or 10 examples per intent, respectively.

[2] `https://github.com/princeton-nlp/SimCSE`

[3] The contrastive loss function used in SimCSE is $L_{CR} = -\log \frac{e^{\text{sim}\left(\mathbf{h}_i, \mathbf{h}_i^+\right)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}\left(\mathbf{h}_i, \mathbf{h}_j^+\right)/\tau}}$, where h and h^+ are representations of positive pair x and x^+, τ is temperature hyper-parameter, and N is number of pairs in a mini-batch.

The 188 intent examples listed for Intact's dataset in Table 3.1 are all associated with final intents, i.e., leaf nodes in Intact's intent tree. The tree also has 6 non-leaf nodes containing 21 superset descriptions.

For Intact's dataset, we followed Section 3.3.3 to generate positive and negative pairs for fine-tuning. Since the public datasets do not have intent trees, positive pairs are generated by enumerating pairs of examples from the examples in the same intent, and negative pairs with intent examples from different intents.

3.4.3 Experimental Settings

For SBERT, we use a pre-trained model *all-mpnet-base-v2* and the learning rate is its default value of 2e-05. Batch sizes were 128 for CLINC150, HWU64, and BANKING77 datasets, and 64 for Intact's dataset due to its small size. The number of epochs is 20 for three public datasets, and 7 for Intact's dataset. For SimCSE, we start from its *sup-simcse-roberta-large* model and use its default learning rate of 5e-05, a batch size of 32 and its default temperature (τ) value of 0.05. For the contrastive loss function with SBERT, we use the default value for hyperparameter margin (m), which is 0.5. We compare with 6 state-of-the-art baseline models discussed in Section 3.2.

3.4.4 Results on Intact's Data

Table 3.2 shows the results of Intact's test data with different LLMs (SBERT or SimCSE), and different similarity-computing approaches (NN or AveSE), with or without fine-tuning of LLMs. On this dataset, we compare with only two SOTA methods: CONVBERT+Ex+Obs [28] and DNNC [26] due to source code availability. We used their default settings of 100 epochs and 10 epochs, respectively. All intent examples in the training set are used to train

Method	Accuracy
CONVBERT+Ex+Obs.	68.69
DNNC	63.84
SBERT+NN w/o fine-tuning	86.14
SBERT+AveSE w/o fine-tuning	86.14
SimCSE+NN w/o fine-tuning	45.45
SimCSE+AveSE w/o fine-tuning	59.60
SBERT+NN after fine-tuning	**89.14**
SBERT+AveSE after fine-tuning	84.16
SimCSE+NN after fine-tuning	61.62
SimCSE+AveSE after fine-tuning	60.61

Table 3.2: Accuracy (%) on the Intact's dataset

the models since they range only from 1 to 8 examples per intent.

Table 3.2 shows that our method with SBERT significantly outperforms the two SOTA methods on Intact's dataset with and without fine-tuning. After fine-tuning, SBERT+NN's performance is further improved. SimCSE appears to have limited performance due to its pre-training on a smaller and less diverse dataset collection than SBERT. SimCSE does not improve much after fine-tuning on Intact's dataset, likely because Intact's dataset is small.

3.4.5 Results on Public Datasets

Table 3.3 compares our methods with 6 SOTA methods on the public benchmark datasets. The results show that our methods outperform the SOTA methods on these datasets with fine-tuning. It can be seen that with fine-tuning on large datasets, SimCSE's performance catches up to SBERT's, compensating for its poorer pre-training. It is also interesting to see that the SimCSE without fine-tuning achieves better performance with AveSE than with NN probably because using the average embedding among the examples in an intent is more noise-tolerate than the NN method. After fine-tuning SimCSE, AveSE and NN perform

Models	**CLINC150**		**BANKING77**		**HWU64**	
	5-shot	10-shot	5-shot	10-shot	5-shot	10-shot
USE+CONVERT [3]	90.49	93.26	77.75	85.19	80.01	85.83
CONVBERT [23]	-	92.75	-	83.99	-	84.52
CONVBERT+Ex+Obser[28]	-	93.97	-	85.95	-	86.28
DNNC [26]	91.02	93.76	80.40	86.71	80.46	84.72
CPFT [27]	92.34	94.18	80.86	87.20	82.03	87.13
ConvFit [29]	-	92.89	-	87.38	-	85.32
SBERT + NN without fine-tuning	81.87	84.82	78.41	85.36	69.89	75.46
SBERT + AveSE without fine-tuning	75.16	90.93	80.58	84.55	75.37	81.13
SimCSE + NN without fine-tuning	63.33	63.38	42.82	48.25	52.97	55.20
SimCSE + AveSE without fine-tuning	85.38	88.44	71.85	77.37	76.86	79.09
SBERT + NN after fine-tuning	92.11	93.60	**83.73**	**88.12**	81.60	**87.27**
SBERT + AveSE after fine-tuning	92.11	93.58	**82.53**	**87.70**	81.88	**87.27**
SimCSE + NN after fine-tuning	**92.36**	**94.51**	**82.11**	**87.73**	**82.71**	**87.27**
SimCSE + AveSE after fine-tuning	**92.49**	**94.60**	**81.98**	**87.69**	**82.34**	**87.17**

Table 3.3: Accuracy (%) on benchmark datasets. Baselines results were taken from [27]. Results in bold are better than the baselines.

similarly. Table 3.2 and 3.3 show that the SBERT is not sensitive to the similarity-computing method. Its performance with NN is similar to that of AveSE.

However, performance varies across different datasets. The CLINC150 dataset encompasses 10 domains, whereas HWU64 comprises 21 distinct domains, and BANKING77 exclusively focuses on the banking domain. Within a single domain, distinguishing between intents presents a challenge, resulting in the lowest performance on the BANKING77 dataset. As the number of domains increases, the differences among utterances become substantial and more discernible.

Method	n_1	n_2	n_3	n_4	Overall
HYBRID	50.00	63.16	40.74	25.00	50.49
SBERT+NN w/o fine-tuning	92.86	79.85	**88.89**	**87.50**	86.14
SBERT+AveSE w/o fine-tuning	82.14	**86.84**	**88.89**	**87.50**	86.14
SBERT+NN after fine-tuning	**96.43**	**86.84**	**88.89**	**87.50**	**89.11**
SBERT+AveSE after fine-tuning	89.28	**86.84**	85.16	**87.50**	**89.11**

Table 3.4: Incremental learning Acc(%) on the Intact's dataset

3.4.6 Results on Incremental Learning

To see how well our method handles new intents, we randomly split 33 intents in the Intact dataset into 4 different groups n_1, n_2, n_3 and n_4 with 10, 10, 10 and 3 intents, respectively. We experiment with our method with or without fine-tuning, and compare it with HYBRID [36], a SOTA incremental few-shot intent detection method. The training examples used in fine-tuning are based on seen intents with the same strategy described in Section 3.3.3, and fine-tuning is conducted on the LLM from the previous round.

