From Knowledge Augmentation to Multi-tasking: Towards Human-like Dialogue Systems

NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

Yang Tianji

School of Computer Science and Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

2023

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

24/7/2022

Date

Yang Tianji

Yang Tianji

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

28/7/2022

. . . .

Date

. . . .

A/Prof. Erik Cambria

Authorship Attribution Statement

This thesis contains material from 3 papers published in the following peer-reviewed journals and conferences in which I am listed as an author. Tom Young is my publication name.

Chapter 2 is published as Young, Tom, Frank Xing, Vlad Pandelea, Jinjie Ni, and Erik Cambria. "Fusing task-oriented and open-domain dialogues in conversational agents." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, no. 10, pp. 11622-11629. 2022.

The contributions of the co-authors are as follows:

- I came up with the key idea, designed most of the experiments, implemented all source code, and conducted most of the experiments. I prepared the manuscript draft.
- Frank Xing played a key role in the data collection process.
- I had regular discussions with Vlad Pandelea, Jinjie Ni, Frank Xing and Prof. Erik Cambria. The former two handled human evaluation schemes.
- Prof. Erik Cambria revised and edited the manuscript drafts.

Chapter 3 is published as Young, Tom, Vlad Pandelea, Soujanya Poria, and Erik Cambria. "Dialogue systems with audio context." Neurocomputing 388 (2020): 102-109.

The contributions of the co-authors are as follows:

- I came up with the key idea, designed most of the experiments, implemented part of the source code, and conducted part of the experiments. I prepared most of the manuscript draft.
- Vlad Pandelea came up some ideas in the experiments including important feature processing techniques. He designed the rest of the experiments, implemented the rest of the source code, conducted the rest of the experiments and prepared the rest of the manuscript draft.
- Soujanya Poria and Prof. Erik Cambria provided useful insight to the project. I had regular discussions with them.
- Prof. Erik Cambria revised and edited the manuscript drafts.

Chapter 4 is published as Young, Tom, Erik Cambria, Iti Chaturvedi, Hao Zhou, Subham Biswas, and Minlie Huang. "Augmenting end-to-end dialogue systems with commonsense knowledge." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32, no. 1. 2018.

The contributions of the co-authors are as follows:

- I came up with the key idea, designed all the experiments, implemented all source code, and conducted most of the experiments. I prepared the manuscript draft.
- I had discussions with Minlie Huang, Hao Zhou, Iti Chaturvedi and Prof. Erik Cambria. And they provided useful feedback.
- Subham Biswas conducted part of data processing.
- Prof. Erik Cambria revised and edited the manuscript drafts.

Yang Tianji

28/7/2022

. . . .

Date   Yang Tianji

Acknowledgements

First and foremost, I wish to express my greatest gratitude to my advisor Prof. Erik Cambria, for his valuable support during the course of my study and research. He has advised me ever since I wrote my first machine learning model. His experience and passion have encouraged me in all the time of my academic research.

Apart from my advisor, I would also like to thank other senior researchers who offered me useful suggestions and support. Most notably, Prof. Soujanya Poria, Prof. Frank Xing, Prof. Minlie Huang, Prof. Iti Chaturvedi, Dr. Yukun Ma, Dr. Qian Chen, Dr. Qian Liu, Dr. Rui Mao, Dr. Yang Li, Dr. Haiyun Peng, and Dr. Xiaoshi Zhong often let me pick their mind.

My labmates and collaborators, such as Mr. Jinjie Ni, Mr. Vlad Pandelea, Ms. Luyao Zhu, Mr. Wei Li, Dr. Hao Zhou and Dr. Devmanyu Hararika were always there to lend me a hand. And I often learned new things from discussions with them.

Administrative staff from the SCSE graduate office, including Mr. Ker Shen Hui, Ms. Chiam Poh Ling, Ms. Len Ah Chan and Ms. Juliana Binte Jaapar helped me navigate many matters.

I want to thank all my colleagues and friends in Singapore for their wonderful company.

Finally, my gratitude and admiration go to my parents. They have been a consistent support to me and their lifelong learning philosophy inspires me greatly.

“What if solving one problem could lead to solutions to thousands more?”

—DeepMind

To my dear family

Contents

Summary

The goal of building dialogue agents that can converse with humans naturally has been a long-standing dream of researchers since the early days of artificial intelligence. The well-known Turing Test proposed to judge the ultimate validity of an artificial intelligence agent on the indistinguishability of its dialogues from humans'. It should come as no surprise that human-level dialogue systems are very challenging to build. But, while early effort on rule-based systems found limited success, the emergence of deep learning enabled great advance on this topic.

The works covered in this thesis originated in an era where data-driven deep learning based dialogue systems were beginning to take off. Dialogue systems trained on message-response pairs found in social media began to show abilities of conducting natural conversations. But they were limited in many ways such as lacking knowledge grounding, multimodality and multi-utility.

In this thesis, we focus on methods that address these numerous issues that have been imposing the gap between artificial conversational agents and human-level interlocutors. These methods were proposed and experimented with in ways that were inspired by general state-of-the-art AI methodologies. But they also targeted the characteristics that dialogue systems possess.

First of all, we expand the variety of information that dialogue systems can be dependent on. In its simplest and most common form, a dialogue consists of responses and their preceding textual context. This representation, however, falls short compared to real-world human conversation, which is often dependent on other modalities and specific knowledge bases.

To the end of conditioning dialogues on more modalities, we explore dialogue generation augmented by the audio representation of the input. We design an auxiliary response classification task to learn suitable audio representation for our dialogue generation objective. We use word-level modality fusion for integrating audio features into the Sequence to Sequence learning framework. Our model can generate

appropriate responses corresponding to the emotion and emphasis expressed in the audio.

Commonsense knowledge has to be integrated into the dialogue system effectively for it to respond to human utterances in an interesting and engaging way. As the first attempt to integrating a large commonsense knowledge base into end-to-end conversational models, we propose a model to jointly take into account the context and its related commonsense knowledge for selecting an appropriate response. We demonstrate that the knowledge-augmented models are superior to their knowledge-free counterparts.

While the two directions mentioned above endeavor to ground the dialogues on various new information, they are not the only challenges that dialogue systems face. Traditionally, the goal of building intelligent dialogue systems has largely been separately pursued assuming two separate utilities: task-oriented dialogue systems, which perform task-specific functions, and open-domain dialogue systems, which focus on non-goal-oriented chitchat. The two dialogue modes can potentially be intertwined together seamlessly in the same conversation, as easily done by a friendly human assistant. This thesis also covers our effort on addressing the problem of fusing the two dialogue modes in multi-turn dialogues. We build a new dataset FusedChat, which contains conversation sessions containing exchanges from both dialogue modes with inter-mode contextual dependency. We propose two baseline models on this task and analyze their accuracy.

Last but not least, we demonstrate our effort on addressing the computational efficiency issue that large-scale retrieval-based dialogue systems face. Strong retrieval-based dialogue systems that are based on a large natural candidate set can produce diverse and controllable responses. However, a large candidate set could be computationally costly. We propose methods that support a fast and accurate response retrieval system. To boost accuracy, we adopt a knowledge distillation approach where a very strong yet computationally expensive joint encoding model is used to facilitate training our encoders. We then boost the retrieval speed by adopting a learning-based candidate screening method to further reduce inference time. We demonstrate that our model performs strongly in terms of retrieval accuracy and speed trade-off.

In summary, this thesis systematically demonstrates our effort on innovating dialogue systems. Through our experiments, we found that through new designs based upon general state-of-the-art NLP methodologies, dialogue systems can be made faster, multimodal, capable of multiple utilities and grounded on useful external information. We believe that the research questions that we focused on are important aspects for ultimately improving automated dialogue agents to human-level. The main contribution of the works covered in the thesis lies in their initializing effects (to a certain degree) on these directions that have been continuously worked on by researchers till this day.

With our effort of innovating dialogue systems spanning the last 4 years, and state-of-the-art NLP models fast evolving year by year, we note that the models used in some of our works in the earlier years (e.g., LSTMs) cannot compete with the state-of-the-art models available today (e.g., GPT4). In such cases, we briefly and systematically explain following works (current state-of-the-art) that stemmed from the methodologies shown in our work, especially those based on recent advances of large language models.

List of Figures

List of Tables

Acronyms

AMT	Amazon Mechanical Turk
BLEU	bilingual evaluation understudy
BERT	Bidirectional Encoder Representations from Transformers
GPT	Generative Pretrained Transformer (117M parameters)
GPT2	Generative Pretrained Transformer (2nd version, 1.5B parameters)
GPT3	Generative Pretrained Transformer (3rd version, 175B parameters)
GPT4	Generative Pretrained Transformer (4th version, unknown number of parameters)
LSTM	Long-short Term Memory
MIPS	Maximum Inner Product Search
NLP	Natural Language Processing
ODD	Open-domain Dialogue
RNN	Recurrent Neural Networks
Seq2Seq	Sequence to Sequence
TOD	Task-oriented Dialogue
e.g.	exemplum gratia (en: for example)
et al.	et alia (en: and others)
i.e.	id est (en: that is)

Chapter 1

Introduction

Dialogue systems are becoming increasingly useful in today's world. In contrast to early rigid rule-based automated dialogue systems used in phone calls to customer services, the modern dialogue systems can converse on diverse topics ranging from your pet to recent financial news, and, even further, they can book a restaurant for your birthday party. At present, dialogue systems are one of the hot topics in natural language processing (NLP) and are demanded for both businesses and individual users. The global chatbot market is foreseen to expand from $3.6 billion in 2020 to $12.4 billion by 2026 [4, 5]. About 30% of people use virtual assistants like Google Assistant and Amazon Echo at least once a month as of 2021 [6]. And it is estimated that about 80% of all companies implement at least one chatbot somewhere in their business [7]. Such prevalent usages of dialogue systems either drastically reduce repetitive human labor for companies or make the lives of individual users more convenient.

In the early years, dialogue systems were mostly rule-based (e.g., ELIZA [8] and PARRY [9]). The general principle for such methods is looking for keywords within a conversation that can be used to instruct an agent to provide predefined responses. It is important to consider that even though the agent has a script of meaningful and intelligent responses, it has a severely limited understanding of the language itself. The dialogue flows of these systems are predetermined, which restrains the applications of the dialogue systems within very limited scenarios [5].

Similar to other fields in NLP, recent breakthroughs made in deep learning and self-supervised pretraining have greatly propelled the field of dialogue systems forward. Chronologically, two significant paradigm shifts happened recently. The first one was the introduction of neural networks for sequence to sequence (Seq2Seq) learning problems (Sutskever et al., 2014). Coupled with abundant data and computational resources, this paradigm first made breakthrough in neural translation and quickly penetrated state-of-the-art dialogue systems (Vinyals and Le, 2015). This paradigm discards feature engineering all together and let the neural network automatically learn to project the conversational context to the response through encoding implicit features and decoding from them. It proved to be able to generate quite natural-sounding responses.

The second paradigm shift coupled large-scale self-supervised pretraining with large neural networks, such as XLNet (Yang et al., 2019), BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019) and GPT3 (Brown et al., 2020), to give them general language understanding capabilities before fine-tuning them on specific tasks. It has been shown that utilizing such pretrained models achieves state-of-the-art results in multiple NLP tasks (Devlin et al., 2018). Dialogue systems are no exception. A lot of dialogue datasets that people normally train their models on can be relatively small and it is hard to learn enough commonsense knowledge or language variations. Therefore they do not always generalize well to unfamiliar contexts. By initializing with pretrained models, these issues are alleviated and the performance of the dialogue systems is further improved.

In this thesis, we cover methods that address the numerous issues that have been imposing the gap between artificial conversational agents and human-level interlocutors. These methods were proposed and experimented with in ways that were inspired by general state-of-the-art AI methodologies at their respective time periods. But they also targeted the characteristics that dialogue systems possess. In the rest of this chapter, we first illustrate the task definitions we use in this thesis, from various angles including utility, modality and response production method. We then illustrate the backbone models that were used, such as LSTM and GPT. After that, we briefly look back at the literature from the standpoint of 2018 (the beginning of my PhD). We end with the overall organization of the thesis.

1.1 Task Definitions

Simply put, a dialogue system is a model that maps a conversational context C to a response R. There are various types of dialogue systems depending on the utility, modality, response production method. The main content chapters (2-5) of this thesis are based on various types of dialogue systems. Therefore, to begin with, we illustrate their definitions here.

1.1.1 Utility

Utility, or dialogue mode, refers to the function that a dialogue system serves. They can be classified into task-oriented dialogue (TOD) systems and open-domain dialogue (ODD) systems. The former serve the utility of performing task-specific functions, while the latter focus on non-goal-oriented chitchat.

ODDs generate the response based on any open-domain context and exhibit general chitchat ability. Their primary goal in a conversation is to keep the user engaged and chat over random open-domain topics that he is interested in. For example, Apple's Siri can chat about your day with you. The dialogues can be sustained by commonsense and empathy without the need for any special databases. TOD models are vastly different. The dialogues exist for the purpose of serving specific functions, such as finding restaurants and booking airlines. They operate on closed domains that are often supported by structured databases and APIs.

Under the scope of this thesis, three characteristics distinguish TODs from ODDs:

(1) TODs have an entity-centered database. This means the the response R is dependent on the database D in additional to the conversational context C. For example, a restaurant reservation bot is dependent on a database of restaurants, containing information such as their names, offerings, etc.

(2) TODs explicitly predict the user's intent in order to query the database. For example, in order to retrieve the correct restaurant, the user's exact preference on price range and cuisine type needs to be explicitly inferred.

(3) Since TODs are dependent on knowledge bases, they usually have a pre-defined set of dialogue domains and functions.

In this thesis, "inter-mode" or "multi-mode" dialogue systems refer to models that can fuse both utilities in the same conversation session.

1.1.2 Modality

While most dialogue systems assume textual input (sometimes converted from speech) and output, human conversation is inherently multi-modal. Humans understand each other through audio and video signals that are sometimes beyond what text can convey.

In multimodal dialogue systems, the context C and the response R are potentially multimodal. For example, if the user inputs his message using a microphone, C might contain C_{text}, which is the textual message, and C_{audio}, which is the audio clip. Similarly, R could be a short video clip of a digital avatar, which would naturally contain R_{audio}, i.e., the audio clip and R_{video}, i.e., the facial expression and the body gesture of the avatar.

1.1.3 Response Production Method

Depending on how the response R is generated, dialogue systems can be classified into generation-based and retrieval-based. The former models attempt to generate brand-new responses from scratch. The latter models select the most appropriate response from a pre-constructed response candidate set.

In other words, a generation-based dialogue system learns a sequence transduction model $R = f_1(C)$. The model generates a response based solely on the context. A retrieval-based dialogue system learns a scoring model $f_2(C, R)$ that computes the compatibility of a context and a response. In addition, the system has access to a large response repository $\mathcal{R}$ from where it chooses the best response for each context.

1.2 Backbone Models

Backbone models such as long short-term memory networks (LSTM) and Transformers have been based upon by many state-of-the-art models in various NLP tasks in the recent years. Our early work in Chapter 3 and 4 is based upon LSTMs. Our later work in Chapter 2 and 5 is based upon transformers.

1.2.1 LSTM

As a version of recurrent neural network, an LSTM network (Hochreiter and Schmidhuber, 1997) is good at handling long-term dependencies and can be used to map an utterance to hidden states as fixed-size embedding representations, based on which decoding can be done to predict the next tokens.

The kth token in an utterance is first embedded into a vector e_k of dimension d using a word embedding matrix. Then, the hidden representation h_k at time step k for the utterance is defined by:

$$\begin{aligned}
i_k &= \sigma(W_i \cdot [h_{k-1}, e_k]) \\
f_k &= \sigma(W_f \cdot [h_{k-1}, e_k]) \\
o_k &= \sigma(W_o \cdot [h_{k-1}, e_k]) \\
l_k &= tanh(W_l \cdot [h_{k-1}, e_k]) \\
c_k &= f_k \cdot c_{k-1} + i_k \cdot l_k \\
h_k &= o_k \cdot tanh(c_k)
\end{aligned} \tag{1.1}$$

where W_i, W_f, W_o, $W_l \in \mathcal{R}^{D \times (D+d)}$. An input gate, a memory gate and an output gate, denoted as i_k, f_k and o_k, are used to update cell state c_k and hidden state h_k iteratively. D is the dimension of hidden state h_k. σ denotes the sigmoid function.

For the mainstream community, deep LSTMs were considered the state-of-the-art models for many NLP tasks up until the rise of transformers. LSTMs were used as backbone models for our work in Chapters 3 and 4.

1.2.2 Transformer

Vaswani et al. (2017) brought forward a type of neural networks that processes tokens in a sequence in a parallel manner using Self-Attention.

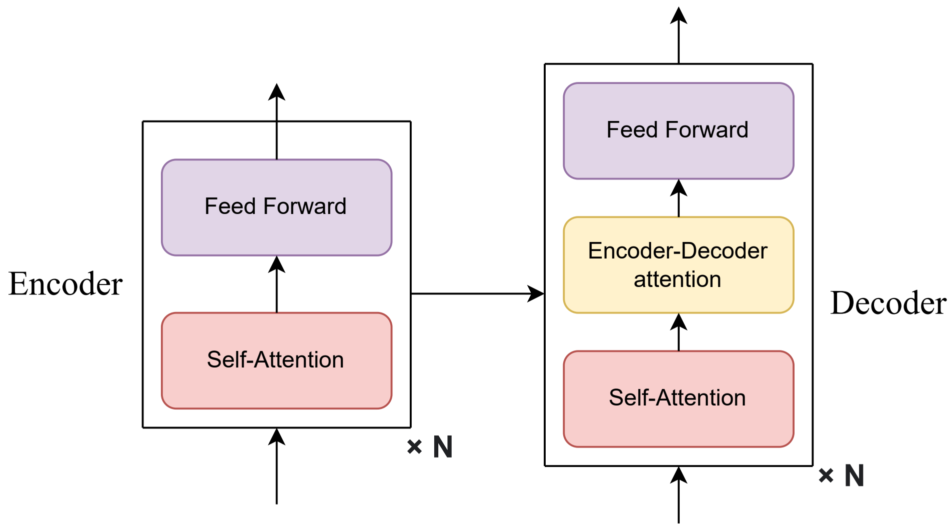


FIGURE 1.1: Transformers used for Seq2Seq learning as defined in Vaswani et al. (2017).

Self-Attention involves three vectors:

Query: The query is a representation of the current word used to score against all the other words (using their keys).

Key: The key vectors are matched against queries to determine the importance of the words.

Value: Value vectors are "actual" word representations. Once how relevant each word is is determined, the values are added up to represent the current word.

We use Q, K, V to represent the query, key, value matrices (vectors from multiple tokens condensed together). They are obtained by multiplying the input matrix X with the transformation matrices W^Q, W^K, W^V.

The Self-Attention mechanism can be represented as:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V, \tag{1.2}$$

where d_k is the dimension of the key vector. Instead of performing a single attention function, it is found beneficial to use Multi-Head attention with h parallel attention functions and then concatenate the output.

$$MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^O \tag{1.3}$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V), i \in [0, h] \tag{1.4}$$

[1] used Self-Attention in both the encoder and the decoder in Seq2Seq learning tasks. In the decoder stack, the Encoder-Decoder attention module is used, which is the same as the Self-Attention module except the the query vectors are from the decoder and the key and value vectors are from the encoder.[1]

One key difference between the encoder and the decoder is attention masking. In the encoder, the encoded sequence is fully given and visible, therefore each token can attend to every other token to the left and to the right. In the decoder, however, the training objective is to predict the following tokens given previous tokens. Therefore each token can attend to the tokens to the left.

Transformers are used in most state-of-the-art NLP models at the moment. For example, when used for large-scale pretraining, they become powerful multi-task learners Radford et al. (2019), as demonstrated in the next two sections.

1.2.3 BERT

The Bidirectional Encoder Representations from Transformers (BERT, Devlin et al., 2018) model is designed to pre-train bidirectional representations from raw text by conditioning on bidirectional context. The pre-trained BERT model can be fine-tuned to create state-of-the-art models for a wide range of tasks, such as question answering and natural language inference, without substantial task-specific architecture modifications.

[1]The queries come from the previous decoder layer, and the memory keys and values come from the output of the encoder. This allows every position in the decoder to attend over all positions in the input sequence. Also termed “cross-attention” (Vaswani et al., 2017).

BERT creates powerful representations for words and sentences that are useful for sentence-level and token-level tasks that be regarded as classification tasks. It is not explicitly created to generate sequences and therefore BERT only contains the encoder part as mentioned in the last section.

BERT is pretrained on two tasks: (1) the Masked Language Modeling task which predicts the masked token based on bidirectional context, and (2) the Next Sentence Prediction task which predicts if one sentence naturally follows another.

The first token of every sequence is always a special classification token ([CLS]). The final hidden state corresponding to this token is used as the aggregated sequence representation for downstream tasks. BERT can naturally generate representation for a single sentence or a sentence pair. For example, a BERT model can be used to encode a context and a response separately. Afterwards a scoring model can use the resulting two representations to calculate their compatibility, through, e.g., inner-product. Alternatively, a BERT model can be used to encode the concatenation of a context and a response as one sequence. The single resulting representation can be used to score how plausible the exchange is, through, e.g., a feed-forward neural network.

In Chapter 5, we explore methods that accelerate retrieval-based dialogue systems based on BERT representations.

1.2.4 GPT

Generative Pretrained Transformers (GPT) are another class of models that proved the effectiveness of large-scale pretraining. In particular, they excel at language generation tasks.