Table 3.4 shows that our methods with or without fine-tuning do not lose overall performance when adding new classes online, and their performance is significantly better than HYBRID.

3.5 Summary

In this chapter, we presented a few-shot intent detection method that uses an intent tree to help resolve the user intent in a deployed dialogue system. The method is based on similarities between LLM embeddings of the user utterance and the descriptions/examples in the intent tree. Our results show that the off-the-shelf LLMs can work reasonably well without fine-tuning and are fine-tuned well when more training examples become available.

Furthermore, the topological intent tree with the proposed approach can be used for all branching decisions and provides more interpretability. Finally, the incremental few-shot intent detection experiments show that our method can easily handle new intents with or without fine-tuning.

Chapter 4

Topic-Aware Response Selection For Persona-based Dialogue Systems

This chapter presents a novel Topic-Aware Response Selection (TARS) model, which captures multi-grained matching between the dialogue context and a response, as well as between the persona and a response, at both the word and topic levels to select an appropriate, topic-aware response from a pool of response candidates. Empirical results using public persona-based empathetic conversation (PEC) data demonstrate the TARS model's promising performance for response selection.

4.1 Introduction

Persona is defined as the social role or the public image of one's personality.[4] A persona-based chitchat system communicates with a user based on the persona of the chitchat agent. As reported in [42, 43, 44, 45, 46], it is important to maintain a consistent persona of the chitchat

[4]https://en.wikipedia.org/wiki/Persona

agent in order to interact with users more naturally. A persona usually has some dominant topics. For example, a persona with the doctor role might have topics of *medicine* and *music*, relevant to the profession and hobbies of the persona. Agents with different personas may respond from different perspectives given the same situation or context, based on their knowledge, experience, and understanding of the user's feelings. Take the dialog context *"I can not sleep well for a month"* as an example; the agent with the doctor role probably replies with, *"That's bad news. Mind your health and take sleeping pills"*, while the agent with the psychologist role might reply with *"I am sorry to hear that. You should take a rest to ease your anxiety"*. The objective of developing a persona-based chitchat system is to simulate human agents with different personas so that when a user communicates with an agent, the agent can provide consistent responses that relevant to both the knowledge and role of the agent, and the user's needs or feelings.

There are two different types of persona-based chitchat systems: retrieval-based and generation-based chitchat system. A retrieval-based chitchat system selects the most appropriate response from a set of candidates according to the dialog context and the persona of the agent. Research in response selection for the retrieval-based dialogue systems falls into two categories: single-turn [47, 48] and multi-turn [49, 50, 51, 52, 53, 54, 55]. The chitchat systems that fuse the persona features are previously studied in [55, 56, 57, 58]. However, the previous studies [57, 58] only incorporate the persona feature as the sentence embedding vector into the models. The authors in [58] proposed the IMPChat model, following a representation-matching-aggregation pipeline. In [57], the CoBERT model was presented for the response selection task, and it adopts 1-hop and 2-hop co-attentions between the contexts or personas and responses. In both studies, the persona is extracted from the historical utterances of the user. However, the previous studies do not consider the coherence and consistency between the

context and response, which may lead to a response inconsistent with the persona of the agent. It might hurt the user's experience when interacting with the chitchat system. To improve the coherence of the persona-based chitchat system, topic modeling can provide additional knowledge at the topic level to evaluate the semantic relevance between the response and context (or persona), and filter out inappropriate candidate responses. Topics have been utilized in personalized web search models [59] and recommendation systems [60]. Also, topics of utterances have been considered in the chitchat system [61, 62, 63, 64]. However, topics of personas have not been explicitly explored and applied to the persona-based chitchat system to retrieve topic-aware responses. Introducing the explicit topical semantics, in addition to the use of word embeddings, might offer a better measurement of semantic similarity between context or persona and response.

4.2 Problem Definition

The goal of the response selection task in the dialogue system is to select the appropriate response r from the pool of candidate responses given the dialog context c and persona p. More specifically, a training example is formulated as a quadruple $< c, p, r, y >$, where p denotes persona, consisting of historical utterances posted by a user with persona p, and y is the binary value indicating whether or not r is an appropriate reply in context c according to persona p. Our objective is to train a model $\mathcal{F}(c, p, r)$ to predict y', where $y' \in [0, 1]$. We rank the candidate responses by score y' and select the response based on the ranking.

4.3 Related Work

4.3.1 Response Selection for Retrieval-based Dialogue System

In the retrieval-based dialogue system, The goal of response selection is to select an appropriate response from a list of candidate responses. And the researches on response selection for retrieval-based dialogue system fall into two categories: single-turn [47, 48] and multi-turn [49, 50, 51, 52, 53, 54, 55]. For the single-turn task, a representative convolutional neural model was proposed in [47]. In their study, context and response are reformatted as word embedding matrices, where each row of the matrix denotes the word embedding of the sentence. A similarity matching matrix is calculated based on the two matrices. The model treats the similarity matching matrix as the "image", and the methods for Computer Vision (e.g., convolutional neural network, max/mean pooling) are applied to the “image". [48] introduced a new Sina Weibo dataset for single-turn response selection. For the multi-turn task, the most commonly used dataset (i.e., Ubuntu Dialogue Corpus) was proposed in [49]. To catch the similarity between the response and each utterance of the context, the end-to-end retrieval framework generally includes three components: representation, matching, and aggregation. [53] proposed a multi-hop selector network (MSN) to select informative utterances from context and calculate the similarity matrices between these selected utterances and the response. [50] constructed the multi-level self and cross representations with the stacked Transformer blocks. In [52], to further capture the signal of interaction between context and response, an interaction-over-interaction network (IoI) was proposed recently.

4.3.2 Persona, Empathy, and Topics

A dialogue system with personality enables the generation of more personalized responses. The early work for Persona-based Chatbot is studied in [56], modeling the speaking styles and background facts of the speaker. PERSONA-CHAT [65] and PEC [57] are the common datasets used for Persona Chatbot. The former dataset was generated and annotated by crowd-source workers, where the persona is constructed by the sentences from the user profile. The latter dataset was extracted from Reddit, where the persona is built with the historical utterances spoken by the user. Recently, [55] proposed the Dual Interactive Matching Network (DIM), which adopts the dual matching strategy to obtain the deep matching between contexts and responses, as well as personas and responses. In [57], CoBERT model was proposed for the response selection task. CoBERT adopts not only 1-hop co-attentions between the contexts or personas and responses, but also the 2-hop co-attentions between them. [58] attempted to learn implicit user profiles from their historical utterances. The user profile consists with personalized language style (i.e., user's historical responses) and personalized preferences (user's historical posts).

For the context-response pair, an appropriate response tends to have coherent topics with its context. [61] incorporated topic information into their topic-aware convolutional neural tensor network (TACNTN). Topic vectors of contexts and responses were learned by TwitterLDA. [62] presented a framework for evaluating context-response topical coherence by integrating Coherence-Pivoted Latent Dirichlet Allocation (cpLDA) and Multi-Hierarchical Coherence Network (MHCN).

4.3.3 Topic representation.

Topic modeling[66] is a Bayesian method aiming at extracting hidden topics from a collection of documents. We represent topic t using its top n words $[(w_1, p_1^w), ..., (w_n, p_n^w)]$, where w_i represents the top i-th word and p_i^w represents its corresponding probability. Each w_i can be reformatted as a word embedding vector e_i via BERT. Therefore, the topic t can be expressed as $\sum_{i=1}^{n} p_i^w e_i$, which is the **embedding vector of topic** t. Take the context c as an example, $[(t_c^1, p_{c,t}^1), (t_c^2, p_{c,t}^2), ..., (t_c^K, p_{c,t}^K)]$ is generated by topic modeling, where t_c^k is the embedding vector of k-th topic in context c and $p_{c,t}^k$ is the probability of the k-th topic in context c.

4.3.4 Attentive Module.

The attentive module is derived from Transformer [67], which has a superior ability in utterance representation. Given query matrix $Q \in \mathbb{R}^{n_q \times d_q}$, key matrix $K \in \mathbb{R}^{n_k \times d_k}$ and value matrix $V \in \mathbb{R}^{n_v \times d_v}$, the attentive module employs an attention function

$$Att(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_q}})V$$

where d_q is the query dimension. The attentive module composites an attention layer and a feed-forward layer. Also, it adopts residual connection and layer normalization. We denote the Attentive Module as $f_{Attention}(Q, K, V)$, expressed as followed:

$$\begin{aligned} X_1 &= f_{norm}(\text{Att}(Q, K, V) + Q) \\ X_2 &= \text{ReLU}(X_1 W_1 + b_1) W_2 + b_2 \\ f_{Attention}(Q, K, V) &= f_{norm}(X_2 + X_1) \end{aligned} \tag{4.1}$$

where W_1, b_1, W_2 and b_2 are learnable parameters, and f_{norm} is the normalization function.

4.4 Proposed Methodology

4.4.1 New Topic-Aware Attentive Module

Inspired by the structure of the attentive module [67], we propose a new attentive module in a topic-aware fashion to encode contexts, responses, and persona. Taking context c as an example, we use matrix U_c to denote the ***word-level representation*** of c. More specifically, $U_c = [e_c^1, e_c^2, ..., e_c^{n_c}]^T$, where n_c is the number of words in the dialog context c, and e_c^i is the BERT word embedding vector of the ith word in c. As mentioned in Section 4.3, given the context c, we can get topic embedding vectors. Let $U_{c,e,t} = [t_c^1, t_c^2, ..., t_c^K]^T$ be the ***topic embedding matrix*** where $U_{c,e,t} \in \mathbb{R}^{K \times m}$, K is the number of topics and m is the dimension of BERT word embedding. From topic modeling, we can also get the probability distribution of topics over context c, which is $[p_{c,t}^1, p_{c,t}^2, ..., p_{c,t}^K]$ where $p_{c,t}^i$ is the probability of the ith topic in c. We define the ***distribution-aware topic embedding matrix*** $U_{c,e,d,t} \in \mathbb{R}^{K \times m}$ as $[p_{c,t}^1 t_c^1, p_{c,t}^2 t_c^2, ..., p_{c,t}^K t_c^K]^T$, where each topic embedding vector is weighted by the probability of the topic in context c. We use U_c, $U_{c,e,d,t}$, and $U_{c,e,t}$ as the query, key and value matrices to construct a topic-aware attention function for context c as follows:

$$\text{TopicAtt}(U_c, U_{c,e,d,t}, U_{c,e,t}) = \text{softmax}(\frac{U_c(U_{c,e,d,t})^T}{\sqrt{m}})U_{c,e,t}$$

where $\text{softmax}(\frac{U_c(U_{c,e,d,t})^T}{\sqrt{m}})$ computes how much each word in context c attends to each of the topics by considering the topic distribution over c, and thus $\text{TopicAtt}(U_c, U_{c,e,d,t}, U_{c,e,t})$

encodes each word in c using the weighted sum of topic embeddings.

Replacing $Att(Q, K, V)$ in Equation (4.1) with the topic-aware attention $TopicAtt$ function, context c can be encoded as $f_{Attention}(U_c$, $U_{c,e,d,t}, U_{c,e,t})$, which is called ***topic-level representation*** of c. We denote the $f_{Attention}$ function that uses the topic-aware attention as $f_{TopicAttention}$.