Unlike BERT, which seeks to learn bi-directional representations of words and sentences, GPTs are unidirectional language models which simply learns to predict the next token given previous tokens. GPTs can be used to generate sequences in a straightforward manner using beam search. To achieve this property, GPTs use the decoder architecture as mentioned in Section 1.2.2 without Encoder-Decoder attention, since no input and output sequences are defined during pretraining.

In Chapter 2, we utilise GPTs as the backbone for end-to-end TOD models and inter-mode dialogue models.

1.3 Early Literature

This chapter briefly reviews the literature that motivated the research works in this thesis from the standpoint of the beginning of my PhD (2018).

• *It should be noted that dialogue systems have always been a hot area of research, so the literature landscape has been constantly evolving. The background that one research work had to consider was often vastly different from another just 1 year or a few months later. Therefore, each chapter also includes its own Related Work section that reviews the literature landscape that motivated each individual research work from its own time point.*

• *It's also worth noting that the latest research work covered by this thesis was done in late 2021. For readers interested in the literature landscape from the standpoint of 2023, please refer to Chapter 6: Summary and future directions.*

At the beginning of my PhD, the stage had been set for data-driven dialogue systems to bloom thanks to drastic increase in compute and online text data.

As early as 2011, Ritter et al. (2011) marked a significant shift from rule-based systems by introducing a data-driven approach to generate responses in social media using statistical machine translation techniques. This new paradigm laid the groundwork for subsequent advancements, such as the sequence-to-sequence learning framework proposed by Sutskever et al. (2014), which transformed the input sequence into a fixed-sized vector and then generated an output sequence from that vector. This framework was proposed for machine translation but soon became a cornerstone for later dialogue system models.

Vinyals and Le (2015) further built on this foundation by leveraging sequence-to-sequence learning in their neural conversational model, opening new possibilities for open-domain dialogue systems. Around the same time, Serban et al. (2016) presented a generative hierarchical neural network model that could effectively handle long context and generate coherent responses. To address the problem of dull and repetitive replies in data-driven dialogue systems, Li et al. (2015) introduced a diversity-promoting objective function, which encouraged the exploration of different responses in neural conversation models.

Advance based on the data-driven paradigm started out for open-domain dialogues but soon found its way into task-oriented dialogue systems. Bordes and Weston (2016) proposed a memory network-based model and introduced the bAbI dialogue dataset, demonstrating the effectiveness of their model in learning to complete tasks through user conversations. Building on this Zhu et al. (2017) presented a novel end-to-end trainable task-oriented dialogue system could handle complex dialogue management and natural language understanding tasks, outperforming traditional pipeline-based approaches.

Despite that the data-driven framework had been mainstream by 2018 for the field of dialogue systems, naively training the model on context-response pairs have many limitations. Attempting to solve different limitations that dialogue systems faced one by one using different datasets and the latest methodologies was the meta-topic of my PhD (and thus this thesis).

While early dialogue systems were either tuned for task-oriented dialogues or open-domain chitchat independently, it is more desirable for conversational agents to integrate the two abilities because it would make them more accessible and useful. Along the direction of combining TODs and ODDs, early efforts were made when Zhao et al. (2017) proposed to artificially augment TODs with randomly sampled utterances from a chitchat corpus, mainly to improve the out-of-domain recovery performance for the TOD system. Chapter 2 touched on the aspect of utility fusion.

Early dialogue systems focused on text as the only modality, which made them fall short compared to humans since we can condition our words on multimodal context. Along the direction of conditioning dialogue systems on more modalities, Mostafazadeh et al. (2017) proposed image-grounded conversation, where two interlocutors generate conversations based on a shared image. On the user side, Yu (2015) proposed to model user engagement and attention in real time by leveraging multimodal human behaviors, such as smiles and speech volume. Our method of word-level modality fusion proposed in Chapter 3 had already seen use in multimodal sentiment analysis. In Chen et al. (2017), the RNN, which acts as the utterance encoder, takes a concatenation of audio, video and text features as input at every time step. Chapter 3 touched on this aspect of considering more modalities.

While training the model on context-response pairs enabled it to grasp the concept of a conversation, in practice, it is often useful to provide the model with additional information relevant to the context . On the front of conditioning NLP systems on external knowledge, the use of an external memory module had gained considerable attention, such as in question answering (Weston et al., 2015) and language modeling. Memory Networks had been employed in dialogue modeling in limited settings, like in movie-based conversations (Dodge et al., 2015), which used a set of fact triples about movies as long-term memory. In a restaurant reservation context, researchers provided local restaurant information to the conversational model (Bordes and Weston, 2016). To incorporate knowledge as external memory into the Seq2Seq framework, several methods had been proposed. One approach involved incorporating the topic words of the message obtained from a pre-trained LDA model into the context vector through a joint attention mechanism (Ghazvininejad et al., 2017). Another method encoded unstructured textual knowledge with RNNs (Lowe et al., 2015). Chapter 4 touched on the aspect of external knowledge grounding.

The practical usability of a dialogue system is determined by the quality of the response and the computational cost of producing one. On the front of improving the inference efficiency of neural networks (thus dialogue systems), distilling knowledge from a high-accuracy network into a low-accuracy network (Hinton et al., 2015) had proven to be an effective way to improve the accuracy of the latter. Traditionally the latter (student network) tries to mimic the former (teacher network) by minimizing a loss defined between the outputs of the two, in addition to the traditional loss based on groundtruth labels. As the multi-class output of the teacher network has higher entropy than the traditional one-hot labels, the student network has access to an information-rich similarity structure over data. The student network *usually* has the advantage of being smaller, which makes training and inference faster. Research in various areas had shown the effectiveness of this approach. Kim and Rush (2016) successfully boosted the inference speed of state-of-the-art machine translation networks by about 10 times with little loss in performance. It also found success in computer vision areas such as object detection (Chen et al., 2017). Chapter 5 touched on the aspect of acceleration.

1.4 Organization

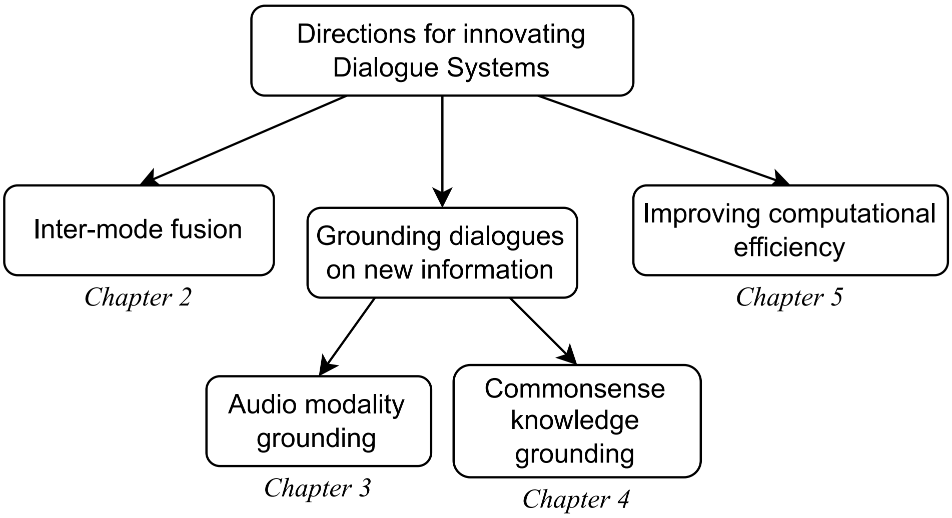


FIGURE 1.2: Thesis structure diagram. Our thesis contains 4 main content chapters. We touch on their similarities and dissimilarities in Table 1.1 and discuss their interconnections more in the chapters.

TABLE 1.1: A chapter in the thesis is defined by at least 6 variables as shown below (from Utility down to Acceleration tricks). All possible combinations of the different values that the variables can take can add to 100+. Each research work shown in a particular chapter in this thesis chose a particular combination due to reasons such as dataset availability and literature landscape at that time, which are explained in each chapter. For a holistic picture of the development of the field of dialogue systems up to 2023 and the chapters' interconnectivity, refer to discussions under the Related Work section under each chapter.

Thesis chapter	Chapter 2	Chapter 3	Chapter 4	Chapter 5
Year of Conduction	2021	2019	2018	2020
Utility	TOD+ODD	ODD	ODD	ODD
Modality	Text	Audio+Text	Text	Text
Response production method	Generation	Generation	Retrieval	Retrieval
Backbone models	GPT2	LSTM	LSTM	LSTM, BERT
Commonsense grounding	No	No	Yes	No
Acceleration tricks	No	No	No	Yes

In this section, we provide an outline of the rest of the thesis. The primary contributions of this thesis are the four main content chapters: Chapters 2-5 (See the overall concept structure in Figure 1.2). The outlines of the chapters are as follows:

Chapter 2: Bridging the gap between task-oriented and open-domain dialogues

The work covered in this chapter focuses on the research question: "How would one effectively construct a dialogue dataset fusing TODs and ODDs and what are some basic methods that can be designed to handle inter-mode dialogues based on pretrained language models?" In this chapter, we illustrate our effort in 2021 on building dialogue systems that seamless fuse the utilities of performing task-oriented dialogues and open-domain chitchat. We detail our effort on expanding the frequently used MultiWOZ dataset into the new FusedChat Dataset - a new dialogue dataset on which inter-mode dialogue systems can be tested. We also evaluate two baseline approaches on this new dataset.

Chapter 3: Grounding dialogues on non-textual modalities

This chapter showcases our effort in 2019 on conditioning the dialogue systems on the audio modality in addition to text. The research question being addressed here is "how can audio features be effectively integrated into the popular Seq2Seq framework?". We first propose an auxiliary response selection task to learn suitable audio representations from raw noisy audio features. We then propose an Audio-Seq2Seq framework which concatenates audio features to traditional word embeddings. Our audio-augmented model outperforms its audio-free counterpart on perplexity, response diversity and human evaluation.

Chapter 4: Grounding dialogues on commonsense knowledge

The work covered in this chapter focuses on the research question: "What are the effective ways of incorporating commonsense knowledge and how may it improve dialogue responses?" In this chapter, we demonstrate our effort in 2018 on augmenting dialogue systems with a large structured commonsense knowledge database. In the retrieval-based scenario, we propose the Tri-LSTM model to jointly take into account the context and commonsense for selecting an appropriate response. Experiments suggest that the knowledge-augmented models are superior to their knowledge-free counterparts in terms of retrieval accuracy.

Chapter 5: Improving the computational efficiency of large-scale response retrieval

Our last main content chapter targets a computational efficiency problem that our effort in 2020 aimed to address: "how can we accelerate large-scale response retrieval?". In this chapter, we present methods to improve the inference speed of a large-scale response retrieval model. We used knowledge distillation to leverage the learning power of a cumbersome joint encoding model to improve the performance of our fast individual encoders. Furthermore, to better handle large response candidate sets, we propose a learning-based screening model that makes the retrieval process about 5 times faster with very little accuracy loss. Finally, we demonstrate a pipeline that performs strongly in terms of speed and quality trade-off compared to other retrieval-based models .

The chronological order of the works covered in this thesis corresponding to the 4 chapters is Chapter 4 <year 2018>, Chapter 3 <year 2019>, Chapter 5 <year 2020>, and Chapter 2 <year 2021>. In this thesis, I ordered the chapters by their relevance from the perspective of 2023.

Each research work in the first 3 chapters can be individually seen as an initiative for a new subdirection for the field of dialogue systems as opposed to an iteration along an existing direction. One may notice that certain design choices such as backbone model often do not transfer from one chapter to the next (Table 1.1). This is due to reasons such as literature landscape changes across the years and dataset availability. While our works in the thesis cover multiple aspects on which dialogue systems can improve, it doesn't often cover the scenarios where they interplay, which are covered in discussions under "Related Work". The main contribution of the works covered in the thesis lies in their initializing effects (to a certain degree) on these directions that have been continuously worked on by researchers till this day.

Chapter 6: Summary and future directions

We conclude this thesis by summarizing our contributions, analyzing the role of dialogue systems in AI, illustrating recent advances brought forward by ChatGPT (OpenAI, 2022) and GPT4 (OpenAI, 2023), and conjecturing future directions.

Chapter 2

Bridging the gap between task-oriented and open-domain dialogues

The goal of building intelligent dialogue systems has largely been *separately* pursued under two paradigms: task-oriented dialogue (TOD) systems, which perform task-specific functions, and open-domain dialogue (ODD) systems, which focus on non-goal-oriented chitchat. The two dialogue modes can potentially be intertwined together seamlessly in the same conversation, as easily done by a friendly human assistant. Such ability is desirable in conversational agents, as the integration makes them more accessible and useful. This chapter addresses the problem of fusing TODs and ODDs in multi-turn dialogues. The work covered in this chapter focuses on the research question: "How would one effectively construct a dialogue dataset fusing TODs and ODDs and what are some basic methods that can be designed to handle inter-mode dialogues based on pretrained language models?" Based on the popular TOD dataset MultiWOZ, we build a new dataset FusedChat, by rewriting the existing TOD turns and adding new ODD turns. This procedure constructs conversation sessions containing exchanges from both dialogue modes. It features inter-mode contextual dependency, i.e., the dialogue turns from the two modes depend on each other. Rich dependency patterns such as co-reference and ellipsis are included. The new dataset, with 60k new human-written ODD turns and 5k re-written TOD turns, offers a benchmark to test a dialogue model's ability to perform inter-mode conversations. This is a more challenging task since the model has to determine the appropriate dialogue mode and generate the response based on the inter-mode context. But such models would better mimic human-level conversation capabilities. We propose and evaluate two models on this task, including the *classification-based* two-stage models and the *two-in-one* fused models.[1]

2.1 Introduction

According to their utility, two mainstream types of dialogue models can be categorized as ODD models (Adiwardana et al., 2020; Roller et al., 2020; Zhang et al., 2019), and the TOD models (Ham et al., 2020; Budzianowski et al., 2018). ODD models generate the response based on the context and exhibit general chitchat ability. Their primary goal in a conversation is to keep the user engaged and chat over random open-domain topics that he is interested in. The dialogues can be sustained by commonsense without the need for any special databases. TOD models are vastly different. The dialogues exist for the purpose of serving specific functions, such as finding restaurants and booking airlines. They operate on closed domains that are often supported by structured databases and APIs (Budzianowski et al., 2018; Rastogi et al., 2020). Commonly three characteristics distinguish them from ODD models: (1) an entity-centered database, (2) explicit dialogue state modeling, and (3) a pre-defined set of dialogue domains and functions (dialogue acts). Humans are able to effortlessly conduct both types of conversations seamlessly together. It is ideal for a dialogue system to be able to do so, because such integration offers a fused system with increased usability. Furthermore, it allows rich interactions between the two dialogue modes, which can not be modeled in either mode independently. Such a dialogue model would better mimic human-level conversation capabilities, e.g., chatting with a friendly assistant (Figure 2.1).

Despite that numerous datasets have been created in recent years for both ODDs and TODs, there is no high-quality human-written dataset on their fusion, especially with inter-mode contextual dependency. Our work aims to fill this void. We use the popular TOD dataset MultiWOZ (Budzianowski et al., 2018) as the backbone and let human creators add ODD turns before or after the existing TOD turns. For roughly half the MultiWOZ dialogues, we prepend ODD turns, creating ODD + TOD sessions. For the other half, we append ODD turns, creating TOD

[1] The work in this chapter has been published in Young et al. (2022).

+ ODD sessions. In both cases, the creator writes an ODD that is contextually related to the existing TOD. We enforce inter-mode dependency in FusedChat. In the prepending case, we make sure the TOD is dependent on the ODD by rewriting the first turn of the TOD, typically with co-reference or ellipsis. In the appending cases, we make sure at least one exchange in the ODD is dependent on concepts or knowledge found in the TOD. In a nutshell, these dependency patterns in our dataset mean that when a dialogue model handles a turn of one dialogue mode, it sometimes has to refer to the contextual information given in the history turns of the other dialogue mode.

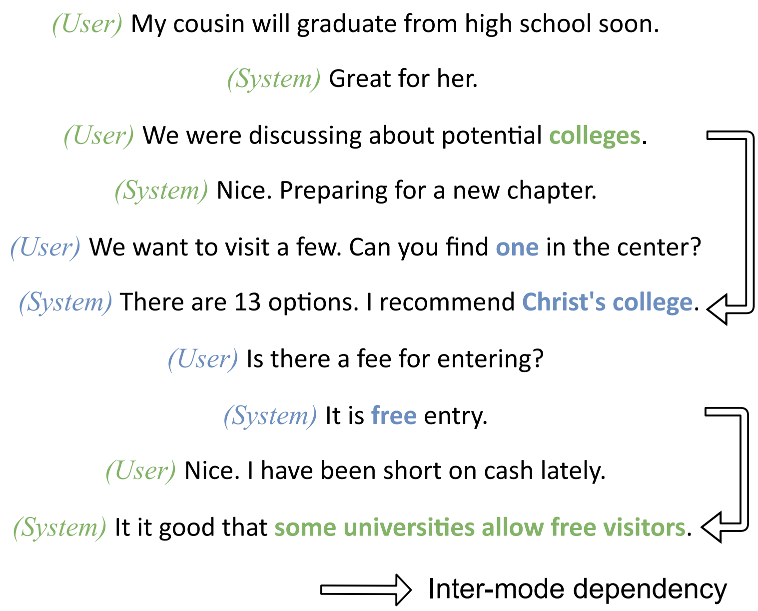


FIGURE 2.1: Example of interaction with our dialogue system. The conversation between a user and a digital assistant seamlessly interchanges between TOD and ODD modes with strong inter-mode dependency. The conversation involves querying about a college entrance fee (TOD, lines 5 to 8) and chitchat about personal development and finance (ODD, the other lines).

This new dataset offers a unique test-bed for training and evaluating inter-mode dialogue systems that possess both TOD and ODD capabilities. Traditional dialogue evaluation metrics for both dialogue modes can be used together for inter-mode evaluation.

We develop and evaluate two baseline models for this new setting: (1) The *classification-based* model. Two response generation models $\mathcal{M}_{tod}$ and $\mathcal{M}_{odd}$ are independently trained on the turns of the respective modes. They generate the response of their respective mode given a conversational context. A separate mode classification model $\mathcal{C}$ is trained and used to determine which mode to invoke given the context. (2) The *two-in-one* fused dialogue model that is trained on dialogue turns of both modes together. It generates a response given any conversational context, by implicitly predicting the dialogue mode as part of sequence generation.

In summary, the main contributions[2] that this chapter covers are: (1) A new dialogue dataset named FusedChat that fuses TODs and ODDs in multi-turn dialogues. The dialogues feature inter-mode contextual dependency for seamless mode fusion, allowing the dialogue model to better mimic human-level conversation capabilities. FusedChat, with 60k new human-written ODD turns and 5k re-written TOD turns, serves as a new benchmark for inter-mode dialogue systems. Traditional metrics used to gauge TOD and ODD systems separately can be combined to evaluate inter-mode dialogue systems. (2) *Two-in-one* models and *classification-based* models are developed and evaluated as inter-mode dialogue models. Our preliminary experiments suggest that the models evaluated on FusedChat perform worse than their single-mode counterparts evaluated on single-mode datasets. And the more computationally expensive *classification-based* model outperforms the cheaper *two-in-one* fused model. This suggests that effectively fusing different dialogue modes is a challenging task for future work.

2.2 Proposed Dataset

2.2.1 FusedChat Construction

To create inter-mode dialogue sessions, our dataset construction process mainly involves having dialogue creators prepend or append self-written ODDs to existing TODs. A dialogue creator plays the part of both the user and the dialogue system by himself. This self-dialogue setting (Byrne et al., 2019) avoids misunderstandings between two human creators and improve the consistency of the created dialogues.

For the existing TODs, the MultiWOZ 2.4 dataset (Ye et al., 2021) is selected because of its popularity in the literature. MultiWOZ contains TODs in 7 domains, including restaurant, attraction, train, police, hospital, taxi and hotel. The user

[2] `https://github.com/tomyoung903/FusedChat`

converses with the dialogue agent for a pre-defined set of functions, such as booking restaurants and locating hospitals. Despite that MultiWOZ was created assuming the user is a tourist (Budzianowski et al., 2018), we find that most dialogues themselves do not necessarily reflect a tourist persona and allow flexibly adding ODDs. In our FusedChat setting, the dialogue creators are free to add any ODD that is contextually consistent with the existing TOD.

In the following sections, we first discuss the general requirement we set for the added ODDs. We then explain how prepending and appending ODDs are executed and how inter-mode dependency is enforced, respectively.

2.2.1.1 General Requirements for the Added ODDs

In this section, we describe the general requirements for the added ODDs for both the prepending and appending cases, as rules for the dialogue creators to follow.

(1) Every creator writes fictitious ODDs for *both* the roles of "system" and "user", where the "system" represents an AI conversational agent that is capable of both friendly open-domain conversation (in the added ODDs) and task-oriented dialogues (in the existing MultiWOZ TODs). And "user" represents a human speaker that converses with the AI agent for friendly chitchat and to achieve certain task objectives.

(2) To ensure the relevance between the existing TOD and the added ODD, we encourage the creators to make the ODD revolve around similar or related topics as in the existing TOD segment, e.g., by talking about the same or related concepts in the TOD. The added ODD turns and the existing TOD turns should connect with each other naturally. There should be strong contextual dependency between the two modes (explained in the next 2 sections).