4.4.2 Topic-Aware Response Selection Model

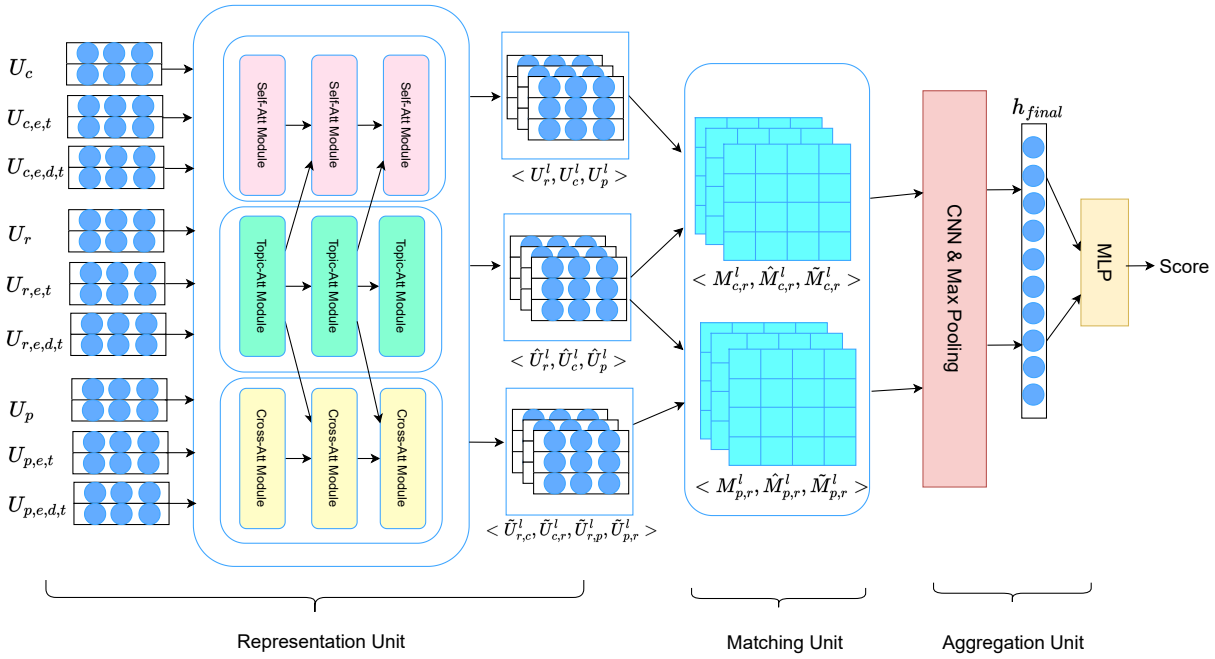

Figure 4.1: Topic-Aware Response Selection Model for predicting a score measuring how much r matches with c and p

The overview of our model is shown in Figure 4.1. The TARS model consists of three units: representation, matching, and aggregation. Given context c, persona p and response r, the *representation* unit contains multiple layers of attentive modules to get multi-level representations of c, p and r. The *matching* unit computes multiple matching matrices to

capture the similarities between c and r and similarities between p and r with different granularities. Finally, the *aggregation* unit concatenates all the matching matrices and feeds them to the CNN layers. The final matching score of r with respect to c and p is obtained via the Multi-Layer Perceptron with the sigmoid function. Next, we elaborate on each unit in detail.

Representation Unit

As described in Section 4.4.1, each of given context c, persona p, and response r has its word-level and topic-level representations. In addition, cross attention is used to jointly represent c and r and also p and r. To get each type of representation, 3 layers of attentive modules are used.

Word-level Representation Given response r, the initial representations of r are U_r^0, $U_{r,e,t}^0$, and $U_{r,e,d,t}^0$. The l+1-th layer representation of r at the word-level can be expressed as:

$$U_r^{l+1} = f_{\text{Attention}}(U_r^l, U_r^l, U_r^l)$$

which is a self-attention module based on word embeddings. Similarly, the word-level representation of context c or persona p at l+1-th layer is obtained as

$$U_c^{l+1} = f_{\text{Attention}}(U_c^l, U_c^l, U_c^l)$$

or

$$U_p^{l+1} = f_{\text{Attention}}(U_p^l, U_p^l, U_p^l)$$

Topic-level representation We use the Topic-Aware Attentive Module on the l-th layer to

attend to the topic-level semantics in word-level representations. That is,

$$\hat{U}_r^l = f_{\text{TopicAttention}}(U_r^l, U_{r,e,d,t}^0, U_{r,e,t}^0)$$

$$\hat{U}_c^l = f_{\text{TopicAttention}}(U_c^l, U_{c,e,d,t}^0, U_{c,e,t}^0)$$

$$\hat{U}_p^l = f_{\text{TopicAttention}}(U_p^l, U_{p,e,d,t}^0, U_{p,e,t}^0)$$

Topic-Aware Attentive Module enables fusing the word-level and topic-level semantics naturally.

Cross-attention representation The cross-attention representations capture the interactive semantics between context c and response r, and those between persona p and response r. Intuitively, c and r (and p and r) share similar semantics if r is an appropriate response given p and r. The expression of response r with crossed contextual semantics is obtained as:

$$\tilde{U}_{r,c}^l = f_{\text{Attention}}(U_r^l, U_c^l, U_c^l)$$

Similarly, we can obtain

$$\tilde{U}_{r,p}^l = f_{\text{Attention}}(U_r^l, U_p^l, U_p^l)$$

Matching Unit

The matching unit focuses on capturing the similarities between the context (or persona) and response at different granularities. Specifically, in each level l, three types of matching matrices of a context-response pair or a persona-response pair are calculated. Taking the

context-response pair as an example, the three matching matrices are:

$$M_{c,r}^{l} = U_{c}^{l}(U_{r}^{l})^{T}, \hat{M}_{c,r}^{l} = \hat{U}_{c}^{l}(\hat{U}_{r}^{l})^{T}, \tilde{M}_{c,r}^{l} = \tilde{U}_{r,c}^{l}(\tilde{U}_{c,r}^{l})^{T}$$

where $M_{c,r}^{l}, \hat{M}_{c,r}^{l}$ or $\tilde{M}_{c,r}^{l}$ is a matching matrix at the word, topic or cross attention level, respectively. The matching matrices of the persona-response pair, $M_{p,r}^{l}, \hat{M}_{p,r}^{l}$ and $\tilde{M}_{p,r}^{l}$, are computed in the same way, such that:

$$M_{p,r}^{l} = U_{p}^{l}(U_{r}^{l})^{T}, \hat{M}_{p,r}^{l} = \hat{U}_{p}^{l}(\hat{U}_{r}^{l})^{T}, \tilde{M}_{p,r}^{l} = \tilde{U}_{r,p}^{l}(\tilde{U}_{p,r}^{l})^{T}$$

Therefore, there are six matching matrices are obtained for each layer l in total.

Aggregation Unit

We concatenate all these matching matrices from different granularities and transform them into the final matching matrix M_{final}. M_{final} are fed to the CNN layers with max poolings and then flattened to the vector h_{final}. Furthermore, h_{final} is passed to the multi-layer perception (MLP) with the Sigmoid function. The matching score is defined as S_i.

The loss function for training our TARS model is expressed as:

$$\mathbb{L} = -1/N(\sum_{i}^{N} y_i \log(S_i) + (1 - y_i) \log(1 - S_i))$$

where N is the number of instances. In response selection, the responses are ranked based on their matching scores and the top one is selected.

4.5 Experiments

4.5.1 Datasets

In [57], a large empathy-oriented conversation corpus named persona-based empathetic conversation (PEC) was crawled from the reddit website. The PEC dataset was also studied in other research work [68]. On Reddit, users are allowed to discuss any topic they are interested in. The PEC dataset includes two subsets of empathetic conversation data: *happy* and *offmychest*, crawled from happy and offmychest Reddit communities, respectively. Users prefer to share posts with positive sentiments on the happy community, while most posts on the offmychest community are negative. Persona data are also offered in the PEC dataset. Persona data consist of the utterances posted by the corresponding user following the rules, such as the first word of the sentence should be "i" (e.g., i love my job.). The training data and testing data of the *happy* community have 157K and 23K conversations, 93K and 19K personas respectively. Additionally, the *offmychest* dataset includes 124K conversations and 89K personas in training data, and 15K conversations and 16K personas in testing data. We conduct our experiments using the PEC dataset, as it provides a corpus of real dialog contexts and responses, as well as a collection of personas.

We use the Askreddit dataset [69] to train our topic models. This dataset is the single-turn conversation data based on the open domain Askreddit community. Notice that one query on Reddit can receive multiple responses. The Askreddit dataset covers a variety of topics. In total, the Askreddit dataset includes 439K queries and 1,528K responses.

4.5.2 Baselines

We consider four state-of-the-art baseline models for the response selection task: **VEC**: It encodes the context, persona, and response with the BERT embeddings. The similarity between context and response (or between persona and response) is calculated by the cosine similarity function. **CLSTM**[49]: It encodes the context and response with the LSTM model. Persona is ignored in this scenario. **CoBERT**[57]: The model learns hop-1 and hop-2 co-attentions between the context and response and also between the persona and response. **DIM**[55]: The model adopts fixed word2vec embeddings to initialize the context, persona, and response. These representations are further encoded by BiLSTM. Cross-matching between the context (or persona) and response is also considered in DIM.

4.5.3 Implementation Details

To train the topic model (i.e., LDA) on the Askreddit data [69], we tested four different topic numbers (50, 100, 200, and 300) on the Askreddit dataset. We trained our TARS model with these 4 different topic models and evaluated their performance on a validation dataset (different from the training and test datasets) of *offmychest* and found that the topic model with 100 topics yielded the best results. Thus, we set the topic number to 100 for the rest of our experiment. Additionally, the number of top words used to represent topics is set to 20. *BERT-base-uncased* is adopted in our TARS model. The BERT embedding dimension is 768. The maximum numbers of words for the context, persona, and response are 256, 231, and 32, respectively. We use three-layer CNNs and optimize the TARS model via Adam with the learning rate 2e-5. The batch size is 16, and our TARS model is trained on a single NVIDIA A100-SXM4-40GB.

4.5.4 Experimental Results

	happy					offmychest				
Models	R@1	R@5	R@10	R@20	MRR	R@1	R@5	R@10	R@20	MRR
VEC	4.53	13.19	20.42	31.90	10.61	6.17	16.55	24.33	36.43	12.97
CLSTM	6.53	22.72	35.35	53.47	16.09	8.76	26.80	40.59	58.15	19.07
DIM	22.03	40.77	52.00	65.94	32.08	29.25	47.49	57.40	70.65	38.94
CoBERT	36.15	60.94	72.40	84.07	48.14	45.90	69.23	79.14	88.11	57.04
TARS	**37.25**	**62.07**	**72.98**	**84.71**	**49.15**	**47.05**	**70.53**	**79.96**	**88.75**	**58.12**

Table 4.1: Results of different models on happy and offmychest datasets

Overall Results. All the models are evaluated by Recall@K and Mean Reciprocal Rank (MRR), same as in [57]. Table 4.1 lists different models' performances on *happy* and *offmychest* data sets. The table shows that VEC performs poorly on the response selection task. One possible reason is that the VEC method adopts original BERT embedding and simply uses cosine similarity. Therefore, the representations of the context, persona and response are not fully explored in VEC. Due to the lack of multi-grained semantic matching, CLSTM also performs poorly. The performance of DIM is worse than CoBERT, which confirms the importance of BERT-based embeddings. More importantly, it is observed that our TARS model outperforms other baselines on both data sets by all metrics with noticeable improvements. Especially, compared with the state-of-the-art CoBERT model, our model gains 1.10% and 1.15% improvement in Recall@1 on *happy* and *offmychest* data sets, respectively. Since in real-life scenarios, the chitchat system is forced to pick up one unique response from the pool of response candidates and return it to the end user. Recall@1 is the gold measure for the response selection task. This result indicates that our TARS model captures more explicit information (i.e., topics) from the context, persona, and response than CoBERT, which only accounts for the hop-1 and hop-2 co-attentions. Meanwhile, TARS employs multi-grained semantic matchings, also contributing to the improvement of the

response selection task.

Ablation Study. In ablation studies for our TARS model, we remove Topic-Aware Attentive Module and cross attentive module, respectively. Table 4.2 and Table 4.3 show that dropping any of these two attentive modules impairs the TARS performance. Especially for the Topic-Aware Attentive Module, without it, the performance of TARS drops to 36.49% and 45.79% measured by Recall@1 on the *happy* and *offmychest* data sets, which demonstrates the benefits of Topic-Aware Attentive Module on exploring the topic-level semantics of the context, persona, and response. Furthermore, cross attentive module also positively impacts the model performance.

Model	R@1	R@5	R@10	R@20	MRR
TARS	**37.25**	**62.07**	**72.98**	**84.71**	**49.15**
w/o Topic Attn	36.49	61.23	72.77	84.55	48.46
w/o Cross Attn	36.65	61.43	72.69	84.22	48.67

Table 4.2: Ablation study on happy Dataset

Model	R@1	R@5	R@10	R@20	MRR
TARS	**47.05**	**70.53**	**79.96**	**88.75**	**58.12**
w/o Topic Attn	45.79	69.49	78.82	87.78	56.97
w/o Cross Attn	46.18	70.31	79.33	88.43	57.43

Table 4.3: Ablation Results on offmychest Dataset

4.6 Summary

In this chapter, we proposed TARS for the topic-aware response selection in the persona-based chitchat system. TARS employs the Topic-Aware Attentive Module, which attends to the topic-level semantics when representing the context, persona, and response. Experimental results demonstrate the importance of the Topic-Aware Attentive Module, and TARS outperforms

the state-of-the-art methods. In the future, we will exploit relations between emotions and topics for better topic-aware response selection.

Chapter 5

Conclusion

This chapter presents a summary of the innovative methods proposed in this thesis and suggests future research directions for few-shot intent detection and response generation in conversational dialogue systems.

5.1 Thesis Conclusion

This thesis introduced and proposed novel models for addressing key challenges in the field of dialogue systems. In particular, the thesis focus on the development of the few-shot intent detection in the task-oriented dialogue systems to enhances interpretability and establish the people's trust in the decision-making process in industry (Chapter 3), and the novel approach of response selection task in the open-domain chitchat dialogue systems to retrieve the most coherent and consistent response to user (Chapter 4).