(3) The created dialogues should adhere to the general characteristics of ODDs as opposed to TODs. They should be casual chitchat exchanges that do not require the "system" to perform any specific task-oriented functionalities or provide any task-specific information.

- Based on the pilot experiment with a sample of creators, we found that the creators had a tendency to write dialogues that are focused on task-specific functionalities, which are technically TODs instead of ODDs as instructed. This is presumably because of a lack of nuanced understanding of their difference, and the ease of fitting those TODs into the context of existing TODs. As an aggressive measure to combat this issue, we deployed a real-time turn-level ODD vs TOD classifier, trained on a combination of three traditional ODD datasets (Zhang et al., 2018; Smith et al., 2020; Dinan et al., 2018) and MultiWOZ. In addition, we outline several pitfalls found in the pilot experiment for the creators to avoid, such as letting the system fabricate information that is beyond commonsense.

Next, we describe the details on how appending ODDs (TOD + ODD) and prepending ODDs (ODD + TOD) are executed, and how inter-mode dependency is enforced, respectively.

2.2.1.2 Appending ODDs

In the appending scenario, the dialogue creators append an ODD to a provided TOD sampled from the MultiWOZ dataset. The ODD should naturally *follow* the TOD.

- We notice that the dialogues from the original MultiWOZ dataset often end with a "User: Thank you. System: Goodbye." exchange. This exchange effectively *ends* the conversation. For appending ODDs, we heuristically remove such exchanges from the end of the TOD based on dialogue act annotations (dialogue-act:thank-you and dialogue-act:goodbye).

In appending cases, the content of the ODD should be dependent on the preceding TOD. We enforce this by letting the creators write at least one round of exchange whose content reflects concepts or knowledge found the existing TOD segment.

2.2.1.3 Prepending ODDs

In prepending cases, the creator is given a TOD segment from MultiWOZ and asked to prepend an ODD to it. The ODD should naturally *lead to* the provided TOD.

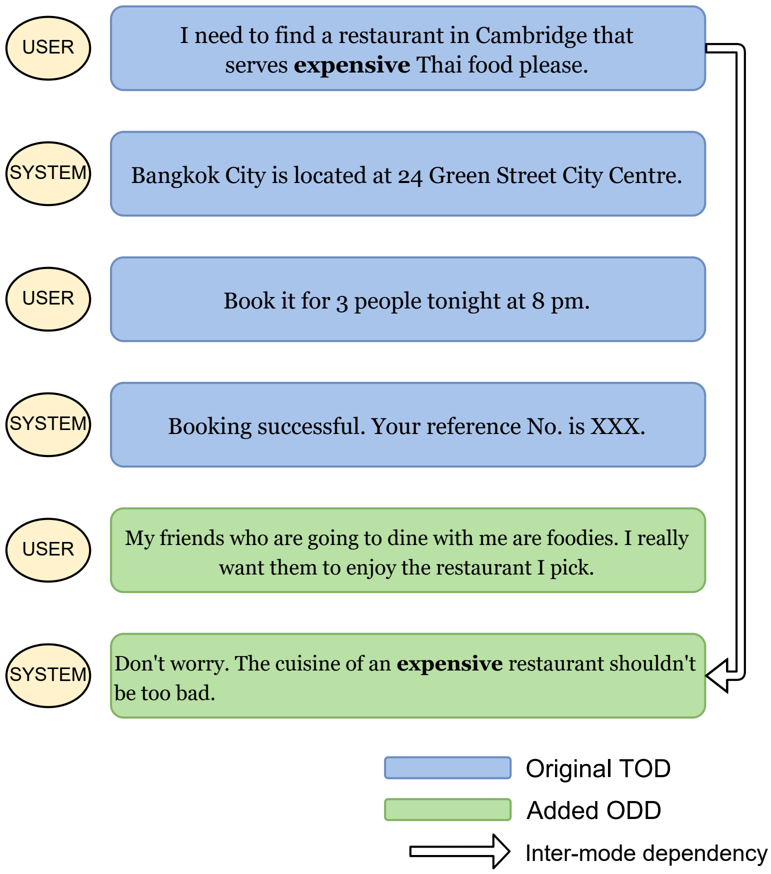


FIGURE 2.2: An excerpt from a TOD + ODD instance from FusedChat. Note how inter-mode dependency is featured in the last system ODD turn by referring to the concept "expensive restaurant" previously mentioned in the TOD.

Note that the original TODs in MultiWOZ are self-contained. For our purpose of modeling inter-mode dependency, we conduct utterance rewriting based on co-reference and ellipsis. In FusedChat, they are the key why the TOD is dependent on the prepended ODD.

We want to create ODD + TOD sessions where the TOD is conditioned on the ODD. The key to a successful TOD is dialogue state tracking, where the dialogue system processes the user utterance for [slot type, slot value] pairs (e.g., [Destination: Cambridge]) in order to understand the user's need and respond properly. Our designed method to model inter-mode dependency in our dataset essentially imposes ODD-dependent dialogue state tracking.

We randomly select a slot value mentioned in the first user turn in the TOD, e.g., "Cambridge" in Figure 2.3. We ask the dialogue creators to use the slot value in the prepended ODD, and rewrite the first dialogue user turn accordingly to refer to it implicitly. Rewriting mainly involves co-reference (e.g., "there" in Figure 2.3), and sometimes ellipsis. Co-reference and ellipsis are important features in multi-turn TODs, attracting researchers to sometimes perform special annotations for them in certain TOD datasets (Quan et al., 2020). See Figure 2.3 for a detailed example on how inter-mode dependency is featured for ODD + TOD sessions.

2.2.2 FusedChat Statistics

A total of 113 undergraduate students from our university were recruited as dialogue creators for FusedChat. The difference between FusedChat and MultiWOZ mainly lies in the additional ODD turns, either grounding or grounded by the original TODs. The added ODD turns in FusedChat are a significant extension to the original MultiWOZ dataset. As shown in Table 2.1, 60k+ new ODD turns are added, including 8k+ new tokens not present in the original MultiWOZ dataset, significantly expanding the vocabulary.

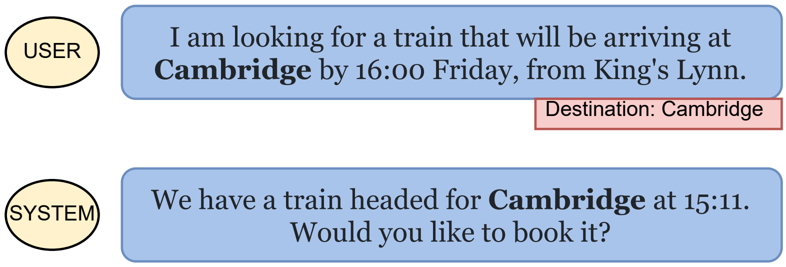
FusedChat also rewrites the first TOD turns (4670 in total) for the scenario of prepending ODDs. For the scenario appending ODDs, FusedChat discards 11320 TOD turns containing only "thank-you" and "goodbye" dialogue acts. Table 2.2 shows the training/validation/testing partitions for FusedChat.

TABLE 2.1: Statistics on the added ODD turns in FusedChat.

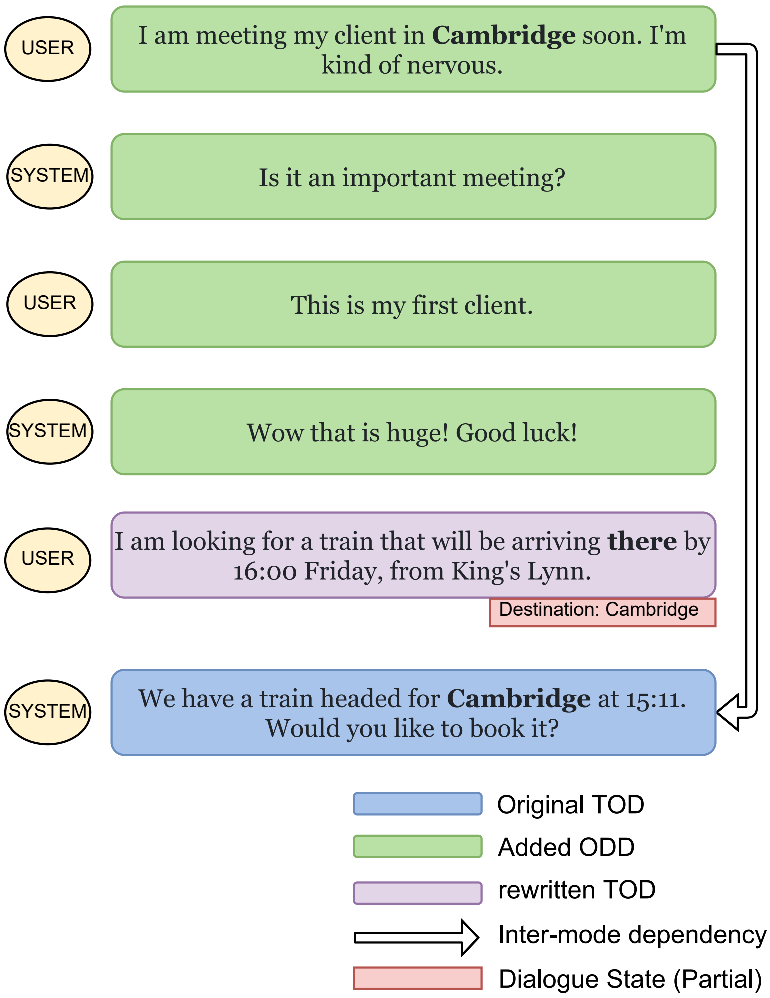
Total No. turns	60579
Total No. tokens	680448
Avg. No. turns per dialogue	5.81
Avg. No. tokens per turn	11.23
No. unique tokens	11822
No. unique tokens not present in MultiWOZ	8075

2.3 Proposed Approaches

In this section, we discuss baseline models we developed for inter-mode dialogues.



(A) An original TOD exchange with the dialogue state [Destination: Cambridge].



(B) New ODD turns are prepended to the original TOD in (A). Note that the TOD user turn is rewritten. The slot value "Cambridge" is mentioned in a prepended ODD turn while co-reference is used in the rewritten user turn. This imposes ODD-dependent dialogue state tracking, forcing the the dialogue system to look for clues in the ODD when it tries to interpret the user's need.

FIGURE 2.3: An ODD + TOD instance from FusedChat.

Table 2.2: FusedChat is composed of ODD + TOD (prepending ODDs) instances and TOD + ODD (appending ODDs) instances.

Partition	ODD + TOD	TOD + ODD	Total
Training	3670	4768	8438
Validation	500	500	1000
Testing	500	500	1000
Total	4670	5768	10438

2.3.1 Task Definition

A multi-turn dialogue system generates a response R based on a multi-turn context C. In inter-mode dialogues, C is composed of both TOD and ODD turns. In the FusedChat setting, R can be in either TOD mode or ODD mode, but has to be in only one of the two.

2.3.2 Models

We experiment with two types of models for inter-mode dialogues. (1) The *classification-based* model that is composed of a mode classification model and two response generation models for TOD and ODD separately and (2) the *two-in-one* fused model where a single response generation model can perform both TOD and ODD generation, implicitly determining the response mode. It is natural to consider two settings because whether or not to model mode classification implicitly is an important design choice that impacts the number of models (thus scale) of the inference pipeline.

(1) The *classification-based* model. Two response generation models $\mathcal{M}_{odd}$ and $\mathcal{M}_{tod}$ are independently trained to handle each conversation mode. A separate classification model $\mathcal{C}$ is trained and used to determine which mode of model to invoke given an inter-mode context. Note that all 3 models above take inter-mode context as input.

- For $\mathcal{M}_{odd}$, we follow Shuster et al. (2019) and experiment with DialoGPT (Zhang et al., 2019) as the pretrained model, fine-tuned on all ODD turns in FusedChat.

- For $\mathcal{M}_{tod}$, we follow the recent progress on end-to-end modeling for TODs. Dialogue state tracking, dialogue act prediction and response generation have been together cast under a Seq2Seq framework (Hosseini-Asl et al., 2020, Ham et al., 2020). For traditional Seq2Seq-based ODD modeling, the problem is cast as [Context $\rightarrow$ Response]. For Seq2Seq-based TOD modeling, the problem is cast as [Context $\rightarrow$ (Dialogue State, Dialogue Act, Response)], where the three latter parts are concatenated together as one sequence as the generation target. This simplistic form allows TOD models to exploit the benefits of large-scale pretrained models, same as ODD models did. We follow *Neural Pipeline* (Ham et al., 2020) for such a model for $\mathcal{M}_{tod}$, initialized with GPT2 (Radford et al., 2019) and fine-tuned on all TOD turns in FusedChat.

- For $\mathcal{C}$, we follow Madotto et al. (2020) and experiment with BERT (Devlin et al., 2018) as the pretrained model. The model is fine-tuned on all turns in FusedChat to predict the dialogue mode (TOD vs ODD).

(2) The *two-in-one* model. Trained on dialogue turns of both modes, it uses a single model that generates a response given any conversational context by implicitly determining the conversational mode. Similar to Sun et al. (2020), we use an additional <ODD> token during sequence construction to indicate when the response is in the ODD mode. The training sequences are composed of [Context $\rightarrow$ (<ODD>, Response)] and [Context $\rightarrow$ (Dialogue State, Dialogue Act, Response)]. The model is initialized with GPT2 and fine-tuned on all dialogue turns in FusedChat.

For all the models above, best checkpoints for testing are selected based on the full validation set of 1000 instances.

We chose to develop and evaluate two baseline models, the two-in-one model and the classification-based model, for several reasons. Firstly, these models represent two different approaches to handling the inter-mode dialogues present in the FusedChat dataset. The two-in-one model combines both ODD and TOD components into a single model, while the classification-based model separates the two components and uses a mode classifier to determine which component should be used for a given dialogue turn. The two models are straightforward choices given the available paradigm most recent to this work (Ham et al., 2020). They represent different choices of sparsity and cumbersomeness.

Secondly, these models serve as a starting point for future research on inter-mode dialogue systems. By providing baseline results, we establish a reference point for evaluating the performance of more advanced models that may be developed in the future. Additionally, comparing the performance of these two models can provide insights into the most effective strategies for handling inter-mode dialogues, which can inform the design of future models.

2.4 Experiments and Results

Depending on the context and the dialogue mode, the dialogue turns in our dataset are naturally separated into 4 types in Figure 2.4: vanilla TODs, vanilla ODDs, ODD-grounded TODs and TOD-grounded ODDs. Vanilla refers to the dialogue turns being grounded on context of its own mode only, resembling traditional datasets. The ODD turns in the "prepending ODDs" scenario and TOD turns in the "appending ODDs" scenario are vanilla.

In the following sections, we illustrate 4 unique evaluation scenarios on which FusedChat can benchmark the performance of inter-mode dialogue systems, including mode classification, TOD-grounded ODDs, ODD-grounded TODs and full inter-mode dialogues.

2.4.1 Mode Classification

The most straightforward problem one encounters in inter-mode dialogues is to decide which mode the generated response should be. Should the system respond with friendly chitchat (ODD), or should it try to interpret the user's task-oriented goal and respond with certain dialogue acts (TOD)? The accuracy for the mode classification model is shown in Table 2.3. We consider two context options: using only the latest user turn as the context (single-turn) or using the whole history containing multiple turns as the context (multi-turn). Results show that the accuracy is quite high in both cases, with "multi-turn" marginally outperforming "single-turn".

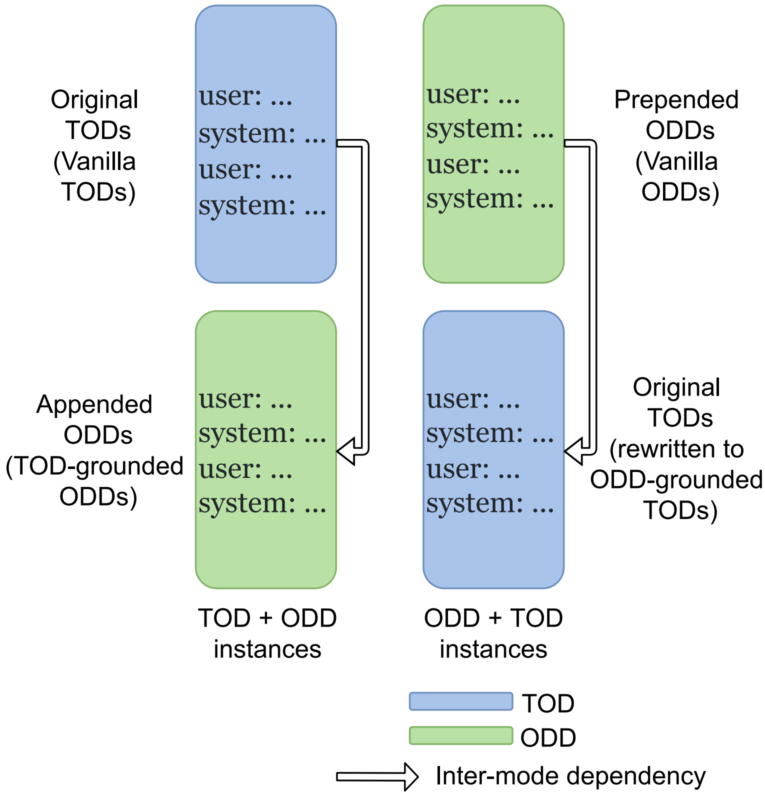


FIGURE 2.4: 4 types of dialogue turns are present in FusedChat, classified by the dialogue mode and the grounding context.

TABLE 2.3: Mode classification accuracy for model $\mathcal{C}$.

Context option	Accuracy
Single-turn	0.993
Multi-turn	0.995

TABLE 2.4: Evaluation results on ODD-grounded TODs in FusedChat and comparison with MultiWOZ results.

Models	Slot Accuracy (SA)	Joint SA	Inform	Success	BLEU
ODD-grounded TODs in FusedChat					
Two-in-one model	0.971	0.574	71.1	56.9	12.16
Classification-based model	0.972	0.584	72.8	60.0	12.58
Original MultiWOZ dataset					
Neural Pipeline [40]	0.976	0.631	79.2	64.3	12.72

2.4.2 ODD-grounded TODs

Part of inter-mode dialogues are ODD-grounded TODs, which correspond to the "prepending ODDs" scenario in FusedChat. Like in regular TODs, the system's response is prompted by a task-oriented user request. However, the preceding context contains ODD exchanges, which create unique challenges.

Table 2.5: Evaluation results on TOD-grounded ODDs in FusedChat.

Models	PPL	Sensibleness	Specificity	SSA
Two-in-one model	9.15	0.44	0.39	0.42
Classification-based model	8.79	0.51	0.45	0.48
Ground-truth	*N/A*	0.97	0.91	0.94

Table 2.6: Evaluation results on the full FusedChat testset.

(a) TOD Metrics

Model	Slot Accuracy	Joint SA	Inform	Success	BLEU
Two-in-one model	0.972	0.592	70.4	57.0	12.05
Classification-based model	0.973	0.600	75.1	60.9	12.17

(b) ODD Metrics

Model	PPL	Sensibleness	Specificity	SSA
Two-in-one model	10.49	0.52	0.47	0.50
Classification-based model	10.50	0.58	0.51	0.55

On the one hand, the model needs to recognize useful task-related information from the ODD context for correct dialogue state tracking. On the other hand, the system's response should correctly perform the required task-oriented function according to the latest user request, instead of derailing to chitchat by following the ODD context in the history.

Evaluation results for this portion of the dialogue turns in FusedChat are shown in Table 2.4. We use the traditional TOD evaluation metrics for MultiWOZ, where slot accuracy measures dialogue state tracking, inform rate and success rate measure user goal success and BLEU (bilingual evaluation understudy) measures response quality.

Slot Accuracy: This metric measures how accurately the dialogue system tracks the state of the conversation, i.e., whether it correctly identifies the values of the slots in the user's utterance. The slot accuracy is calculated as the percentage of slots correctly identified by the system out of the total number of slots in the user's utterance.

Inform Rate: This metric measures the percentage of user goals that are successfully identified by the system. A user goal is considered successfully identified if all its slots are correctly identified by the system.

Success Rate: This metric measures the percentage of dialogues in which the system successfully completes all the user goals. A dialogue is considered successful if all the user goals are achieved by the system.

BLEU: This metric measures the quality of the system's responses in terms of their similarity to the reference responses. BLEU (bilingual evaluation understudy) is a widely used metric for evaluating the quality of machine translations. It calculates a score between 0 and 1, with 1 indicating perfect similarity to the reference response.

In addition, we evaluate the *Neural Pipeline* approach using the original MultiWOZ dataset (trained and tested on MultiWOZ). Remember that the *classification-based* model contains $\mathcal{M}_{tod}$, which exactly follows the *Neural Pipeline* model. This is to evaluate the difficulty of the new ODD-grounded TOD task compared with the vanilla TOD task in MultiWOZ. Table 2.4 shows that:

(1) The *classification-based* model outperforms the *two-in-one* model marginally.

(2) The *Neural Pipeline* model evaluated on the same vanilla TOD dialogues in MultiWOZ significantly outperforms the *classification-based* model evaluated on ODD-grounded TODs in FusedChat. Such significant difference suggests that ODD-grounded TODs are a more challenging task than vanilla TODs. Presumably, this is because (a) the extra requirement to correctly determine the response mode and (b) the extra need for ODD-dependent dialogue state tracking.

2.4.3 TOD-grounded ODDs

Another part of inter-mode dialogues are TOD-grounded OODs, which correspond to the "appending ODDs" scenario in FusedChat. The system's ODD response should be conditioned on both the TOD and ODD turns in the context.

The evaluation on ODD generation is notoriously difficult and numerous evaluation methods have been proposed [5]. In our experiment, we follow [37] and use perplexity plus sensibleness and specificity average (SSA) as metrics. SSA represents the average between sensibleness (*Does the response make sense given the context?*) and specificity (*Is the response specific to the context?*). Both of them are binary for each response. A response can only be deemed specific if it is deemed sensible. SSA results are computed by averaging 5 expert human evaluators' judgement on 100 randomly sampled dialogue turns from the testset. Table 2.5 shows the performance of the inter-mode dialogue models on this task.

The *classification-based* model outperforms the *two-in-one* model marginally. Results also show that ground-truth responses receive very high SSA scores, significantly exceeding the better dialogue model of the two we developed. This suggests that there is huge room for improvement on this task.

2.4.4 Full Inter-mode Dialogues

We show the results on the full FusedChat testset (containing all 4 types of dialogue turns) in Table 2.6. A combination of TOD and ODD metrics discussed above can be used to holistically gauge a dialogue system's capability to perform inter-mode dialogues. The *classification-based* model marginally outperforms the *two-in-one* model.

Note that for the evaluation of ODD-grounded TODs, TOD-grounded ODDs and full inter-mode dialogues, we evaluate the response in a mode-tolerant manner. This means that even when the model generates a response of the wrong mode, we still evaluate that instance normally, instead of directly punishing the metric value to 0. For example, when evaluating BLEU, we still normally calculates the BLEU score against the ground-truth response even if the response generated by the inter-mode dialogue model is an ODD response. Of course, getting the mode wrong typically means poor scores.

2.5 Related Work

Preceding work

There have been multiple efforts on developing dialogue systems multi-tasking on various types of dialogues [5]. Adapter-Bot [51] uses a fixed backbone conversational model (DialoGPT) and triggers on-demand dialogue skills (e.g., empathetic responses, weather information, movie recommendation) via different adapters [53]. [54] trained a dialogue system that independently parameterizes different dialogue skills, and learns to select and combine each of them through Attention over Parameters. Shuster et al. (2019) multi-tasked on 12 separate dialogue datasets that focus on different skills and showed that a single unified model can perform decently well on all tasks. However, these works do not model the dependency between different types of dialogues in the multi-turn setting. Thus, they are not guaranteed to converse seamlessly and naturally in multiple dialogue modes simultaneously in a multi-turn conversation session.

Unlike the models trained on separate dialogue datasets, Smith et al. (2020) tried to fuse multiple skills into one conversation session. They built a new dialogue dataset named Blendedskilltalk containing dialogues where knowledge, emotional and personalizing skills are shown together in the same multi-turn conversation. They showed that systems fine-tuned on the new multi-skill dataset have improved ability in handling multiple skills simultaneously in the same multi-turn conversation session. However, they only targeted open-domain conversations. Our work, on the other hand, targets the fusion of general ODDs and TODs, as we view them as the two most mainstream forms of dialogues for the research community currently. Along the direction of fusing TODs and ODDs, Zhao et al. (2017) proposed to artificially augment TODs with randomly sampled utterances from a chitchat corpus, mainly to improve the out-of-domain recovery performance for the TOD system.

Sun et al. (2020) proposed to decorate TOD responses with ODD snippets, in order to make the dialogue agent sound more engaging and interactive. Unlike Sun et al. (2020), where ODD snippets act as a supplementary role to TOD responses, our dataset tackles the fusion of TODs and ODDs by treating them as parallel dialogue modes of equal importance, and focuses on modeling inter-mode dependency in the multi-turn setting.

Following work

There were efforts that followed our work that further explored dialogue systems with multiple utilities. Chen et al. (2022) additionally conditioned the chit-chat to knowledge snippets. Dialogue systems under their consideration are capable of TODs, chit-chat based on commonsense, and additionally chit-chat based on knowledge snippets. Li et al. (2022) created a new dataset OB-MultiWOZ, where

TOD sessions are enriched with QA-like information seeking grounded on external knowledge.

The zero-shot learning or in-context learning capabilities demonstrated by recent large language models [34] can help with TODs by generating dialogue states, dialogue acts and the final response. [57] prompted ChatGPT for state update prediction with a structured snippet containing task definition, domain description, dialogue history and user utterance.

2.6 Chapter Summary and Future Prospects

Our work demonstrated in this chapter serves the goal to develop dialogue systems that are capable of performing both TODs and ODDs with inter-mode dependency. Compared with traditional datasets, the new dataset FusedChat uniquely contains ODD-grounded TODs and TOD-grounded ODDs. It endeavors to fuse the two common forms of human conversations, i.e., casual open-ended conversations supported only by commonsense, and task-oriented conversations supported by specific knowledge bases. We show preliminary experiment results on two baseline models, which suggest huge room for improvement. We release dataset and baselines in order to propel future work on inter-mode dialogue systems.

We note that the framework set by FusedChat is limited. The dataset does not contain dialogue sessions containing more than one mode switch, which represents a gap with real-world scenarios. We suspect more mode switches could make inter-mode dialogues even more challenging. Our choice of TODs and ODDs does not represent the full scope of possible dialogue settings. We chose the most simple form of ODDs where the response is only determined by the context. Yet in the literature, ODDs have been grounded on various forms of information, such as personas (Zhang et al., 2018). We chose the classical setting of TODs as in MultiWOZ, which is defined by structured entity-centric knowledge bases. However, the concept of TODs has seen expansion, such as with unstructured knowledge access (Kim et al., 2020). We expect the fusion of more complex forms of ODDs and TODs to be more challenging, but they would even better represent human-level conversational abilities.

The construction of FusedChat required a lot of manual creative effort. It is thus very expensive to replicate the same routine for every new inter-mode dialogue scenario. Alternatively, zero-shot or few-shot models that can learn to perform inter-mode dialogues by mostly relying on separate single-mode dialogues are a promising direction. The zero-shot and in-context learning capabilities of large language models (Touvron et al., 2023; OpenAI, 2023) may prove especially useful. FusedChat can also serve as a test-bed for such paradigms.

This chapter has presented our efforts on fusing different dialogue modes in one dialogue agent by constructing a new dataset and proposing basic methods. Such a feature makes it more humanlike and usable. Another direction that pushes dialogue systems forward is to make them multimodal. In the next chapter, we illustrate our early effort in 2019 on grounding the system's response on audio signals that accompany text.

Chapter 3

Grounding dialogues on the audio modality

In the last chapter, we demonstrated our effort on fusing TODs and ODDs in a single dialogue agent. It is an essential capability that dialogue systems need to possess for them to perform on the human level. Another aspect that dialogue systems generally need to be better at is multimodality. Effort on developing dialogue systems started out assuming text-based conversation, where the user message was modeled as a sequence of words in a vocabulary. Real-world human conversation, in contrast, involves other modalities, such as voice, facial expression and body language, which can influence the conversation significantly in certain scenarios.

For computational budget concerns and availability of datasets, our initial effort performed in early 2019 focused only on audio signals and open-domain chitchat. This chapter demonstrates our work at exploring the impact of incorporating the audio features of the context into generative dialogue systems. The research question being addressed here is "how can audio features be effectively integrated into the popular Seq2Seq framework?"

Specifically, we first design an auxiliary response retrieval task for audio representation learning. Then we use word-level modality fusion to incorporate the audio features as additional context to our main generative model. Experiments show that our audio-augmented model outperforms the audio-free counterpart on perplexity, response diversity and human evaluation.[1]

3.1 Introduction

There are many ways that audio signals play a role in conversation. Audio signals naturally carry emotional information. For example, "Oh, my god!" generally expresses surprise. But depending on the voice shade, a wide range of different emotions can also be carried, including fear, anger and happiness. Audio signals can have strong semantic functions as well. They may augment or alter the meaning expressed in text. For example, "Oh, that's great!" usually shows positive attitude. But with a particular voice shade of contempt, the same utterance can be construed as sarcastic. Stress also plays a role in semantics: "I think *she* stole your money" emphasizes the speaker's opinion on the identity of the thief while "I think she stole *your* money" emphasizes the speaker's opinion on the identity of the victim.

Therefore, while identical from a written point of view, utterances may acquire different meanings based solely on audio information. Empowering a dialogue system with such information is necessary to interpret an utterance correctly and generate an appropriate response.

In this chapter, we explore dialogue generation augmented by the audio modality under the commonly-used Seq2Seq framework. First, because of the noisiness of the audio signal and the high dimensionality of raw audio features, we design an auxiliary response classification task to learn suitable audio representation for our dialogue generation objective. Second, we use word-level modality fusion for integrating audio features into the Seq2Seq framework. We design experiments to test how well our model can generate appropriate responses corresponding to the emotion and emphasis expressed in the audio. They show that our model captures the following phenomena in conversation: Vocally emphasized words in an utterance are relatively important to response generation; and emotion expressed in the audio of an utterance has influence on the response.

[1] The work in this chapter has been published in Young et al. (2020).

3.2 Proposed Approaches

3.2.1 Audio Representation Learning

Raw features extracted from audio sequences are high-dimensional and noisy. They are not suited as direct input to the dialogue generative model. For example, the number of dimensions for word embeddings used in RNNs is typically below 648. However, the number of raw audio features can reach 10000 (Schuller et al., 2013).

Therefore, we need an audio representation learning method to reduce the number of dimensions of the audio features and also make them suitable for the dialogue generation task.

For this purpose, we design an auxiliary response classification task based on audio features.

Specifically, we construct a set of $<context, response, label>$ triples, where *label* is binary indicating whether the context and response combination comes from a real conversation dataset D or is randomly assembled as a negative example. The goal of this task is to predict *label* based on the $<context, response>$ pair.

Following Lowe et al. (2015), our classification model is defined as:

$$f(x, y) = sigmoid(\boldsymbol{x}^T \boldsymbol{W} \boldsymbol{y}), \tag{3.1}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are representations of the context x [2] and response y respectively. Matrix $\boldsymbol{W}$ is the model parameter.

We use a universal sentence encoder Conneau et al. (2017) for the representation of response $\boldsymbol{y}$. For the purpose of finding the best audio context representation, $\boldsymbol{x}$ is determined only by audio features $\boldsymbol{a}_i$ of individual words in the context:

$$\boldsymbol{c} = avg(P(\boldsymbol{a}_i)), i \in [0, n), \tag{3.2}$$

where P is a perceptron and n is the number of words in the context. The model is shown in Figure 3.1.

[2]The meaning of a mathematical symbol stays the same in the same chapter. In different chapters we may use the same symbol for different meanings in order to keep the symbol set simple.

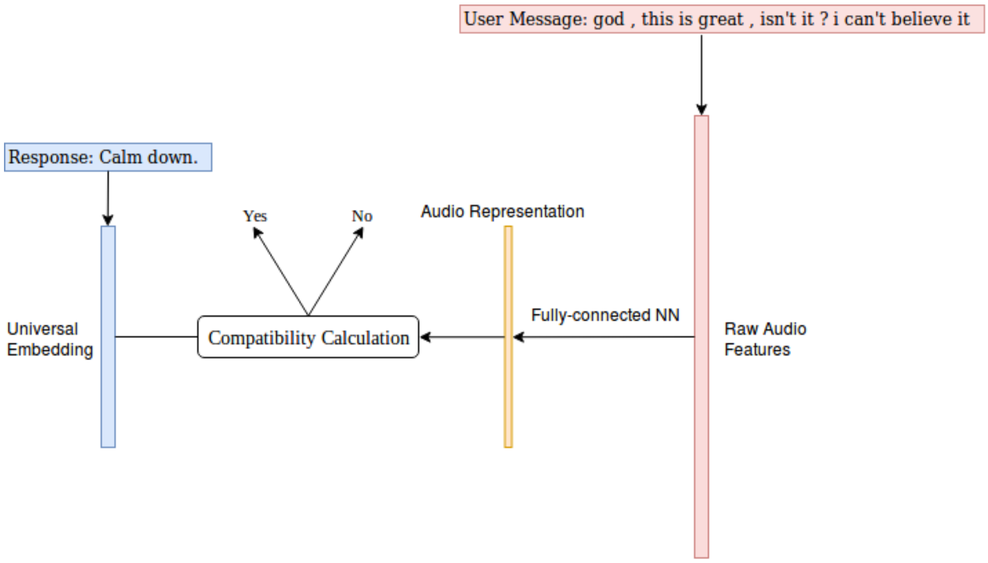


FIGURE 3.1: A response classification model is used as the auxiliary task for audio representation learning.

This model is trained on a conversation dataset D for best classification accuracy using mean squared loss between *label* and $f(x, y)$ in Equation (3.1). After training, the output of the perceptron $\tilde{\boldsymbol{a}}_i = P(\boldsymbol{a}_i)$ is taken as the word-level audio representation used in the generative dialogue systems.

Note that separately learning audio representations also serves the purpose of reducing memory burdens when the main generation model is trained.

3.2.2 Audio-augmented Seq2Seq Model

We build upon the general encoder-decoder framework which is based on sequence to sequence learning (Sutskever et al., 2014). The encoder represents a user message (context) $x = x_1 x_2 \cdots x_n$ with hidden representations $\boldsymbol{H} = \boldsymbol{h}_1 \boldsymbol{h}_2 \cdots \boldsymbol{h}_n$, which is briefly defined as below:

$$\boldsymbol{h}_n = \mathbf{LSTM_E}(\boldsymbol{h}_{n-1}, \boldsymbol{e}(x_n)), \tag{3.3}$$

where $\mathbf{E}$ denotes encoder. The decoder takes as input a context vector $\bar{\boldsymbol{c}}_{t-1}$ produced by an attention mechanism and the embedding of a previously decoded word

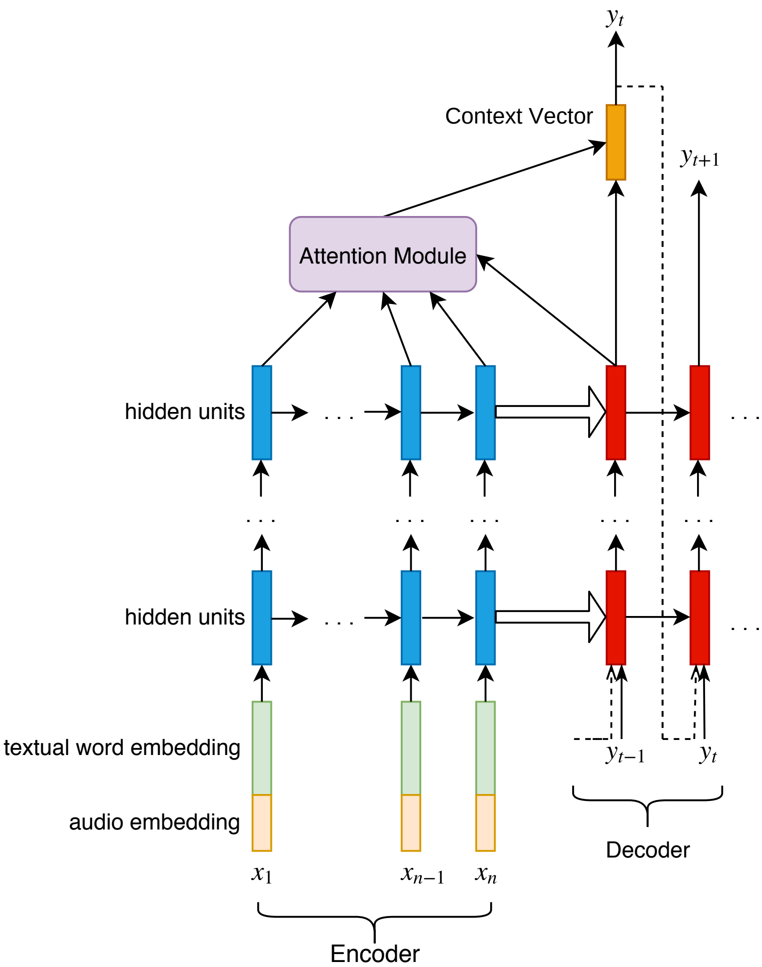


FIGURE 3.2: Audio-Seq2Seq model.

$\boldsymbol{e}(y_{t-1})$, and updates its state $\boldsymbol{s}_t$ using another LSTM:

$$\boldsymbol{s}_t = \mathbf{LSTM_D}(\boldsymbol{s}_{t-1}, [\bar{\boldsymbol{c}}_{t-1}; \boldsymbol{e}(y_{t-1})]), \tag{3.4}$$

where $\mathbf{D}$ denotes decoder. The decoder generates a token by sampling from the output probability distribution which is determined by $\bar{\boldsymbol{c}}_t$.

Following Chen et al. (2017), we use a simple word-level embedding concatenation method for integrating audio features into word representation:

$$\boldsymbol{e}(x_n) = [\boldsymbol{w}_n; \tilde{\boldsymbol{a}}_n], \tag{3.5}$$

where $\boldsymbol{w}_n$ is the traditional word embedding and $\tilde{\boldsymbol{a}}_n$ is the word-level audio representation. Thus, in our new Audio-Seq2Seq model (Figure 3.2), the word representation contains both textual and audio information.

3.3 Experiments and Results

3.3.1 Dataset

Most of the existing and consolidated datasets used in dialogue system related research come with textual content only (Lowe et al., 2015; Ritter et al., 2010). Fortunately, along with the growing interest in multimodal systems, some datasets are fit for our task. We experiment with two such datasets, the Interactive Emotional Dyadic Motion Capture dataset (IEMOCAP, Busso et al., 2008) and the Multimodal EmotionLines dataset (MELD, Poria et al., 2018).

IEMOCAP was designed with the main intent of providing a corpus of dyadic conversations capable of conveying emotions. Two types of dialogue sessions were created for IEMOCAP to achieve this task: scripted and spontaneous sessions. In the scripted case, two actors, a male and a female, were asked to rehearse some previously memorized scripts, as this supposedly leads to a more genuine expression of emotions than directly reading off a script. In the spontaneous case, the actors were given more liberty to use their own words to discuss about selected emotion-evoking topics. This supposedly allows the actors to express more natural emotions. The dataset contains a total of 10039 utterances with their corresponding audio segments.

MELD is a dataset containing utterances from the TV series *Friends*. For each utterance multimodal information in the form of text, audio and video is provided. MELD consists of 1433 dialogues for a total of 13708 utterances.

These two datasets are suitable for our purposes as the audio component is strongly representative of the speaker's emotional state and plays a pivotal role in the meaning to be conveyed.

TABLE 3.1: Number of utterances, average length of utterances, development set sizes, test set sizes and vocabulary sizes for IEMOCAP and MELD datasets.

	IEMOCAP	*MELD*
No. Utt.	7901	12274
Avg Utt. Length	15.26	10.69
Train. set size	6000	10000
Dev. set size	1000	1174
Test set size	901	1000
Vocabulary size	2171	3123

3.3.2 Experiment Details

3.3.2.1 Data Preprocessing

From IEMOCAP and MELD set of dialogues we extract <*sentence, response*> pairs by taking successive utterances within individual dialogues. Formally, from dialogue $d_i = \{u_1, .., u_{n_i}\}$, where $u_1, .., u_{n_i}$ are the utterances composing the dialogue, we extract the set of pairs $\{<u_1, u_2>, .., <u_{n_i-1}, u_{n_i}>\}$. From the resulting pairs we create a vocabulary, for each of the datasets, containing only the terms with more than one occurrence in the respective corpus and that are present in the standard English vocabulary [66] and those that are not present in the English vocabulary but occur ten or more times in the dataset. Our final vocabulary sizes are 2171 for IEMOCAP and 3123 for MELD. After these procedures we end up with a total of 7901 utterances for IEMOCAP and 12274 for MELD.

The audio segments provided within the datasets are given at a sentence granularity. We therefore conduct word alignment and obtain word-level audio features. We first use the GENTLE forced aligner (Ochshorn and Hawkins, 2016) to find the start and end timestamps of each word within a sentence. Then, with OpenSMILE (Eyben et al., 2010), we extract 6373 raw audio features for each word. We use the *IS13_ComParE.conf* configuration (Schuller et al., 2013) that has been widely used in emotion recognition tasks (Poria et al., 2018; Xianyu et al., 2016), rendering it a suitable choice for our case, as impacting the conveyed emotion is one of the primary ways audio features make an impact in conversation. We randomly sample utterances from the datasets to split into training, development, and test sets. In Table 3.1 we report some of the most important statistics regarding the datasets we operate on.

3.3.2.2 Model training details

In our audio representation learning model (Section 3.2.1), the response sentence embedding given by the universal sentence encoder has size 4096. During the training process, the best audio representation extractor is obtained at the point when the classification accuracy on the development dataset is the highest.

We use the Seq2Seq model with Luong attention mechanism (Luong et al., 2015) as the backbone of our main audio-augmented model. It is a pruned version of the main model (Figure 3.2) that does not use audio features in its word-level representation. After being trained on a large text-based conversation dataset, the resulting model parameters are transferred to the main model as initialization of its parameters corresponding to textual input.

A 3.3M Reddit Conversation Dataset [71] is used for this purpose. We filter it using the vocabularies previously created for the audio conversation datasets. Specifically a conversation pair $<u_j, u_{j+1}>$ is removed if it contains more than one out-of-vocabulary term.