In Chapter 3, we presented a few-shot intent detection model for an enterprise's conversational dialogue system, utilizing an intent topological tree to efficiently guide the searching for user intents though large language models (LLMs). The intents detection is resolved

based on semantic similarities between user utterances and either the textual descriptions of internal nodes or intent examples within leaf nodes. As a result, the topological tree enables the effective intent resolution for few-shot intent detection. This finding demonstrates that an off-the-shelf language model can work reasonably well in a large enterprise deployment without fine-tuning, and its performance can be further improved with contrastive fine-tuning as more domain-specific data becomes available. In addition, the utilization of a topological intent tree offers increased interpretability, fostering greater trust in the decision-making process of the model.

In Chapter 4, a Topic-Aware Response Selection (TARS) model for persona-based chitchat systems is proposed, designed to retrieve the most appropriate response from a set of candidates based on the dialog context and the agent's persona. This model captures multi-grained matching between the dialogue context and a response, as well as between the persona and a response at both word and topic levels. Empirical results on public persona-based empathetic conversation (PEC) data demonstrates the promising performance of the TARS model in the response selection task.

In conclusion, this thesis has introduced two innovative models aimed at addressing crucial challenges in the field of task-oriented and open-domain chitchat dialogue systems. Through resolving challenges related to natural language understanding and response generation, the proposed models have significantly enhanced the capabilities and user experiences of dialogue systems. Furthermore, the models have contributed to the interpretability, trustworthiness, and overall performance of conversational dialogue systems in deployment.

5.2 Future Work

While this study has made important contributions to the field of conversational dialogue systems, there are several future research that could build upon tthe foundations laid by this thesis. In this final section, we outline several potential research directions for future work that could further enhance and expand upon the findings presented in this thesis.

Few-shot Intent Detection In Chapter 3, we introduced the topological intent tree as a guide for searching user utterances in enterprise dialogue systems. To advance research on few-shot intent detection, future studies could focus on designing a dynamic graph that adjusts to the diverse relationships between intents and utterances. Furthermore, we aim to integrate external knowledge sources to create a more comprehensive intent graph.

Recently, Graph Neural Networks (GNNs) have demonstrated promising outcomes in few-shot learning tasks, including [70, 71, 72]. To enhance few-shot intent detection, we plan to investigate integrating pre-trained language models (such as BERT, GPT, RoBERTa, etc.) into various GNN architectures, such as Graph Convolutional Networks (GCN), GraphSage, and Graph Attention Networks (GAN). Additionally, we will utilize dynamic graphs to represent the relationships between intents and utterances, leveraging semantic representations for improved performance in the few-shot intent detection task in dialogue systems.

Response Generation In Chapter 4, we eexplores methods for retrieving the most coherent and consistent response from a set of candidate utterances in open-domain chit-chat dialogue systems. However, this study focuses solely on the persona of the responder or agent, disregarding the responder's emotions and other external contextual information. By integrating these contextual information into our model, we can enhance its ability to capture relevant information and retrieve coherent and accurate responses.

Furthermore, the generation of appropriate and coherent responses to users is a significant

challenge of open-domain chitchat dialogue systems. Currently, prompt learning plays an essential role in response generation, it optimizes only a small portion of task-specific prompts or related modules, while keeping the Pre-trained Language Models (PLMs) parameters frozen [73]. By incorporating contextual information, such as personas and emotions of responders or agent, it is possible to adapt pre-trained models to generate high-quality responses with minimal or no labeled data. This is accomplished by optimizing continuous prompt embeddings specific to the dialogue context. Moreover, prompt learning has been proven to be substantially more effective than fine-tuning in zero or few-shot learning for response generation [74]. Future research will focus on few-shot learning with prompt learning for response generation.

Bibliography

[1] Nils Reimers and Iryna Gurevych. "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Nov. 2019. URL: https://arxiv.org/abs/1908.10084.

[2] Hongshen Chen et al. "A Survey on Dialogue Systems: Recent Advances and New Frontiers". In: *SIGKDD Explor.* 19.2 (2017), pp. 25–35.

[3] Iñigo Casanueva et al. "Efficient Intent Detection with Dual Sentence Encoders". In: *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*. Online: Association for Computational Linguistics, July 2020, pp. 38–45. DOI: 10.18653/v1/2020.nlp4convai-1.5. URL: https://aclanthology.org/2020.nlp4convai-1.5.

[4] Joseph Weizenbaum. "ELIZA—a Computer Program for the Study of Natural Language Communication between Man and Machine". In: *Commun. ACM* 9.1 (Jan. 1966), pp. 36–45. ISSN: 0001-0782. DOI: 10.1145/365153.365168. URL: https://doi.org/10.1145/365153.365168.

[5] Suket Arora, Kamaljeet Batra, and Sarabjit Singh. "Dialogue system: A brief review". In: *arXiv preprint arXiv:1306.4134* (2013).

[6] Jinjie Ni et al. "Recent advances in deep learning based dialogue systems: A systematic survey". In: *Artificial intelligence review* (2022), pp. 1–101.

[7] Eleni Adamopoulou and Lefteris Moussiades. "An Overview of Chatbot Technology". In: May 2020, pp. 373–383. ISBN: 978-3-030-49185-7. DOI: 10.1007/978-3-030-49186-4_31.

[8] Diksha Khurana et al. "Natural Language Processing: State of The Art, Current Trends and Challenges". In: *CoRR* abs/1708.05148 (2017). arXiv: 1708.05148. URL: http://arxiv.org/abs/1708.05148.

[9] Pavel Kucherbaev, Alessandro Bozzon, and Geert-Jan Houben. "Human-Aided Bots". In: *IEEE Internet Computing* 22.6 (2018), pp. 36–43. DOI: 10.1109/MIC.2018.252095348.

[10] Kiran Ramesh et al. "A Survey of Design Techniques for Conversational Agents". In: *Information, Communication and Computing Technology*. Ed. by Saroj Kaushik et al. Singapore: Springer Singapore, 2017, pp. 336–350. ISBN: 978-981-10-6544-6.

[11] Ho Thao Hien et al. "Intelligent Assistants in Higher-Education Environments: The FIT-EBot, a Chatbot for Administrative and Learning Support". In: *Proceedings of the Ninth International Symposium on Information and Communication Technology*. SoICT 2018. Danang City, Viet Nam: Association for Computing Machinery, 2018, pp. 69–76. ISBN: 9781450365390. DOI: 10.1145/3287921.3287937. URL: https://doi.org/10.1145/3287921.3287937.

[12] S. Singh et al. "Open source NLG systems: A survey with a vision to design a true NLG system". In: 9 (Jan. 2016), pp. 4409–4421.

[13] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. "Sequence to sequence learning with neural networks". In: *Advances in neural information processing systems* 27 (2014).

[14] Jason Weston, Sumit Chopra, and Antoine Bordes. "Memory networks". In: *arXiv preprint arXiv:1410.3916* (2014).

[15] Xiuyi Chen, Jiaming Xu, and Bo Xu. "A Working Memory Model for Task-oriented Dialog Response Generation". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 2687–2693. DOI: 10.18653/v1/P19-1258. URL: https://aclanthology.org/P19-1258.

[16] Wanwei He et al. "Amalgamating Knowledge from Two Teachers for Task-oriented Dialogue System with Adversarial Training". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 3498–3507. DOI: 10.18653/v1/2020.emnlp-main.281. URL: https://aclanthology.org/2020.emnlp-main.281.

[17] Zhiliang Tian et al. "Learning to Abstract for Memory-augmented Conversational Response Generation". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 3816–3825. DOI: 10.18653/v1/P19-1371. URL: https://aclanthology.org/P19-1371.

[18] Qingfu Zhu et al. "Retrieval-Enhanced Adversarial Training for Neural Response Generation". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 3763–3773. DOI: 10.18653/v1/P19-1366. URL: https://aclanthology.org/P19-1366.

[19] Cao Liu et al. "Vocabulary Pyramid Network: Multi-Pass Encoding and Decoding with Multi-Level Vocabularies for Response Generation". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 3774–3783. DOI: 10.18653/v1/P19-1367. URL: https://aclanthology.org/P19-1367.

[20] OpenAI. *GPT-4 Technical Report*. 2023. arXiv: 2303.08774 [cs.CL].

[21] Zheng Zhang et al. "Recent advances and challenges in task-oriented dialog systems". In: *Science China Technological Sciences* (2020), pp. 1–17.

[22] Jetze Schuurmans and Flavius Frasincar. "Intent Classification for Dialogue Utterances". In: *IEEE Intelligent Systems* 35.1 (2020), pp. 82–88. DOI: 10.1109/MIS.2019.2954966.

[23] S. Mehri, M. Eric, and D. Hakkani-Tur. "DialoGLUE: A Natural Language Understanding Benchmark for Task-Oriented Dialogue". In: *ArXiv* abs/2009.13570 (2020).

[24] Daniel Cer et al. "Universal Sentence Encoder for English". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Brussels, Belgium: Association for Computational Linguistics, Nov. 2018, pp. 169–174. DOI: 10.18653/v1/D18-2029. URL: https://aclanthology.org/D18-2029.

[25] Matthew Henderson et al. "ConveRT: Efficient and Accurate Conversational Representations from Transformers". In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, Nov. 2020, pp. 2161–2174. DOI: 10.18653/v1/2020.findings-emnlp.196. URL: https://aclanthology.org/2020.findings-emnlp.196.

[26] Jianguo Zhang et al. "Discriminative Nearest Neighbor Few-Shot Intent Detection by Transferring Natural Language Inference". In: *Proceedings of the 2020 Conference on*

Empirical Methods in Natural Language Processing (EMNLP). Online: Association for Computational Linguistics, Nov. 2020, pp. 5064–5082. DOI: 10.18653/v1/2020.emnlp-main.411. URL: https://aclanthology.org/2020.emnlp-main.411.

[27] Jianguo Zhang et al. "Few-Shot Intent Detection via Contrastive Pre-Training and Fine-Tuning". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 1906–1912. DOI: 10.18653/v1/2021.emnlp-main.144. URL: https://aclanthology.org/2021.emnlp-main.144.

[28] Shikib Mehri and Mihail Eric. "Example-Driven Intent Prediction with Observers". In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, June 2021, pp. 2979–2992. DOI: 10.18653/v1/2021.naacl-main.237. URL: https://aclanthology.org/2021.naacl-main.237.

[29] Ivan Vulić et al. "ConvFiT: Conversational Fine-Tuning of Pretrained Language Models". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 1151–1168. DOI: 10.18653/v1/2021.emnlp-main.88. URL: https://aclanthology.org/2021.emnlp-main.88.

[30] Raia Hadsell, Sumit Chopra, and Yann LeCun. "Dimensionality Reduction by Learning an Invariant Mapping". In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA*. IEEE Computer Society, 2006, pp. 1735–1742.

[31] Ting Chen et al. "A Simple Framework for Contrastive Learning of Visual Representations". In: *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*. Vol. 119. Proceedings of Machine Learning Research. PMLR, 2020, pp. 1597–1607.

[32] Tianyu Gao, Xingcheng Yao, and Danqi Chen. "SimCSE: Simple Contrastive Learning of Sentence Embeddings". In: *CoRR* abs/2104.08821 (2021).

[33] Che Liu et al. "DialogueCSE: Dialogue-based Contrastive Learning of Sentence Embeddings". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. Association for Computational Linguistics, 2021, pp. 2396–2406.

[34] Xiao Pan et al. "Contrastive Learning for Many-to-many Multilingual Neural Machine Translation". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*. Ed. by Chengqing Zong et al. Association for Computational Linguistics, 2021, pp. 244–258.

[35] Yixin Liu and Pengfei Liu. "SimCLS: A Simple Framework for Contrastive Learning of Abstractive Summarization". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021*. Ed. by Chengqing Zong et al. Association for Computational Linguistics, 2021, pp. 1065–1072.

[36] Congying Xia et al. "Incremental few-shot text classification with multi-round new classes: Formulation, dataset and system". In: *arXiv preprint arXiv:2104.11882* (2021).

[37] R. Hadsell, S. Chopra, and Y. LeCun. "Dimensionality Reduction by Learning an Invariant Mapping". In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 2. 2006, pp. 1735–1742. DOI: 10.1109/CVPR.2006.100.

[38] Zhuofeng Wu et al. "Clear: Contrastive learning for sentence representation". In: *arXiv preprint arXiv:2012.15466* (2020).

[39] Tianyu Gao, Xingcheng Yao, and Danqi Chen. "SimCSE: Simple Contrastive Learning of Sentence Embeddings". In: *arXiv preprint arXiv:2104.08821* (2021).

[40] Stefan Larson et al. "An Evaluation Dataset for Intent Classification and Out-of-Scope Prediction". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 2019. URL: https://www.aclweb.org/anthology/D19-1131.

[41] Xingkun Liu et al. *Benchmarking natural language understanding services for building conversational agents*. 2019.

[42] Stephen Roller et al. "Recipes for Building an Open-Domain Chatbot". In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*. 2021, pp. 300–325.