We follow Luong et al. (2017) for most of the hyperparameter settings. All generation models are trained for 50000 steps with batch size 256 after initialization with the pretrained model. The learning rate is set to 0.1. Dropout rate is 0.3. We use 2 layers of hidden units. Keeping audio representation fixed, we search for the optimal text embedding dimension in $10, 25, 50, 100$. 100 yields the best results. By manually inspecting the generated responses at different steps, we find that they are most natural-sounding when the models slightly overfit. In contrast, the models generate overly simple responses when the development perplexity is lowest. This might be due to the fact that the audio conversation datasets are relatively small. We manually choose the best checkpoint for testing based on human perception of response quality on the development set after the models start overfitting.

TABLE 3.2: Auxiliary response classification task accuracy varying the dimension of the audio representation.

Dataset \ Dimension	*25*	*50*	*100*
IEMOCAP	59.4%	62.4%	61.8%
MELD	54.8%	54.8%	54.6%

3.3.3 Experiment Results

3.3.3.1 Results on audio representation learning

The results are shown in Table 3.2. The fact that the accuracies are much higher than 50% indicates that audio features indeed carry information that is relevant to conversation. Overall, the accuracies show only a slight improvement in spite of a substantial increase in dimension moving up from 25. We choose 25 as the size of the audio representation that is adopted in all the experiments.

3.3.3.2 Perplexity, Diversity and Human Evaluation

To In this work we consider two automatic metrics in addition to human judgment: perplexity and diversity. Lower perplexity on the testset indicates that the model fits the testset better. *Diversity* is defined as the number of unique words generated by the model over the test set. Lack of diversity and tendency to generate similar, short responses regardless of the different inputs is a notorious problem in generative conversational models (Liu et al., 2018). A model that generates interesting and information-rich responses is characterized by high diversity. Automatic and human evaluation results are shown in Tables 3.3 and 3.4. "± value" indicates standard deviation. We see that the Audio-Seq2Seq model achieves lower perplexity and higher diversity on both datasets. Since adding audio features essentially enriches the representation of input to the Seq2Seq model, it helps the model generate the correct output (lower perplexity). Also, the additional audio information increases the diversity of the input, which helps generate diverse responses (higher diversity).

A number of samples were manually selected from the test set for human evaluation. During this process, we only chose samples with high-quality user messages,

TABLE 3.3: Statistics on IEMOCAP.

Model \ Metric	*Perplexity*	*Diversity*	*Human Preference*
Seq2Seq	36.83 ± 0.34	805 ± 10.5	44.4%
Audio-Seq2Seq	31.13 ± 0.31	831 ± 12.8	55.6%

TABLE 3.4: Statistics on MELD.

Model \ Metric	*Perplexity*	*Diversity*	*Human Preference*
Seq2Seq	47.83 ± 0.44	567 ± 8.7	46.5%
Audio-Seq2Seq	46.19 ± 0.49	629 ± 10.0	53.5%

TABLE 3.5: Number of responses by the corresponding model that are deemed as better according to AMT judges. See the corresponding percentages in Tables 3.3 and 3.4.

	Audio-Seq2Seq	*Seq2Seq*
IEMOCAP	354	283
MELD	359	313

considering both audio and text. A sample is excluded if the audio is noisy or the user message is too short or generic.

We conducted pair-wise comparisons between responses generated by the Audio-Seq2Seq model and the Seq2Seq model. Seven judges were recruited on the Amazon Mechanical Turk (AMT) platform. After listening to the audio of the user message, the judges were asked to select the best response from those generated by the two models. We asked the judges to follow two rules: (1) Prefer the response that is compatible with the tone of the user message. (2) Prefer longer and more informative responses. The results of the judges' evaluation are shown in Table 3.5.

Table 3.6 shows cases where the responses produced by the Audio-Seq2Seq model suit the audio context better. For example, in the second case, the voice of the user message shows excitement and restlessness, which is captured by the "calm down" in the response.

TABLE 3.6: Samples produced by Audio-Seq2Seq. Considering the tone of the context, the responses Audio-Seq2Seq produced are more appropriate.

Context	Seq2Seq Response	Audio-Seq2Seq Response
because i know you're at a desk right now in your corporate building and you're going to send me a check for sixteen dollars and forty two cents!	right, ok.	oh, no, no, no, no, no. i don't know. i don't know. it's not what i can do.
what time is it? god, this is great, isn't it? i can't believe it.	yeah, it is	okay, all right. just just calm down
no. it's his newspapers is what it is now every month there's a new boy comes home and larry's going to be the next one.	what?	alright. alright. alright just listen.

3.3.3.3 Emotional appropriateness

Audio features can be strongly indicative of the speaker's emotion, and thus have influence on the response. In order to quantitatively evaluate the ability of our model to generate emotionally appropriate responses, we design an experiment with an artificially constructed set of 200 audio message samples of two different emotional states. Specifically, 100 textual message samples are selected from the test set and for each sample we manufacture two audio segments of different emotions by availing ourselves of the MARY Text-To-Speech system (MaryTTS, Schröder and Trouvain, 2001).

Following Russell's Circumplex model of affect (Russell, 1980) we vary the valence dimensions of the synthesized audio segments. With arousal and valence in the range $[0, 1]$, we use a fixed arousal value of 0.9 combined with the two valence values 0.1 and 0.9. When valence $= 0.9$, the synthesized speech is fast and highly-pitched, exhibiting an excited emotional state. Whereas when valence $= 0.1$, the synthesized speech is slow and calm. Our Audio-Seq2Seq model generates two responses corresponding to those two audio segments of different emotion states. To evaluate how well a response matches an emotional state, we shuffle the two responses and ask human judges to match audio segments with the responses to see if the results agree with the model's.

This association task performed by three judges shows that human evaluation tends to agree with the responses generated by the model more often than random guess. Details are given in Table 3.7. Table 3.8 shows cases where the model seems to be able to perceive the emotional state of the speaker and adapt its response accordingly. When the audio expresses an excited state (valence $= 0.9$), the model is able to tune its response in a suitable manner. For instance in the first sample the second response shows a strong correlation with the excited and agitated state of the speaker by asking him to calm down. In the second sample, the higher rate with which the valence = 0.9 context is uttered due to the excited state makes the speaker sound less sincere thus eliciting a stuttered and complaining response as compared to the more composed and calm one when valence = 0.1.

Table 3.7: Percentage of cases on which the judges' verdicts agree or disagree with the model. The number of agreement cases exceeds the number of disagreement cases. This indicates that to a certain degree, the model's response captures the emotion expressed in the audio features of the message.

Model	*Agree*	*Disagree*	*Cannot determine*
IEMOCAP	25.4%	15.1%	59.5%
MELD	28.2%	16.7%	55.1%

Table 3.8: The model adapts its response based on the emotion expressed in the message through audio features.

Context	*Valence = 0.1*	*Valence = 0.9*
turn it off it's driving me mad.	I won't.	well, do try to control yourself darling.
okay that's helpful. thanks.	i've been trying to work this backwards	this is all this is unfair.

3.3.3.4 Attention on vocally emphasized words

In a conversation, vocally emphasized words in an utterance are most important to information communication. To evaluate how well our model captures this phenomenon, we calculate the correlation between the volume/duration of the audio segments of words in the user message and the attention the words get during the generation process.

We take the length of the audio segment of an individual word as the *duration* of that word and *maximum amplitude* is used to indicate *volume*.

For calculating attention on a word in the message, we sum all attention scores it gets during the response generation process. Specifically, for the generation of response word y_t, the attention score on message word x_i is a_{it}. For the generated response $[y_1, y_2, ..., y_n]$, the total attention on x_i is $a_i = \sum_{t=1}^{n} a_{it}$.

We normalize attention, duration and maximum amplitude by dividing them by average values over the message. Pearson and Spearman correlations are calculated on attention-duration and attention-maximum amplitude pairs. The results are shown in Tables 3.9 and 3.10. On both datasets our experiment shows relatively strong positive correlation between attention and duration. For attention and maximum amplitude, however, our calculation only shows slightly positive correlation. This implies that in our dataset, length is more indicative of a word's importance to the dialogue system than volume. But this observation cannot be generalized without more experiments on more datasets.

Two examples are shown in Figure 3.3. In the message "turn it off it's driving me mad", "off", "driving" and "mad" are vocally emphasized. Accordingly, attention scores on those three words are relatively high. In a shorter example, "oh that's attractive", the word "attractive" contains the most semantic information. It is vocally emphasized and gets the most attention.

TABLE 3.9: The correlation between word attention and duration/maximum amplitude on IEMOCAP.

IEMOCAP	*Pearson's r*	*Spearman's ρ*
Attention/Duration	0.418	0.384
Attention/Max Amp.	0.096	0.128

TABLE 3.10: The correlation between word attention and duration/maximum amplitude on MELD.

MELD	*Pearson's r*	*Spearman's ρ*
Attention/Duration	0.312	0.334
Attention/Max Amp.	0.094	0.069

3.4 Related Work

As a research effort on developing dialogue systems that are additionally conditioned on the audio modality in 2019, our work was inspired by previous work on multimodal NLP and representation learning. At the same time, we gladly notice following work that further explored multi-modal dialogue systems. This section discusses both categories of related work.

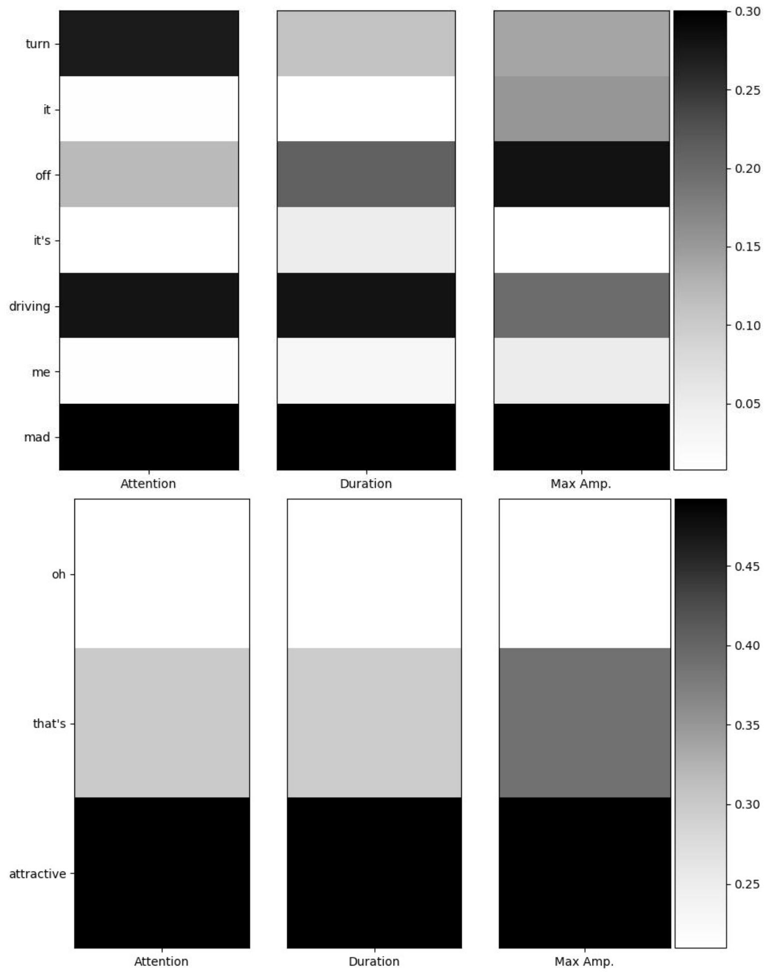


FIGURE 3.3: The attention of a word in the source sequence is positively correlated with both of its duration and maximum amplitude.

Preceding work

Human conversation naturally involves multiple modalities. This important fact had been noticed in research preceding our work demonstrated in this chapter.

First, the subject/background of a conversation can be multimodal. For example, in image-grounded conversation (Mostafazadeh et al., 2017), two interlocutors generate conversations based on a shared image. For this task, visual features of the image need to be infused into the context vector. Alamri et al. (2018) proposed Visual Scene-aware Dialogs, a scenario where the dialogue system discusses dynamic scenes with humans. A scene, in the form of a short video, is presented to the interlocutors as the conversational context. For this task, Hori et al. (2018) incorporated techniques for multimodal attention-based video description into an end-to-end dialogue system. Audio and visual features that come from deep video description models are used to augment the context vector. Saha et al. (2018) proposed a large domain-aware multimodal conversation dataset where shoppers and sales agents converse about products in the fashion domain. Each conversational turn is composed of text and corresponding images being referred to. For this scenario, Agarwal et al. (2018) proposed a multimodal extension to the Hierarchical Recurrent Encoder-Decoder (HRED, Serban et al., 2016) for in-turn multimodality and multi-turn context representation.

Second, human conversation itself involves multiple channels of information. Voice, body language and facial expressions all play roles in conversation. In an ideal human-machine conversational system, machines should understand this multimodal language. This information had seen use in conversation analysis. Yu (2015) proposed to model user engagement and attention in real time by leveraging multimodal human behaviors, such as smiles and speech volume. Gu et al. (2018) performed emotion recognition, sentiment analysis, and speaker trait analysis on conversation data using a hierarchical encoder that formulates word-level features from video, audio, and text data into conversation-level features with modality attention.

Our method of word-level modality fusion had already seen use in multimodal sentiment analysis. In Chen et al. (2017), the RNN, which acts as the utterance encoder, takes a concatenation of audio, video and text features as input at every time step. On the IEMOCAP dataset (Busso et al., 2008; Chen et al., 2017) showed considerable improvement on dialogue emotion classification accuracy by integrating audio features. This result motivated our work - since incorporating audio features improves emotion classification accuracy in conversation and emotion is useful to response generation (Zhou et al., 2018), we hypothesized that incorporating audio features improves response generation.

Following work

We would also like to note works done following our work along the research line of multi-modal dialogue systems.

First, following the breakthrough on large-scale self-supervised pretraining for language models (Devlin et al., 2018; Brown et al., 2020), multi-modal learning has adopted the same routine.

ViLBERT (Lu et al., 2019, Vision-and-Language BERT) presented a model for learning task-agnostic joint representations of image content and natural language. They extended the popular BERT architecture to a multi-modal two-stream model, processing both visual and textual inputs in separate streams that interact through co-attentional transformer layers. ViLT (Kim et al., 2021, Vision-and-Language Transformer) commissioned the transformer module to extract and process visual features in place of a separate deep visual embedder. Initialized from ViT, the transformer module extracts visual features from the image modality and processes them along with the language modality. Their design led to significant runtime and parameter efficiency. Akbari et al. (2021) presented a framework for learning multimodal representations from unlabeled data using Transformers. Specifically, their Video-Audio-Text Transformer (VATT) takes raw signals as inputs and extracts multimodal representations that benefit various downstream tasks. They trained VATT end-to-end from scratch using multimodal contrastive losses. Its usefulness was proven based on video action recognition, audio event classification, and text-to-video retrieval. Data2vec (Baevski et al., 2022) was another attempt at performing self-supervised learning across modalities. It uses the same learning method for either speech, NLP or computer vision. It predicts latent representations of the full input data based on a masked view of the input in a self-distillation setup using a standard Transformer architecture. Instead of predicting modality-specific targets such as words, visual tokens or units of human speech which are local in nature, data2vec predicts contextualized latent representations that contain information from the entire input.

Second, researchers have come up new scenarios that further enrich multi-modal dialogues. For example, Kamezawa et al. (2020) proposed a visually-grounded first-person dialogue dataset with verbal and non-verbal responses. It provides manually annotated first-person images and eye-gaze locations of the speakers. Moon et al. (2020) envisioned dialogue systems to both take multimodal inputs and perform multimodal actions. It introduced Situated Interactive Multi-Modal Conversations (SIMMC) as a new dataset aimed at training agents that take multimodal actions (such as showing pictures) grounded in a co-evolving multimodal input context in addition to the dialogue history.

Third, new approaches have been proposed based on the two advances above. Le and Hoi (2020) leveraged the power of pre-trained language models for improving video-grounded dialogues. GPT2 was extended to tackle these challenges by formulating the video-grounded dialogue task as a Seq2Seq learning task, combining both visual and textual representation into a structured sequence. Li et al. (2021) proposed a very similar approach with GPT2. Their difference lies in their choices of fine-tuning tasks. They both fine-tune the model for the traditional response generation task. Additionally, Le and Hoi (2020) fine-tuned their model for masked multi-modal modeling, and to match video-text pairs. Li et al. (2021) on the other hand, chose video-audio feature regression and caption generation. Most recently OpenAI (2023) demonstrated a huge jump in performance on multimodal tasks such as visual question answering, conditioned on both images and text. This success could easily be transferred to dialogues in the near future.

Multimodal dialogue systems discussed in this chapter have mostly been studied in the ODD (open-domain dialogue) scenario because of dataset availability. To connect with Chapter 2, we note that there have been efforts on setting up multimodal TOD scenarios, because next-generation TOD systems need to understand conversational contexts with their perceived surroundings, to effectively help users in the real-world multimodal environment. For example, Yang et al. (2021) proposed a new dataset MMDialKB that conditions end-to-end TOD systems on multimodal knowledge bases. Kottur et al. (2021) presented SIMMC 2.0, which includes task-oriented user-assistant dialogues in the shopping domain, grounded in immersive and photo-realistic scenes.

3.5 Chapter Summary and Future Prospects

In this chapter, we augment the common Seq2Seq dialogue model with audio features and show that the resulting model outperforms the audio-free baseline on several evaluation metrics. It also captures interesting audio-related conversation phenomena.

Although only using text in dialogue systems is a good enough approximation in a lot of scenarios, other modalities have to be integrated before automatic dialogue systems can reach human performance. Our work belongs to such a line of research that strives to build multimodal dialogue systems.

We view multimodal dialogue systems as a very promising direction worthy of future investigation. We believe it implies immediate application value and at the same time strongly relates to the long-term success of the field of AI in general. Although recent breakthroughs such as GPT4 (OpenAI, 2023) could push dialogue systems forward along this direction, we believe that employing large-scale multimodal pretraining on conversational data directly could further help. For example, one might consider explicitly injecting speaker awareness into the model pretraining process, in order to increase the ability of the model to handle the bi-party turn-by-turn nature of dialogue modeling.

Multimodality is an aspect that dialogue systems generally need to improve on to be more humanlike. Another aspect is knowledge grounding. The next chapter will cover our early attempt on addressing it in 2018.

Chapter 4

Grounding dialogues on commonsense knowledge

The last 2 chapters discussed how different dialogue modes can be fused together and how the audio modality may be incorporated in addition to text. They can be seen as different aspects that both serve the purpose of pushing vanilla dialogue systems naively trained on context response pairs towards human-like performance. This chapter discusses a third aspect, that is, grounding dialogue systems on external knowledge. It covers an early work done in 2018 along this direction and we used LSTMs as backbones and limited our effort on the open-domain retrieval-based scenario as it was easier to evaluate. In addition, retrieval-based systems based on a human-written response repository generally produced responses of higher quality before the advent of large language models. We also limited ourselves to commonsense knowledge as it was readily available in the form of large semantic networks such as ConceptNet (Speer et al., 2017). The work covered in this chapter focuses on the research question: "What are the effective ways of incorporating commonsense knowledge and how may it improve dialogue responses?"

Commonsense knowledge is considered a key part of human intelligence (Cambria et al., 2022). Since dialogue systems are expected to respond to human utterances in an interesting and engaging way, commonsense knowledge has to be integrated into the model effectively.

In this chapter, we explain our effort on investigating the impact of providing commonsense knowledge about the concepts covered in the dialogue. In the retrieval-based scenario, we propose the Tri-LSTM model to jointly take into account message and commonsense for selecting an appropriate response. Our experiments suggest that the knowledge-augmented models are superior to their knowledge-free counterparts in automatic evaluation.[1]

4.1 Introduction

By training on a large number of message-response pairs, most dialogue systems attempt to produce an appropriate response based solely on the message itself, without any memory module. In natural human conversation, however, people respond to each other's utterances in a meaningful way not only by paying attention to the latest utterance of the conversational partner itself, but also by recalling relevant information about the concepts covered in the utterance and integrating it into their responses. Such information may contain personal experience, recent events, commonsense knowledge and more (Figure 4.1). As a result, it's speculated that a conversational model with a "memory look-up" module can mimic human conversations more closely (Ghazvininejad et al., 2017; Bordes and Weston, 2016).

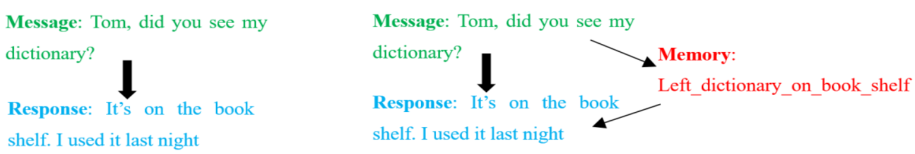


FIGURE 4.1: Left: In traditional dialogue modeling, the response is determined solely by the message. (Arrows denote dependencies) Right: The responder recalls relevant information from memory about the message; memory and message jointly determine the response. As in the illustrated example, the responder model retrieves the event "Left_dictionary_on_book_shelf" from memory, which, along with the message, triggers a meaningful response.

In open-domain human-computer conversation, where the model is expected to respond to human utterances in an interesting and engaging way, commonsense knowledge has to be integrated into the model effectively. In artificial intelligence,

[1]The work in this chapter has been published in Young et al. (2018).

commonsense knowledge is the set of background information that an individual is intended to know or assume and the ability to use it when appropriate (Minsky, 1986; Cambria et al., 2009; Tran et al., 2016). Due to the vastness of such knowledge, in an era without ultra-large language models (OpenAI, 2023), we speculated that this goal might be better suited by employing an external memory module containing such knowledge than forcing the model to encode it in model parameters. Hence we investigate augmenting end-to-end dialogue systems with commonsense knowledge as external memory.

Several commonsense knowledge bases have been constructed during the past decade, such as ConceptNet (Speer and Havasi, 2012) and SenticNet (Cambria et al., 2014). The aim is to give a foundation of real-world knowledge to a variety of AI applications. Typically a commonsense knowledge base can be seen as a *semantic network* where *concepts* are nodes in the graph and *relations* are edges. Each $<concept1, relation, concept2>$ triple is termed an *assertion*. Based on the Open Mind Common Sense project (Singh et al., 2002), ConceptNet not only contains objective facts such as "Paris is the capital of France" that are constantly true, but also captures informal relations between common concepts that are part of everyday knowledge such as "A dog is a pet". This feature of ConceptNet is desirable for our purpose, because the ability to recognize the informal relations between common concepts is necessary in the open-domain conversation setting in this chapter.

4.2 Proposed Approaches

4.2.1 Task Definition

Our effort concentrated on integrating commonsense knowledge into retrieval-based conversational models as the first step, as they were easier to evaluate (Liu et al., 2016; Lowe et al., 2016) and generally took a lot less data to train before the dawn of large-scale pretrained models. We left the generation-based scenario to future work.

Message (*context*) x and *response* y are a sequence of tokens from vocabulary V. Given x and a set of response candidates $[y_1, y_2, y_3, ..., y_K] \in Y$, the model chooses the most appropriate response $\hat{y}$ according to:

$$\hat{y} = \arg\max_{y \in Y} f(x, y), \tag{4.1}$$

where $f(x, y)$ is a scoring function measuring the "compatibility" of x and y. The model is trained on $<message, response, label>$ triples with cross entropy loss, where *label* is binary indicating whether the $<message, response>$ pair comes from real data or is randomly combined as a negative example.

4.2.2 Dual-LSTM Encoder

The Dual-LSTM encoder (Lowe et al., 2015) represents the message x and response y as fixed-size embeddings $\vec{x}$ and $\vec{y}$ with the last hidden states of the same LSTM. The compatibility function of the two is thus defined by:

$$f(x, y) = \sigma(\vec{x}^T W \vec{y}), \tag{4.2}$$

where matrix $W \in \mathcal{R}^{D \times D}$ is learned during training. Dual-LSTM Encoder is used as both the baseline and the backbone model for our proposed model Tri-LSTM (Chapter 4.2.4).

4.2.3 Commonsense Knowledge Retrieval

We assume that a commonsense knowledge base is composed of assertions A about concepts C. Each assertion $a \in A$ takes the form of a triple $<c_1, r, c_2>$, where $r \in R$ is a *relation* between c_1 and c_2, such as *IsA*, *CapableOf*, etc. c_1, c_2 are concepts in C. The relation set R is typically much smaller than C. c can either be a single word (e.g., "dog" and "book") or a multi-word expression (e.g., "take_a_stand" and "go_shopping"). We build a dictionary H out of A where every concept c is a key and a list of all assertions in A concerning c, i.e., $c = c_1$ or $c = c_2$, is the value. Our goal is to retrieve commonsense knowledge about every concept covered in the message.

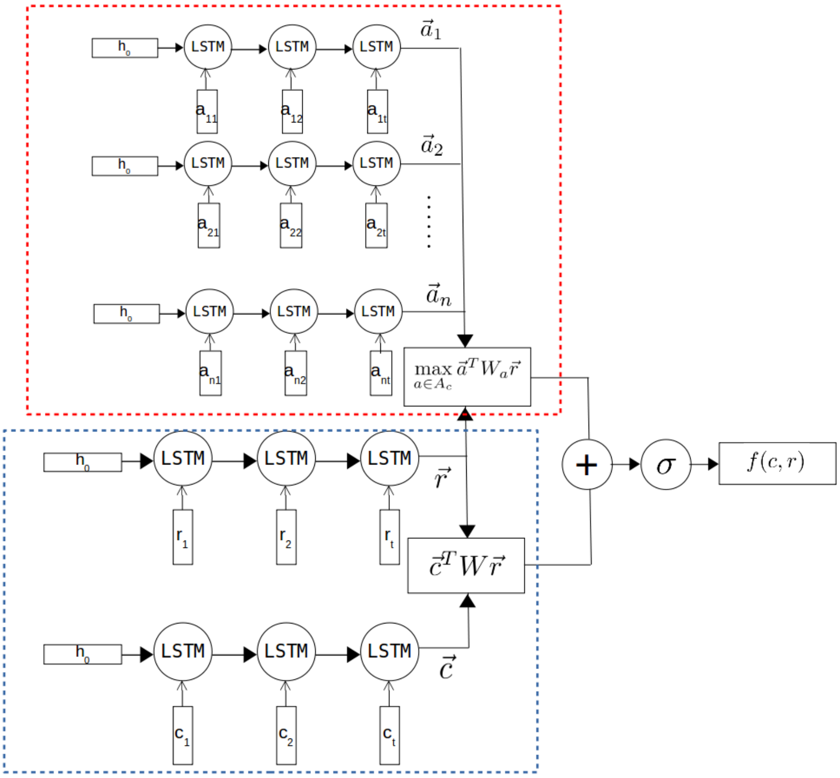


FIGURE 4.2: Tri-LSTM encoder. We use LSTM to encode message, response and commonsense assertions. LSTM weights for message and response are tied. The lower box is equal to a Dual-LSTM encoder. The upper box is the memory module encoding all commonsense assertions.

We define A_x as the set of commonsense assertions concerned with message x. To recover concepts in message x, we use simple n-gram matching ($n \leq N$). Every n-gram in c is considered a potential concept[2].

- During preliminary trials, we experimented with the automatic concept parser proposed in Rajagopal et al. (2013). We quickly found that the recall rate was sub-optimal for various reasons. For example, the parser would always take the whole "roasted duck" from a sentence instead of just "duck". This is an issue since ConceptNet (or any commonsense knowledge base) is incomplete. And we would ideally like to take "duck" as well to get more matches in general.

  Therefore we ended up with simple n-gram matching, i.e., we consider any and all segments from the sentence. Note that this is not a perfect solution

[2]For unigrams, we exclude a set of stopwords. Both the original version and stemmed version of every word are considered.

either since it doesn't recover "roasted duck" from a sentence like "I roasted a duck this morning". But the recall rate was much more favorable. The choice of $N = 5$ is based on the fact the 99.2% of concepts in ConceptNet are less than or equal to 5 words in length. Most of the >5 ones can be deconstructed.

If the n-gram is a key in H, the corresponding value, i.e., all assertions in A concerning the concept, are added to A_x (Figure 4.3).

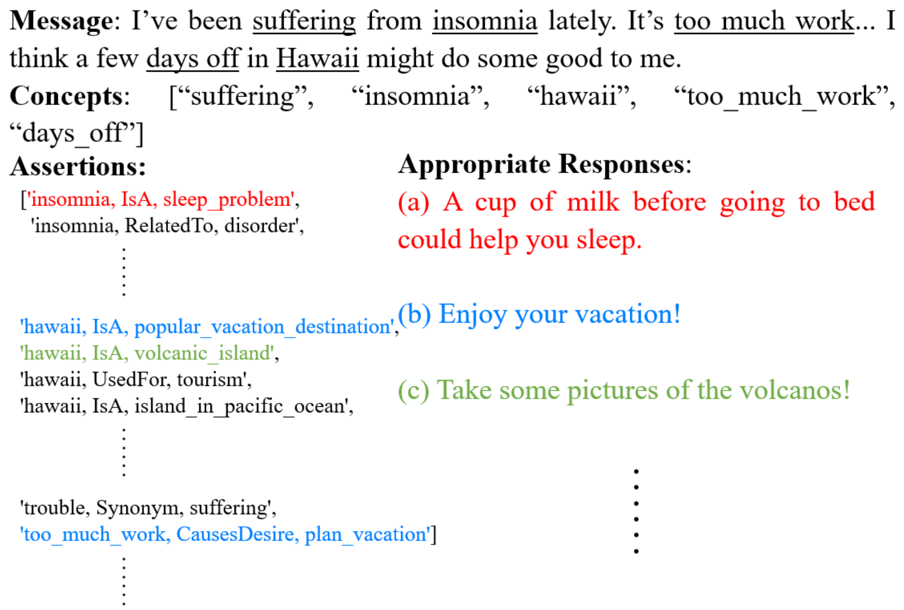


FIGURE 4.3: In the illustrated case, five concepts are identified in the message. All assertions associated with the five concepts constitute A_x. We show three appropriate responses for this single message. Each of them is associated with (same color) only one or two commonsense assertions, which is a paradigm in open-domain conversation and provides ground for our max-pooling strategy. It is also possible that an appropriate response is not relevant to any of the common assertions in A_x at all, in which case our method falls back to Dual-LSTM.

4.2.4 Tri-LSTM Encoder

Our main approach to integrating commonsense knowledge into the conversational model involves using another LSTM for encoding all assertions a in A_x, as illustrated in Figure 4.2. Each a, originally in the form of $<c_1, r, c_2>$, is transformed into a sequence of tokens by chunking c_1, c_2, concepts which are potentially multi-word phrases, into $[c_{11}, c_{12}, c_{13}...]$ and $[c_{21}, c_{22}, c_{23}...]$. Thus, $a = [c_{11}, c_{12}, c_{13}, ..., r, c_{21}, c_{22}, c_{23}...]$.

We add R to vocabulary V, that is, each r in R will be treated like any regular word in V during encoding. We decide not to use each concept c as a unit for encoding a because C is typically too large (>1M). a is encoded as embedding representation $\vec{a}$ using another LSTM. Note that this encoding scheme is suitable for any natural utterances containing commonsense knowledge in addition to well-structured assertions. We define the *match score* (compatibility score) of assertion a and response y as:

$$m(a, y) = \vec{a}^T W_a \vec{y}, \tag{4.3}$$

where $W_a \in \mathcal{R}^{D \times D}$ is learned during training. The number of commonsense assertions A_x associated with a message is usually large (>100 in our experiment). We observe that in a lot of cases of open-domain conversation, response y can be seen as triggered by certain perception of message x defined by one or more assertions in A_x, as illustrated in Figure 4.3. For example, the word 'Insomnia' in the message is related to the commonsense assertion 'Insomnia, IsA, sleep_problem'. The appropriate response ('go to bed') is then matched to 'sleep_problem'. Similarly, the word 'Hawaii' in the message is related to the commonsense assertion 'Hawaii, UsedFor, tourism'. The appropriate response ('enjoy vacation') is then matched based on 'tourism'.

Our assumption is that A_x is helpful in selecting an appropriate response y. However, usually very few assertions in A_x are related to a particular response y in the open-domain setting. As a result, we define the *match score* of A_x and y as[3]

$$m(A_x, y) = \max_{a \in A_x} m(a, y), \tag{4.4}$$

that is, we only consider the commonsense assertion a with the highest match score with y, as most of A_x are not relevant to y. Incorporating $m(A_x, y)$ into the Dual-LSTM encoder, our Tri-LSTM encoder model is thus defined as:

$$f(x, y) = \sigma(\vec{x}^T W \vec{y} + m(A_x, y)), \tag{4.5}$$

[3]Here the symbol m is overloaded with 2 meanings.

i.e., we use simple addition to supplement x with A_x, without introducing a mechanism for any further interaction between x and A_x. This simple approach is suitable for response selection and proves effective in practice.

The intuition we are trying to capture here is that an appropriate response y should not only be compatible with x, but also related to certain memory recall triggered by x as captured by $m(A_x, y)$. In our case, the memory is commonsense knowledge about the world. In cases where $A_x = \emptyset$, i.e., no commonsense knowledge is recalled, $m(A_x, y) = 0$ and the model degenerates to Dual-LSTM encoder.

4.2.5 Comparison Approaches

4.2.5.1 Supervised Word Embeddings

We follow Bordes and Weston (2016) and Dodge et al. (2015) and use supervised word embeddings as a baseline. Word embeddings are most well-known in the context of unsupervised training on raw text as in Mikolov et al. (2013), yet they can also be used to score message-response pairs. The embedding vectors are trained directly for this goal. In this setting, the "compatibility" function of x and y is defined as:

$$f(x, y) = \vec{x}^T \vec{y} \tag{4.6}$$

In this setting, $\vec{x}, \vec{y}$ are bag-of-words embeddings. With retrieved commonsense assertions A_x, we embed each $a \in A_x$ to bag-of-words representation $\vec{a}$ and have:

$$f(x, y) = \vec{x}^T \vec{y} + \max_{a \in A_x} \; \vec{a}^T \vec{y}. \tag{4.7}$$

This linear model differs from Tri-LSTM encoder in that it represents an utterance with its bag-of-words embedding instead of RNNs.

4.2.5.2 Memory Networks

Memory networks (Sukhbaatar et al., 2015, Weston et al., 2014) are a class of models that perform language understanding by incorporating a memory component. They perform attention over memory to retrieve all relevant information that may help with the task. In our dialogue modeling setting, we use A_x as the memory component. Our implementation of memory networks, similar to Bordes and Weston (2016) and Dodge et al. (2015), differs from supervised word embeddings described above in only one aspect: how to treat multiple entries in memory. In memory networks, output memory representation $\vec{o} = \sum_i p_i \vec{a}_i$, where $\vec{a}_i$ is the bag-of-words embedding of $a_i \in A_x$ and p_i is the attention signal over memory A_x calculated by $p_i = softmax(\vec{x}^T \vec{a}_i)$. The "compatibility" function of x and y is defined as:

$$f(x, y) = (\vec{x} + \vec{o})^T \vec{y} = \vec{x}^T \vec{y} + (\sum_i p_i \vec{a}_i)^T \vec{y} \tag{4.8}$$

In contrast to supervised word embeddings described above, attention over memory is determined by message x.

4.3 Experiments and Results

4.3.1 Twitter Dialogue Dataset

1.4M Twitter <message, response> pairs were used for our experiments. They were extracted over a 5-month period, from February through July in 2011. 1M Twitter <message, response> pairs are used for training. With the original response as ground truth, we construct 1M <message, response, label=1> triples as positive instances. Another 1M negative instances <message, response, label=0> are constructed by replacing the ground truth response with a random response in the training set.

For tuning and evaluation, we use 20K <message, response> pairs that constitute the validation set (10K) and test set (10K). They are selected by a criterion that encourages interestingness and relevance: both the message and response have to be at least 3 tokens long and contain at least one non-stopword. For every message, at least one concept has to be found in the commonsense knowledge base. For each instance, we collect another 9 random responses from elsewhere to constitute the response candidates.

Our use of the Twitter dialogue dataset was a common practice in conversational modeling research, as it was a large dataset with real-world conversational data. The use of 20k pairs in the validation and test sets was also a common practice (Lowe et al., 2017).

Preprocessing of the dataset includes normalizing hashtags, "@User", URLs, emoticons. Vocabulary V is built out of the training set with 5 as minimum word frequency, containing 62535 words and an extra $<UNK>$ token representing all unknown words.

4.3.2 ConceptNet

In our experiment, ConceptNet[4] is used as the commonsense knowledge base. ConceptNet was a widely used knowledge base for commonsense reasoning and has been used in previous studies (Weston et al., 2015; Zhou et al., 2016). Preprocessing of this knowledge base involves removing assertions containing non-English characters or any word outside vocabulary V. 1.4M concepts remain. 0.8M concepts are unigrams, 0.43M are bi-grams and the other 0.17M are tri-grams or more. Each concept is associated with an average of 4.3 assertions. More than half of the concepts are associated with only one assertion.

An average of 2.8 concepts can be found in ConceptNet for each message in our Twitter Dialogue Dataset, yielding an average of 150 commonsense assertions (the size of A_x). Unsurprisingly, common concepts with more assertions associated are favored in actual human conversations.

It is worth noting that ConceptNet is noisy due to uncertainties in the constructing process, where 15.5% of all assertions are considered "false" or "vague" by human evaluators [110]. Our max-pooling strategy used in the Tri-LSTM encoder and supervised word embeddings is partly designed to alleviate this weakness.

[4] https://conceptnet.io.

4.3.3 Parameter Settings

In all our models excluding term frequency–inverse document frequency (TF-IDF, Ramos et al., 2003), we initialize word embeddings with pretrained GloVe embedding vectors (Pennington et al., 2014). Following previous settings (Lowe et al., 2016), the size of hidden units in LSTM models is set to 256 and the word embedding dimension is 100. We use stochastic gradient descent (SGD) for optimizing with batch size of 64. We fixed training rate at 0.001 (Kingma and Ba, 2014).

4.3.4 Results and Analysis

The main results for TF-IDF, word embeddings, memory networks and LSTM models are summarized in Table 4.1. We observe that:

(1) LSTMs perform better at modeling dialogues than word embeddings on our dataset, as shown by the comparison between Tri-LSTM and word embeddings.

(2) Integrating commonsense knowledge into conversational models boosts model performance, as Tri-LSTM outperforms Dual-LSTM by a certain margin.

(3) Max-pooling over all commonsense assertions depending on response y is a better method for utilizing commonsense knowledge than attention over memory in our setting, as demonstrated by the gain of performance of word embeddings over memory networks.

TABLE 4.1: Model evaluation. * indicates models with commonsense knowledge integrated. The TF-IDF model is trained following (Lowe et al., 2015). The "Recall@k" method is used for evaluation (Lowe et al., 2016). The model is asked to rank a total of N responses containing one positive response and $N-1$ negative responses ($N = 10$ according to our test set). If the ranking of the positive response is not larger than k, Recall@k is positive for that instance.

Recall@k	TF-IDF	Word Embeddings*	Memory Networks*	Dual-LSTM	Tri-LSTM*	Human
Recall@1	32.6%	73.5%	72.1%	73.6%	**77.5%**	87.0%
Recall@2	47.3%	84.0%	83.6%	85.6%	**88.0%**	-
Recall@5	68.0%	95.5%	94.2%	95.9%	**96.6%**	-

We also analyze samples from the test set to gain an insight on how commonsense knowledge supplements the message itself in response selection by comparing the Tri-LSTM encoder and the Dual-LSTM encoder.

TABLE 4.2: Case studies for the impact of commonsense assertions. "Activated Assertion" is the commonsense assertion entry in A_x chosen by max-pooling. $\diamondsuit$ indicates correct selection. All 4 instances displayed are taken from the test set.

Instance	Message	Response selected by Dual-LSTM
1	i was helping my brother with his chinese.	did yoga help?
2	bonjour madame, quoi de neuf.	yeah me too!
3	help what colour shoes can i wear with my dress to the wedding?	very pale pink or black.$^\diamondsuit$
4	helping mum paint my bedroom.	what color are you going for?$^\diamondsuit$

Instance	Response selected by Tri-LSTM	Activated Assertion (total size of A_x)
1	the language sounds interesting! i really gotta learn it!$^\diamondsuit$	*chinese, IsA, human_language (755)*
2	loool . you can stick with english , its all good unless you want to improve your french.$^\diamondsuit$	*bonjour, IsA, hello_in_french (9)*
3	very pale pink or black.$^\diamondsuit$	*pink, RelatedTo, colour (1570)*
4	what color are you going for?$^\diamondsuit$	*paint, RelatedTo, household_color (959)*

As illustrated in Table 4.2, instances 1,2 represent cases where commonsense assertions as an external memory module provide certain clues that the model without one fails to capture. For example in instance 2, Tri-LSTM selects the response "...improve your french" to message "bonjour madame" based on a retrieved assertion "$bonjour, IsA, hello\_in\_french$", while Dual-LSTM selects an irrelevant response. Unsurprisingly, Dual-LSTM is also able to select the correct response in some cases where certain commonsense knowledge is necessary, as illustrated in instance 3. Both models select "... pink or black" in response to message "...what color shoes...", even though Dual-LSTM does not have access to a helpful assertion "$pink, RelatedTo, color$".

Informally speaking, such cases suggest that to some extent, Dual-LSTM (a model with no external knowledge) is able to encode certain commonsense knowledge in model parameters (e.g., word embeddings) in an implicit way. In other cases, e.g., instance 4, the message itself is enough for the selection of the correct response, where both models do equally well.

4.4 Related Work

As an early research effort on developing dialogue systems that are conditioned on commonsense knowledge, our work was inspired by previous work along related directions. At the same time, we gladly notice following work that further explored the relationship between knowledge and dialogue systems. This section discusses both categories of related work.

Preceding work

We start with a discussion about the grounding works that our effort was based upon. The use of an external memory module in NLP tasks had received considerable attention, such as in question answering (Weston et al., 2015) and language modeling (Sukhbaatar et al., 2015). It had also been employed in dialogue modeling in several limited settings. With Memory Networks, Dodge et al. (2015) used a set of fact triples about movies as long-term memory when modeling reddit dialogues, movie recommendation and factoid question answering. Similarly in a restaurant reservation setting, Bordes and Weston (2016) provided local restaurant information to the conversational model. Researchers had also proposed several methods to incorporate knowledge as external memory into the Seq2Seq framework. Xing et al. (2016) incorporated the topic words of the message obtained from a pre-trained LDA model into the context vector through a joint attention mechanism. Ghazvininejad et al. (2017) mined FoodSquare tips to be searched by an input message in the food domain and encoded such tips into the context vector through one-turn hop. The Tri-LSTM model we proposed in this chapter shares similarities with Lowe et al. (2015), which encoded unstructured textual knowledge with RNN. Our work distinguished itself from previous research in that we considered a large heterogeneous commonsense knowledge base in the open-domain retrieval-based dialogue setting.

Following work

Next we discuss the research done by the community that followed our work.