[43] Chen Xu et al. "COSPLAY: Concept Set Guided Personalized Dialogue Generation Across Both Party Personas". In: *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '22. Madrid,

Spain: Association for Computing Machinery, 2022, pp. 201–211. ISBN: 9781450387323. DOI: 10.1145/3477495.3531957. URL: https://doi.org/10.1145/3477495.3531957.

[44] Jia-Chen Gu et al. "Partner matters! an empirical study on fusing personas for personalized response selection in retrieval-based chatbots". In: *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2021, pp. 565–574.

[45] Zhengyi Ma et al. "One Chatbot Per Person: Creating Personalized Chatbots Based on Implicit User Profiles". In: *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '21. Virtual Event, Canada: Association for Computing Machinery, 2021, pp. 555–564. ISBN: 9781450380379. DOI: 10.1145/3404835.3462828. URL: https://doi.org/10.1145/3404835.3462828.

[46] Haoyu Song et al. "A Stack-Propagation Framework for Low-Resource Personalized Dialogue Generation". In: *ACM Trans. Inf. Syst.* (Sept. 2022). Just Accepted. ISSN: 1046-8188. DOI: 10.1145/3563389. URL: https://doi.org/10.1145/3563389.

[47] Baotian Hu et al. "Convolutional neural network architectures for matching natural language sentences". In: *Advances in neural information processing systems* 27 (2014), pp. 2042–2050.

[48] Hao Wang et al. "A Dataset for Research on Short-Text Conversations". In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*. ACL, 2013, pp. 935–945.

[49] Ryan Lowe et al. "The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems". In: *CoRR* abs/1506.08909 (2015).

[50] Xiangyang Zhou et al. "Multi-Turn Response Selection for Chatbots with Deep Attention Matching Network". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*. Association for Computational Linguistics, 2018, pp. 1118–1127.

[51] Yu Wu et al. "Sequential Matching Network: A New Architecture for Multi-turn Response Selection in Retrieval-Based Chatbots". In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*. Ed. by Regina Barzilay and Min-Yen Kan. Association for Computational Linguistics, 2017, pp. 496–505.

[52] Chongyang Tao et al. "One Time of Interaction May Not Be Enough: Go Deep with an Interaction-over-Interaction Network for Response Selection in Dialogues". In: *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*. Ed. by Anna Korhonen, David R. Traum, and Lluıs Màrquez. Association for Computational Linguistics, 2019, pp. 1–11.

[53] Chunyuan Yuan et al. "Multi-hop Selector Network for Multi-turn Response Selection in Retrieval-based Chatbots". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, 2019, pp. 111–120.

[54] Xiangyang Zhou et al. "Multi-view Response Selection for Human-Computer Conversation". In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics, Nov. 2016, pp. 372–381. DOI: 10.18653/v1/D16-1036. URL: https://aclanthology.org/D16-1036.

[55] Jia-Chen Gu et al. "Dually Interactive Matching Network for Personalized Response Selection in Retrieval-Based Chatbots". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*. Association for Computational Linguistics, 2019, pp. 1845–1854.

[56] Jiwei Li et al. "A Persona-Based Neural Conversation Model". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics, 2016.

[57] Peixiang Zhong et al. "Towards Persona-Based Empathetic Conversational Models". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*. Association for Computational Linguistics, 2020, pp. 6556–6566.

[58] Hongjin Qian et al. "Learning Implicit User Profile for Personalized Retrieval-Based Chatbot". In: *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*. Ed. by Gianluca Demartini et al. ACM, 2021, pp. 1467–1477.

[59] Jiashu Zhao et al. "Are Topics Interesting or Not? An LDA-Based Topic-Graph Probabilistic Model for Web Search Personalization". In: *ACM Trans. Inf. Syst.* 40.3 (Dec. 2022). ISSN: 1046-8188. DOI: 10.1145/3476106. URL: https://doi.org/10.1145/3476106.

[60] Qiming Li et al. "Topic-aware Intention Network for Explainable Recommendation with Knowledge Enhancement". In: *ACM Transactions on Information Systems* (2023).

[61] Yu Wu et al. "Response selection with topic clues for retrieval-based chatbots". In: *Neurocomputing* 316 (2018), pp. 251–261.

[62] Jinhua Peng et al. "Integrating Bayesian and Neural Networks for Discourse Coherence". In: *Companion of The 2019 World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*. Ed. by Sihem Amer-Yahia et al. ACM, 2019, pp. 294–300.

[63] Lixing Zhu et al. "Topic-Driven and Knowledge-Aware Transformer for Dialogue Emotion Detection". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Virtual Event*. Association for Computational Linguistics, 2021, pp. 1571–1582.

[64] Yicheng Zou et al. "Topic-Oriented Spoken Dialogue Summarization for Customer Service with Saliency-Aware Topic Modeling". In: *Thirty-Fifth AAAI Conference on Artificial Intelligence, Virtual Event*. AAAI Press, 2021, pp. 14665–14673.

[65] Saizheng Zhang et al. "Personalizing Dialogue Agents: I have a dog, do you have pets too?" In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*.

Ed. by Iryna Gurevych and Yusuke Miyao. Association for Computational Linguistics, 2018, pp. 2204–2213.

[66] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. "Latent Dirichlet Allocation". In: *J. Mach. Learn. Res.* 3 (2003), pp. 993–1022.

[67] Ashish Vaswani et al. "Attention is All you Need". In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. 2017, pp. 5998–6008.

[68] Chujie Zheng et al. "CoMAE: A Multi-factor Hierarchical Framework for Empathetic Response Generation". In: *Findings of the Association for Computational Linguistics: ACL/IJCNLP, Online Event*. Association for Computational Linguistics, 2021, pp. 813–824.

[69] Yuwei Wu, Xuezhe Ma, and Diyi Yang. "Personalized Response Generation via Generative Split Memory Network". In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*. Ed. by Kristina Toutanova et al. Association for Computational Linguistics, 2021, pp. 1956–1970.

[70] Victor Garcia and Joan Bruna. *Few-Shot Learning with Graph Neural Networks*. 2018. arXiv: `1711.04043 [stat.ML]`.

[71] Fan Zhou et al. "Meta-gnn: On few-shot node classification in graph meta-learning". In: *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2019, pp. 2357–2360.

[72] Tianyuan Yu et al. "Hybrid graph neural networks for few-shot learning". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 36. 3. 2022, pp. 3179–3187.

[73] Xiaodong Gu, Kang Min Yoo, and Sang-Woo Lee. "Response generation with context-aware prompt learning". In: *arXiv preprint arXiv:2111.02643* (2021).

[74] Chujie Zheng and Minlie Huang. "Exploring prompt-based few-shot learning for grounded dialog generation". In: *arXiv preprint arXiv:2109.06513* (2021).