Immediately following our work was an attempt on grounding generation-based dialogue systems on knowledge graphs (Zhou et al., 2018). During dialogue generation, the model attentively reads the relevant knowledge graphs and the knowledge triples within each graph to improve responses through a dynamic graph attention mechanism.

As opposed to LSTMs, recent effort on building dialogue systems was based on transformers usually drew inspiration from the success of large-scale pretraining and either fine-tuned their dialogue models using large-scale pretrained language models as backbones (Ham et al., 2020) or used large-scale dialogue data for pre-training (Zhang et al., 2019; Adiwardana et al., 2020). Such models are often hundreds or even thousands times as large as earlier models. Such drastic increase in model size begs the question: is it possible to implicitly learn to converse with commonsense knowledge by encoding it in the model parameters without explicitly referring to an external knowledge base? After all, large language models have been shown to be able to learn commonsense knowledge implicitly (Zhou et al., 2020).

GPT4 (OpenAI, 2023) made strides forward on this aspect by almost reaching human performance on a challenging commonsense reasoning dataset (Zellers et al., 2019). Considering the size of GPT4, which is presumably upwards of 350 GB (size of GPT3), and the size of a typical commonsense dataset (such as ConceptNet, which is less than a few GB), it is not unimaginable that commonsense knowledge might be better encoded directly by a neural network as opposed to existing as an external knowledge base.

However, coupling dialogue systems with general information retrieval engines or other types of tools can still be beneficial. They can be seen as a new type of TOD systems that are not necessarily based on entity-centric databases as in Chapter 2. In such cases, the boundary between ODDs and TODs is blurred because such systems can often perform the two utilities together reasonably well. We discuss examples below.

Blenderbot2 (Komeili et al., 2021) improved upon its predecessor Blenderbot (Roller et al., 2020) by incorporating a search engine. During conversation, the model can generate contextual internet search queries, read the results, and incorporate that information when responding to the user's messages. This important knowledge-grounding technique enables the model to stay up-to-date in an ever-changing world.

SeeKeR (Search engine→Knowledge→Response) (Shuster et al., 2022) proposed to apply a single language model for the 3 modular tasks that ground dialogue responses on knowledge: search, generating knowledge, and generating a final response.

LaMDA (Language Models for Dialog Applications) (Thoppilan et al., 2022) took a step forward by coupling the dialogue system with a toolset containing an information retrieval system, a calculator, and a translator. The dialogue system is capable of consulting all 3 external information sources at will in real-time.

Most recently, closed-source models such as Bard (Google, 2023) and Bing Chat (Microsoft, 2023) have landed reasonably successfully in large-scale real-world application.

We note that search engines are very often much more up-to-date than dialogue data collected under restricted scenarios. Therefore, by introducing a real-time information retrieval engine to the dialogue system, the temporal generalization problem (Lazaridou et al., 2021) is effectively alleviated for response generation. Essentially, instead of being constrained by the static training examples, the dialogue system can reflect the most recent version of human knowledge.

It is also worth noting that the knowledge grounding and multimodality grounding (Chapter 3) can be combined. For example, Yang et al. (2021) proposed a new dataset MMDialKB that conditions end-to-end TOD systems on multimodal knowledge bases.

4.5 Chapter Summary and Future Prospects

In this chapter, we emphasized the role of external knowledge in conversational models. In the ODD setting, we illustrated our effort on experimenting with commonsense knowledge as external memory and proposed a method of using the LSTM to encode commonsense assertions to enhance response selection.

Although the gains presented by our new method were not spectacular according to recall@k, our view represented the first attempt at integrating a large commonsense knowledge base that potentially describes the external world into conversational models as a memory component.

The massive research effort put forward by the community that followed our work helped make great advance towards the goal of building knowledge-grounded dialogue systems. However, the current state-of-the-art systems still fall short compared to humans in many ways. While external knowledge improves output groundedness, the model can still generate responses that do not *accurately* reflect the contents of authoritative external sources. While the system can generally respond to a simple question based on a single knowledge snippet, it often makes mistakes given a context that requires sophisticated reasoning (Yasunaga et al., 2021). These issues are alleviated by ChatGPT and GPT4 but still call for further research in the future.

The last 3 chapters have demonstrated our efforts on improving the capabilities of dialogue models such that they can generate better responses. Orthogonal to capabilities, another aspect that impacts the usability of dialogue systems is inference efficiency. We show an early attempt of ours on improving that in the next chapter.

Chapter 5

Improving the computational efficiency of large-scale response retrieval

The usability of a dialogue system in application is determined by the quality of its response and the computational cost of producing one. The last 3 chapters have discussed various ways to make the produced response more grounded and useful. In the chapter of this thesis, we touch on the computational efficiency aspect.

Generally speaking, large neural networks have been the state-of-the-art machine learning models in recent years. Yet they can be slow to make inference with. When they are used for dialogue systems, their slow inference speed can sometimes make real-time application impossible. While this issue exists for both generation and retrieval-based dialogue systems, this chapter focuses on our early effort in 2019 on addressing this issue in the context of retrieval-based dialogue systems. We discuss more recent efforts on generation-based dialogue systems based on large language models in Section 5.4.

Specifically, we note that strong retrieval-based dialogue systems that are based on large pre-constructed natural response candidate sets can produce diverse and controllable responses. However, a large candidate set could be computationally costly, as every response candidate needs to be paired with the input context for scoring and ranking. In this chapter, we address the research question "how can we accelerate large-scale response retrieval?" We propose methods that support fast and accurate response retrieval systems that can operate on large-scale response candidate sets.

We utilize a computationally efficient dual encoding scheme in which contexts and responses are encoded into a sentence embedding space individually, where inner product is used for scoring. To boost accuracy, we adopt a knowledge distillation approach where a very strong yet computationally expensive joint encoding model is used to facilitate training our encoders. We then significantly boost the retrieval speed by adopting a learning-based candidate screening method that predicts a subset from the whole response candidate set. We show in the experiments that our model performs strongly in terms of retrieval accuracy and human evaluation. At the same time our retrieval speed is improved by orders of magnitude.

5.1 Introduction

Trained on a large set of natural context-response pairs, retrieval-based models attempt to select the most appropriate response from a response candidate set based on a scoring model that indicates the compatibility of a context and a response (Lowe et al., 2015; Boussaha et al., 2019). Compared to generation-based models, retrieval-based models have the advantage of being more controllable, since the responses come from a pre-constructed response set. Given a strong context-response pair scoring model, a retrieval-based system can be expected to produce long, interesting and diverse responses. The size and quality of the response candidate set matter a lot. Intuitively, a larger response set increases the possibility of finding a suitable response, especially for open-domain chitchat (Figure 5.1). However, the size of the response set heavily affects retrieval speed, as the context needs to be paired with every response for scoring. Sometimes the retrieval time is linear with respect to the size of the candidate set, which makes some accurate but cumbersome retrieval networks practically infeasible (Humeau et al., 2019). For example, the "sentence pair classification" setup in the BERT model (Devlin et al., 2018) can be directly applied to the retrieval-based dialogue scenario. The context and response are concatenated and fed into the transformer for scoring. However, pairing every response candidate with the context and forwarding them through the network during inference is computationally intractable for large response candidate sets. Another way is to encode the context and response separately with

2 transformers of the same architecture and use their respective embeddings for scoring. Yet separate encoding loses accuracy.

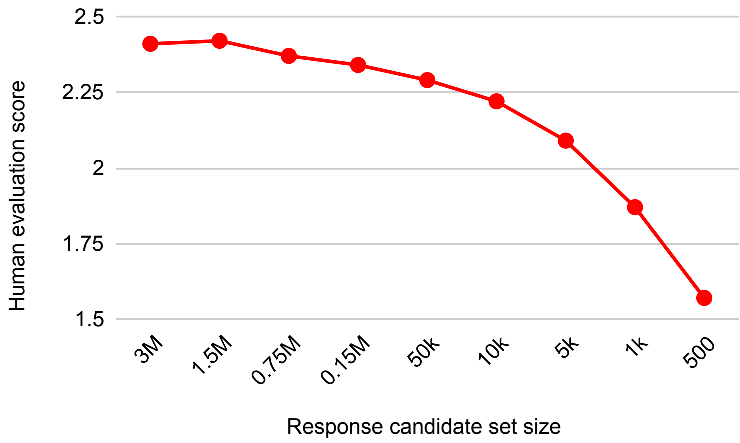


FIGURE 5.1: Size of the response candidate set VS Response quality (human evaluation using the model of this paper's contribution as in Section 5.3.6).

Our work is driven by the idea of developing methods to achieve fast large-scale response retrieval while maintaining competitive accuracy. In our work, deep transformers are used to encode contexts and responses individually into a sentence embedding space, where inner product is used for scoring. This separate encoding scheme allows for much faster retrieval. To make up for the lost accuracy, we adopt a knowledge distillation approach where scores from the strong yet computationally expensive joint encoding model are used to facilitate the training process of our encoders. We further boost our retrieval speed by adopting a learning-based screening method that predicts a response candidate subset that the best response lies in based on the context. The subset is much smaller than the whole candidate set. Therefore, the retrieval time is drastically reduced.

The main contributions that we demonstrate in this chapter are:

(1) We propose a fast large-scale response retrieval system based on knowledge distillation and deep transformer encoders. Knowledge distillation from a joint encoding model is performed for accuracy boosting.

(2) A learning-based screening method is proposed for optimal response search based on maximum inner product in the embedding space, which enables very efficient response retrieval from large candidate sets.

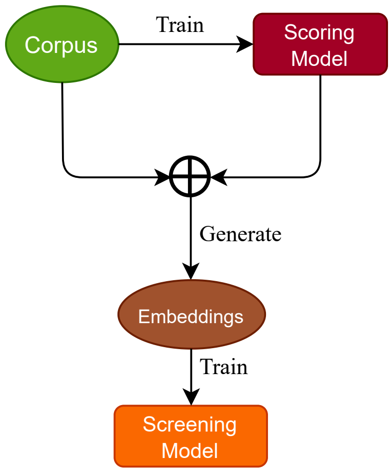


FIGURE 5.2: Pipeline overview.

(3) Extensive experiments on both single-turn and multi-turn conversation settings show that our model performs favorably compared with strong retrieval-based baselines in terms of accuracy and speed trade-off.

5.2 Proposed Approaches

5.2.1 Overview

A pipeline of the whole framework is shown in Figure 5.2. We have the scoring model (which contains the encoding model) that encodes the context and response into embeddings and assigns a matching score. It is trained on the corpus of context-response pairs. The screening model tries to predict a response subset given the context, it does so based on their embeddings. Thus the screening model is trained on embeddings produced by a fully trained scoring model run on the corpus.

5.2.2 Knowledge Distillation

One state-of-the-art method for response retrieval is to feed the concatenation of the context c and the response r through a deep transformer for joint encoding, termed the *Cross-Encoder* (Figure 5.3a). The scoring function is defined as below.

$$score_{\text{cross}} = \mathbf{FFN}(\mathbf{DT}([c; r])), \tag{5.1}$$

where $[;]$ indicates concatenation and **FFN** is a feed forward network with Sigmoid activation. **FFN** reduces an embedding to a scalar score. **DT** stands for the deep transformer encoder, which encodes a word array into an embedding.

This encoding scheme allows the response to interact with input context in the deep transformer, which leads to high accuracy. However, the fact that during inference, the context needs to be paired with every response candidate for encoding and scoring, makes the retrieval process far too slow for large candidate sets.

In contrast, the *Dual-Encoder* (Figure 5.3b) encodes the context and response separately, and uses the inner product between the two embeddings for scoring (Equations 5.2-5.4). It loses accuracy as it does not allow direct interaction between the context and the response in the transformer. However, with Dual-Encoders, we are able to cache the encoded candidate response embeddings, and reuse them for each new context. This results in significantly faster prediction.

$$\mathbf{c} = \mathbf{DT}(c) \tag{5.2}$$

$$\mathbf{r} = \mathbf{DT}(r) \tag{5.3}$$

$$score_{\text{dual}} = sigmoid(\mathbf{c}^T\mathbf{r}), \tag{5.4}$$

where $\mathbf{c}, \mathbf{r} \in \mathbb{R}^D$ are the output embedding vectors of the deep transformer encoders. D is the embedding dimension. To keep the Dual-Encoder's speed while boosting its accuracy, we adopt knowledge distillation. In our case, the scores given by the fully optimized Cross-Encoder are used to facilitate the training of the Dual-Encoder. Specifically, the loss used to train the new Dual-Encoder model (denoted as *Dual-Encoder-KD*, Figure 5.4) is the sum of the traditional cross entropy loss based on the ground-truth labels and the L2 loss between the Dual-Encoder-KD score and the Cross-Encoder score, as shown below.

$$
\begin{aligned}
L_{\text{dual-kd}} = \beta(score_{\text{dual}} - score_{\text{cross}})^2 + \\
BCE(score_{\text{dual}}, label),
\end{aligned}
\tag{5.5}
$$

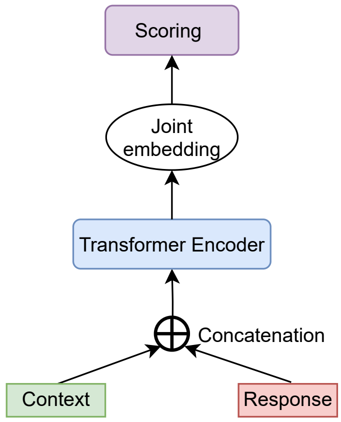
where β is a weighting coefficient, and BCE is the traditional binary classification loss.

To recap, our motivation for employing knowledge distillation is as follows:

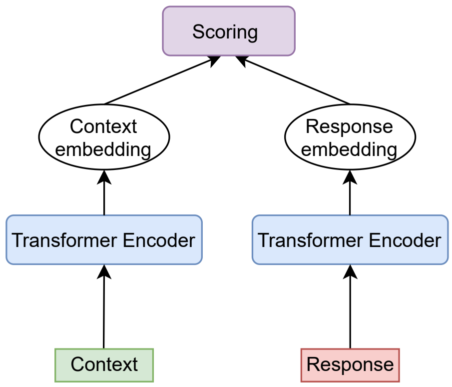
Retrieval-based dialogue systems involve searching through large datasets to find the most appropriate response. Traditional approaches like Cross-Encoders, which are highly accurate, are computationally expensive, making them impractical for real-time applications. Dual-Encoders, on the other hand, are more computationally efficient but do not achieve the same level of accuracy.

To maintain the Dual-Encoder's computational efficiency while enhancing its accuracy, we employ knowledge distillation. Knowledge distillation is a technique in which a smaller, less complex model (the student model) learns from a larger, more complex model (the teacher model). This approach enables the student model to acquire the teacher model's knowledge while retaining its own computational advantages.

In our case, the fully optimized Cross-Encoder serves as the teacher model, providing scores that guide the training of the Dual-Encoder, which acts as the student model. By utilizing the information from the Cross-Encoder, the Dual-Encoder can achieve higher accuracy than it would by training alone. This knowledge transfer process enables the Dual-Encoder to capture more intricate relationships between the context and the response, while still maintaining its advantage in terms of speed.



(A) Cross-Encoder



(B) Dual-Encoder

FIGURE 5.3: Two baseline encoding schemes that differ greatly in accuracy and speed.

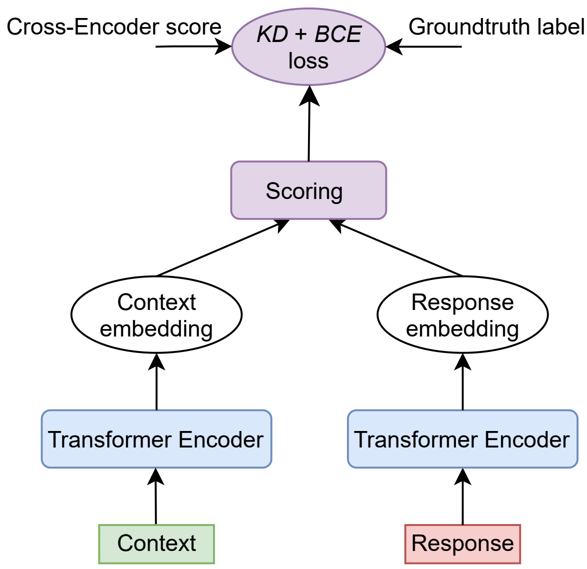


FIGURE 5.4: Dual-Encoder with Knowledge Distillation.

5.2.3 Learning-based Candidate Screening

Dual-Encoder allows encoding contexts and responses separately. Therefore, the retrieval time is drastically reduced by pre-computing and caching all the response candidate embeddings. According to Equation 5.4 and the monotonicity of the Sigmoid function, the best response in the response candidate set is found by

$$\hat{r} = \arg\max_{r \in R} \mathbf{c}^T \mathbf{r}, \tag{5.6}$$

where R is the whole response candidate set. We further accelerate this maximum inner product search (MIPS) process. This algorithm uses a light-weight screening model to predict a much smaller set of candidate responses given the context, and then find the best response within that subset with vanilla calculation.

This screening method has two sets of parameters - a set of K context cluster centroids $\{\mathbf{a}_1, ..., \mathbf{a}_K\} \in \mathbb{R}^D$, and the corresponding response subsets $\{\mathbf{s}_1, ..., \mathbf{s}_K\} \in \{0, 1\}^N$, which are binary representations for which responses belong in the subset. N is the total number of response candidates. A new context is assigned to a context cluster based on inner product with the centroids, and the corresponding response subset is the screening model's prediction.

The probability of a context $\mathbf{c}_i$ belonging to context cluster k is modeled by

$$\mu_{ik} = \frac{exp(\mathbf{c}_i^T \mathbf{a}_k)}{\sum_l exp(\mathbf{c}_i^T \mathbf{a}_l)} \tag{5.7}$$

The probability of retrieving $\mathbf{r}_j$ for context $\mathbf{c}_i$ is

$$p_{ij} = \sum_k \mu_{ik} \mathbf{s}_k[j], \tag{5.8}$$

where $\mathbf{s}_k[j] \in \{0, 1\}$ denotes the jth element of $\mathbf{s}_k$. It indicates whether the kth cluster's subset includes response $\mathbf{r}_j$. Essentially, equation 5.8 describes p_{ij} as the sum of the existence of $\mathbf{r}_j$ across all subsets, weighted by cluster assignment.

The groundtruth best candidate label for $\mathbf{c}_i$ and $\mathbf{r}_j$ is $y_{ij} \in \{0, 1\}$, which indicates whether the best response for $\mathbf{c}_i$ is $\mathbf{r}_j$. Our loss is defined by

$$l_{ij} = \begin{cases} \lambda p_{ij} & y_{ij} = 0 \\ 1 - p_{ij} & y_{ij} = 1 \end{cases} \tag{5.9}$$

Another way to write it is:

$$l_{ij} = \lambda p_{ij}(1 - y_{ij}) + (1 - p_{ij})y_{ij} \tag{5.10}$$

When $\mathbf{r}_j$ is the best response for $\mathbf{c}_i$, the model is punished by $1 - p_{ij}$. Otherwise it is punished by p_{ij}. $\lambda \in [0, 1]$ is the balancing coefficient. It controls how much the screening model values speed vs. accuracy. Intuitively, the downside of including a single redundant candidate in the subset (e.g., $y_{ij} = 0$ and $p_{ij} = 1$) is much smaller than completely missing the ground-truth candidate (e.g., $y_{ij} = 1$ and $p_{ij} = 0$), thus λ is set to be much smaller than 1. The overall optimization goal is thus the sum over all contexts and response candidates:

$$\underset{\{\mathbf{a}_k\}_{k=1}^K, \{\mathbf{s}_k\}_{k=1}^K}{minimize} \; L = \sum_i \sum_j l_{ij} \tag{5.11}$$

To solve this optimization problem, we use alternating minimization. We optimize $\{\mathbf{a}_k\}_{k=1}^K$ and $\{\mathbf{s}_k\}_{k=1}^K$ alternatively while keeping the other one fixed. First, with $\{\mathbf{a}_k\}_{k=1}^K$ fixed, L can be expressed as

$$L = \sum_k \sum_j \alpha_{kj} \mathbf{s}_k[j] + \sum_i \sum_j y_{ij} \tag{5.12}$$

through Equation 5.8 and 5.10, and α_{kj} is the coefficient for $\mathbf{s}_k[j]$:

$$\alpha_{kj} = \sum_i \mu_{ik}[\lambda - (\lambda + 1)y_{ij}] \tag{5.13}$$

Remember $\mathbf{s}_k[j] \in \{0, 1\}$. To minimize L, $\mathbf{s}_k[j]$ is set to 0 if $\alpha_{kj} > 0$, otherwise it is set to 1.

Then we fix $\{\mathbf{s}_k\}_{k=1}^K$ and update context cluster centroids $\{\mathbf{a}_k\}_{k=1}^K$ (continuous, as opposed to the concrete $\{\mathbf{s}_k\}_{k=1}^K$) using stochastic gradient descent (SGD) based on Equation 5.11. We avoid the need to use the gumbel trick (Jang et al., 2016) as in Chen et al. (2018) by having defined the loss with probabilities. Following Chen et al. (2018), we initialize context cluster centroids $\{\mathbf{a}_k\}_{k=1}^K$ with spherical k-means clustering. The overall learning process is given in Algorithm 1.

Algorithm 1: Learning-based screening algorithm

Input: Context embeddings $\{\mathbf{c}_i\}_{i=1}^M$, response candidate embeddings $\{\mathbf{r}_j\}_{j=1}^N$, and labels $\mathbf{Y} \in \{0, 1\}^{M \times N}$ indicating the best responses for each context given by exact inner product calculation according to Equation 5.6.
Hyperparameter: Number of clusters K, balancing coefficient λ, number of iterations T.
Output: K context cluster centroids $\{\mathbf{a}_k\}_{k=1}^K$, K response subsets $\{\mathbf{s}_k\}_{k=1}^K$.
Training process: Initialize $\{\mathbf{a}_k\}_{k=1}^K$ by running spherical k-means clustering on $\{\mathbf{c}_i\}_{i=1}^M$, Initialize $\{\mathbf{s}_k\}_{k=1}^K$ with 0's.
Execute the following procedures alternatively for T iterations.
(1) Keep context cluster centroids $\{\mathbf{a}_t\}_{t=1}^K$ fixed and update response subsets $\{\mathbf{s}_t\}_{t=1}^K$ using Equation 5.12.
(2) Keep response subsets $\{\mathbf{s}_t\}_{t=1}^K$ fixed and update context cluster centroids $\{\mathbf{a}_t\}_{t=1}^K$ using SGD based on Equation 5.11.

5.3 Experiments and Results

5.3.1 Datasets

Three datasets are used for our experiments. The first comes from Reddit online chat curated by [71]. It contains conversations from discussion threads on a variety of topics on the website Reddit. The second is the DailyDialog dataset (Li et al., 2017), which contains human-written daily conversations that cover various topics.

The third is the ConvAI2 dataset (Dinan et al., 2019), which is based on the PersonaChat dataset (Zhang et al., 2018). At the time when our research was conducted, these 3 datasets were suitable for our purposes because of their sizes and their nature of being open-domain conversations. The partitioning of the 3 datasets for the scoring models is shown in Table 5.1.

TABLE 5.1: The partitioning of the datasets for the *scoring* models.

Dataset	DailyDialog	ConvAI2	Reddit
No. Training	75000	131438	3300000
No. Validation	5000	7800	30000
No. Testing	1000	Unreleased	10000

TABLE 5.2: The origin of contexts and responses used for the *screening* model. They are taken from the Reddit partitions for the *scoring* models as in Table 5.1. Yes/no indicates whether they come from the respective partitions.

Partitioning for scoring models	No. instances	Training contexts for screening	Testing contexts for screening	Screening candidate responses
Training	3300000	yes	no	random 160k
Validation	30000	yes	no	yes
Testing	10000	no	yes	yes
Total instances for screening		3330000	10000	200000

Since our goal is to train a scoring model on whether or not a (context, response) pair is matched, we need negative pairs alongside the positive pairs that naturally exist in the dataset. For each positive (context, response) instance in the training and validation sets, we create a negative instance by replacing the ground-truth response with a random response in the training set.

For scoring-based retrieval models, a popular way to quantify their accuracy is to test how well they can identify the ground-truth response among distractors. Thus for Reddit and DailyDialog we build each instance in the test set by mixing the ground truth with 9 distractors. We then measure the frequency of the model scoring the ground-truth candidate higher than all other 9 distractor candidates, which is termed Recall@1/10. For ConvAI2, we report on the validation set since the test set is not released. It is constructed for Recall@1/20 evaluation.

5.3.2 Model Training Details

We use the pre-trained BERT model based on Devlin et al. (2018) and Wolf et al. (2019) as our deep transformer encoder. Its output embedding length is 768. Our hyper-parameters are optimized with grid search. We train the Dual-LSTM (replacing the transformers in Dual-Encoder with LSTMs) with knowledge distillation from Cross-Encoder (Dual-LSTM-KD) (Tang et al., 2019) and re-implement the Poly-Encoder (Humeau et al., 2019) as additional baseline models as they were proposed as similar efforts to improve inference speed while maintaining good accuracy. The Poly-Encoder framework uses global-level self-attention on multiple context embeddings and the response embedding. We also run Poly-Encoder-KD as another reference, i.e., Poly-Encoder enhanced with knowledge distillation as in Dual-Encoder-KD.

For all the models on DailyDialog and Reddit, the best learning rate is searched for from options $\{5e{-}6, 1e{-}5, 5e{-}5\}$. For the Poly-Encoder model, an extra hyperparameter "code length" (Humeau et al., 2019) is searched for from $\{16, 64, 128, 256\}$. For knowledge distillation, our weighting parameter β comes from $\{0.2, 0.5, 1\}$. We use the Adam optimizer (Kingma and Ba, 2014). For ConvAI2, we report on the official validation set. Since this leaves us no practical validation set, the hyperparameters for ConvAI2 are simply the same as the best performing ones for Reddit.

We run our candidate screening model with different hyperparameters to find the best speed accuracy trade-off point. The number of clusters K is from $\{10, 20, 50\}$ and the balancing coefficient λ that balances accuracy and speed is from $\{1e{-}5, 5e{-}6, 1e{-}6, 5e{-}7\}$. Since the purpose of the screening model is to improve speed for large-scale retrieval, we evaluate on the relatively large Reddit dataset. As our screening model requires supervised learning, it is most effective when the number of training contexts M (conceptually similar to no. data points in multi-class classification) is significantly larger than the number of response candidates N (similar to no. class categories). Therefore, we use 200k responses (including 160k from the scoring model training partition, and both validation and test partitions. See Table 5.2) in the dataset as our response candidate set $\{\mathbf{r}_j\}_{j=1}^{N}$. We leave 10k contexts (from the test partition) for testing and use all the rest in the dataset as the training contexts $\{\mathbf{c}_i\}_{i=1}^{M}$ to train our screening model.

5.3.3 Retrieval accuracy of the scoring models

We conduct automatic evaluations on how well our scoring models can tell suitable responses apart from distractors. The retrieval accuracy results are shown in

Table 5.3. The results suggest that on all three datasets, our Dual-Encoder-KD outperforms the baseline models with significant margins. Yet, it still falls short in comparison to its teacher model, that is Cross-Encoder. Poly-Encoder-KD and Dual-Encoder-KD perform very closely, in contrast to the stark difference between Poly-Encoder and Dual-Encoder. One possible explanation for this is that the supervision signal from Cross-Encoder is useful in a way that renders the more sophisticated architecture of Poly-Encoder less impactful.

Note that Recall@1/10 and Recall@1/20 can only test how reliable a scoring model scores context-response pairs, and cannot be used for evaluation, when a screening model is used. Because a screening model is only useful when a very large response candidate set is involved.

TABLE 5.3: Recall@1/10 results for Reddit and DailyDialog and Recall@1/20 results for ConvAI2.

Dataset	Reddit	DailyDialog	ConvAI2
Dual-Encoder	0.814	0.710	0.802
Cross-Encoder	**0.882**	**0.767**	**0.831**
Poly-Encoder	0.832	0.726	0.815
Poly-Encoder-KD	0.859	0.741	0.824
Dual-LSTM-KD	0.810	0.703	0.801
Dual-Encoder-KD	**0.857**	**0.743**	**0.822**

5.3.4 Accuracy and speed trade-off in candidate screening

For retrieving the best response from a large candidate set for a context, our candidate screening model drastically reduces the candidate set size at the cost of sometimes disregarding the best response. To quantitatively determine the performance of the model, we measure (1) *speedup ratio*, defined as the expected ratio of the original candidate set size to the reduced candidate set size and (2) *accuracy*, defined as the frequency of the reduced candidate set containing the best response in the original candidate set according to exact maximum inner product calculation.

We vary hyperparameters to achieve a good speedup ratio and accuracy trade-off. As shown in Figure 5.5, in general, more speedup means less accuracy. As expected from Equation 5.9, the higher λ is, the more the model values speed. Our experiments suggest the more clusters there are, the smaller each subset is, and the faster and less accurate the model is. At $\lambda = 1e{-}6$ and $K = 20$, we have a $5.14\times$ speedup with a 1.3% accuracy loss. This is the configuration we use in later experiments.

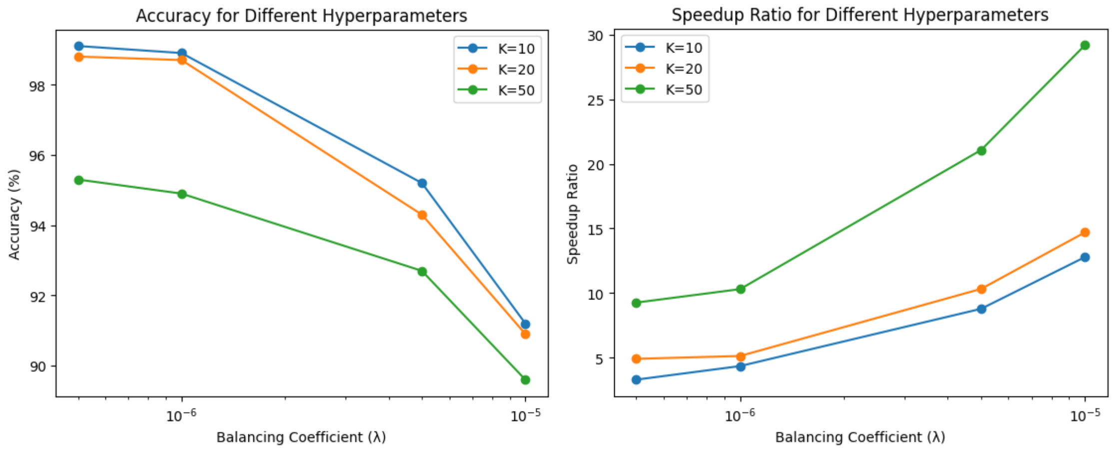


FIGURE 5.5: Accuracy and Speedup Ratio for Different Hyperparameters.

5.3.5 Retrieval speed in wall clock time

We further measure the retrieval speed of the different models that have been discussed with wall clock time. Table 5.4 shows the average time it takes to retrieve a response from 200k candidates for a single context. Cross-Encoder is orders of magnitude slower than the rest. Dual-Encoder-KD outperforms Poly-Encoder. Our candidate screening model further increases the speed by about 5 times.

The retrieval process of Dual-Encoder-KD contains 2 steps: (a) Encode the context into an embedding using the encoder model and (b) Find the best response by running the context embedding against cached candidate response embeddings. In our experiment we find that step (a) takes less than 5% as much time as step (b), due to the fact that the size of the candidate set is relatively large. With our screening model targeting time saving in step (b), the speedup achieved in wall-clock time by Dual-Encoder-KD-Screening compared to Dual-Encoder-KD is approximately the same as the "speedup ratio" in Section 5.3.4.

Table 5.4: Time in milliseconds to retrieve a response from 200k candidates. CPU computations were run on a 20 core Intel Xeon E5-2698v4. GPU computations were run on a single Tesla P100 with cuda 9.0. Dual (Poly)-Encoder and Dual (Poly)-Encoder-KD have the same speed as they share the same architecture.

Model	Retrieval time	
	CPU	GPU
Cross-Encoder	90.5k	8.3k
Poly-Encoder	90.2	12.3
Dual-Encoder-KD	22.5	6.9
Dual-Encoder-KD-Screening	4.1	1.7

5.3.6 Human evaluation on response quality

For subjective human evaluation, we resorted to AMT workers to directly score the quality of a single response produced by the system given the context. 200 random instances in the test set of Reddit were used. Each one was judged by 3 random judges that participated through AMT. Score levels and their meanings are ["3 - Very natural and appropriate", "2 - Somewhat relevant", "1 - Completely irrelevant"]. We treat the scores as numerical values and calculate their mean as a model's human evaluation response quality. Table 5.5 shows the results. We see that human evaluation results for the retrieval-based models roughly align with Recall@1/10 results as in Section 5.3.3. Cross-Encoder performs the best by a small margin. It is followed by Dual-Encoder-KD, Poly-Encoder-KD and Dual-Encoder-KD-Screening, whose scores are very close to each other. Performing our screening method on Dual-Encoder-KD affects the score very marginally. Dual-Encoder-KD outperforms Dual-Encoder and Dual-LSTM-KD by a relatively large margin.

Table 5.5: Human evaluation scores.

Model	Avg. score
Cross-Encoder	**2.44**
Poly-Encoder	2.35
Dual-Encoder	2.29
Dual-LSTM-KD	2.26
Poly-Encoder-KD	**2.43**
Dual-Encoder-KD	**2.39**
Dual-Encoder-KD-Screening	**2.40**

5.4 Related Work

The methods proposed in this chapter were inspired by previous works on knowledge distillation and maximum inner product search, which we review under "Preceding work" below. We also review recent efforts on improving the inference efficiency of generation-based dialogue systems, especially based on the advance of large language models under "Following work".

Preceding work

We first discuss preceding work on Knowledge distillation in neural networks.

Distilling knowledge from a high-accuracy network into a low-accuracy network (Hinton et al., 2015) has proven to be an effective way to improve the accuracy of the latter. Traditionally the latter (student network) tries to mimic the former (teacher network) by minimizing a loss defined between the outputs of the two, in addition to the traditional loss based on groundtruth labels. As the multi-class output of the teacher network has higher entropy than the traditional one-hot labels, the student network has access to an information-rich similarity structure over data. The student network *usually* has the advantage of being smaller, which makes training and inference faster.

Research in various areas has shown the effectiveness of this approach. Kim and Rush (2016) successfully boosted the inference speed of state-of-the-art machine translation networks by about 10 times with little loss in performance. It also found success in computer vision areas such as object detection (Chen et al., 2017) and semantic segmentation (Liu et al., 2019).

Large-scale pretrained language models such as BERT are great teacher models. Recently there have been numerous efforts to distill knowledge from them and make new models that are smaller and faster. TinyBERT (Jiao et al., 2019) and BERT-PKD (Sun et al., 2019) distill knowledge from BERT through its embedding layers, hidden states and attention matrices into a smaller transformer-based model that is similar to BERT. Tang et al. (2019) showed that distilling knowledge from large-scale transformers into an LSTM makes the LSTM more competitive on sentence-level tasks.

The works mentioned above view the student model as a neural network of smaller size. Our distillation approach, proposed for the specific scenario of large-scale response retrieval, is different. It is used on two models that contain the same original BERT architecture but with different representation formats of the input (context and response). For the teacher model, they are encoded together using one BERT, resulting in intractable inference and high accuracy. For the student model, two BERTs are used to encode context and response separately, resulting in faster inference but lower accuracy (Figure 5.3). Experiments suggest that knowledge distillation also improves performance in our setting.

We next discuss preceding work on maximum inner product search.

Representing the context and response candidates in the embedding space separately is a time-efficient approach. With the matching criterion defined as inner product, the problem of finding the optimal response is reduced to a MIPS problem. Methods that accelerate MIPS have been studied in the setting of Neural Language Models. Neural Language Models consist of a softmax layer over a large vocabulary, which also poses as a MIPS problem during inference. Zhang et al. (2018) reduced MIPS to nearest neighbor search and showed a graph-based approach that performs well. The result of Zhang et al. (2018) was later surpassed by Chen et al. (2018). They proposed a learning based approach that assign vocabulary subsets to hidden state clusters, which inspired our work's screening model.

Following work

Although our current chapter focused on accelerating retrieval-based dialogue systems, recent advances on large language models like GPT4 (OpenAI, 2023) and fine-tuning them based on human feedback (Ouyang et al., 2022) have made generation-based dialogue systems tremendously successful (OpenAI, 2022). We thus review efforts on improving the efficiency of generation-based dialogue systems based on large language models in this section.

From the aspect of reducing the model size without losing too much accuracy, two popular methods are weight quantization and knowledge distillation.

Quantization is a post-training technique used to compress the size of large language models, such as GPT4, without significantly compromising their performance. It involves converting the continuous weights (floating-point values) of a model into a smaller set of discrete values, typically using fewer bits. For example, Smoothquant (Xiao et al., 2022) uses a smooth function to approximate the non-linear activation functions in the model, which helps to preserve the accuracy of the model after quantization.

Knowledge distillation on large language models has recently taken the form of naive data distillation. GPT4All (Anand et al.) is a chatbot that is trained over a massive curated corpus of assistant interactions including word problems, story descriptions, multi-turn dialogue, and code, which is distilled from OpenAI APIs.

There are other inference-time tricks that accelerate generation without changing the model.

In application scenarios where retrieval from reference text is very common, e.g., with search engines and in multi-turn conversations, Yang et al. (2023) proposed LLMA, an accelerator to losslessly speed up large language model inference with references. LLMA is motivated by the observation that there are abundant identical text spans between the decoding result by a large language model and the reference text. LLMA first selects a text span from the reference and copies its tokens to the decoder and then efficiently checks the tokens' appropriateness as the decoding result in parallel within one decoding step. The improved computational parallelism allows LLMA to achieve over $2\times$ speed-up. LLMA represents accelerating dialogue systems that are based on additional knowledge as mentioned in Chapter 4.

On the prompt design aspect, simple batch prompting, i.e., prompting groups of multiple samples in one batch and letting the model generate multiple responses for the batch in inference (Cheng et al., 2023) proved helpful.

5.5 Chapter Summary and Future Prospects

In this chapter, we presented methods for a fast large-scale response retrieval model for human-computer interaction. Based on deep transformer encoders, we used knowledge distillation to leverage the learning power of a cumbersome joint encoding model to improve the performance of our fast individual encoders. Furthermore, to better handle large response candidate sets, we proposed a learning-based screening model that makes the retrieval process about 5 times faster with very little accuracy loss. Finally, we demonstrated a pipeline that performs strongly in terms of speed and quality trade-off compared with other retrieval-based models.

Our work focused on the retrieval-based scenario and was accompanied by other efforts on the generation-based scenario around the same time period. From the standpoint of 2023, we expect future work on dialogue systems to focus on the generation-based scenario more often due to their flexibility and due to the recent progress demonstrated by ChatGPT and GPT4. Apart from knowledge distillation and the directions illustrated under "Following work", another promising direction for reducing the computational cost of neural networks is through sparse activation (Fedus et al., 2021). Similar to the brain selectively activating different brain regions given different prompts, sparsely activating a large neural network would drastically reduce the computational cost.

Some of the accelerating methods discussed in this chapter, either retrieval-based or generation-based, can potentially be migrated to task-oriented dialogue systems (Chapter 2) and multimodal dialogue systems (Chapter 3). But we also expect to see more acceleration methods specifically designed for those as well in the future.

Chapter 6

Summary and future directions

This chapter aims to summarize the main chapters of this thesis, and then present future directions. The most recent work covered in this thesis, discussed in Chapter 2, was done in late 2021. Most recently, the field of dialogue systems was revolutionized by ChatGPT (OpenAI, 2022) and GPT4 (OpenAI, 2023).

After briefly summarizing the thesis in Section 6.1, this chapter splits the future work discussions into two parts. The first part focuses on the future work that naturally stems from the content of the thesis, which is covered in Section 6.2. Then Section 6.3 "ChatGPT and GPT4" discusses the recent tremendous advance brought forward by scaling language models and adapting them for dialogues. Lastly, the second part of future work discussions, i.e., Section 6.4 "Future work from GPT4" will discuss how the community might move forward given the demonstrated success of GPT4.

6.1 Summary of the Thesis

This thesis demonstrated our effort on innovating dialogue systems on multiple aspects, including (1) fusing different utilities, (2) fusing different modalities, (3) grounding dialogues on external knowledge and (4) improving the computational efficiency. Ultimately, an ideal dialogue system, equipped with useful knowledge bases, should be able to perform different functions seamlessly in a multi-modal process, and within a practical and affordable compute budget.

6.2 Future Work from the Thesis

This section discusses the future work that may naturally follow this thesis. Note that this section's standpoint is from late 2021, that is, when the last work (Chapter 2) covered in this thesis was done, as opposed to Section 6.4, which is from the standpoint of Mar. 2023.

We first discuss the generality of the task of dialogue systems. Compared to other NLP tasks that focus on a specific aspect of language-related human intelligence, dialogue modeling can be seen as an all-in-one master task as it can naturally cover other tasks because of its very broad definition. Almost every NLP task can be converted to a dialogue modeling problem. (This is later confirmed by ChatGPT as it can be seen as a universal text-based task solver.)

For example, question answering can be directly seen as a dialogue if one regards the question as the user message and the answer as the reply. Knowledge-grounded or multi-modal question answering can theoretically be covered by knowledge-grounded or multi-modal dialogue modeling. Even tasks that don't seem to be related to dialogues at all can be converted to dialogue modeling. For example, machine translation projects a sentence from its source language form to its target language form. By prepending "please translate this sentence to English for me:" to the source language sentence, the sentence pair becomes "message + response" like in dialogue systems. Dialogue modeling as a language generation task implicitly covers certain modular tasks. As an example, solving TODs implicitly requires solving named entity recognition and co-reference resolution.

Given such nature of dialogue systems - it may take on a very broad definition and cover almost all aspects of language-related human intelligence - it is no surprise that dialogue systems are a very challenging task. Yet every step further implies a considerable progress in AI.

Looking into the future from the standpoint of this thesis, we expect dialogue systems to continue to improve, very likely following general innovation in AI methodologies. For example, the following directions may be considered important for the long-term success of AI and thus also dialogue systems.

(1) Fast and easy generalization based on few-shot learning, zero-shot learning or meta learning. Compared to human intelligence, neural networks can be unbearably data-hungry. Thousands of carefully curated data points can be needed to learn to perform a new task. In contrast, humans require much less data (experience) because we can effectively utilize prior knowledge. Fortunately, recent research has began to shine light on this problem. For example, GPT3 (Brown et al., 2020) demonstrated that a sufficiently large language model possesses remarkable few-shot or even zero-shot learning skills on a variety of tasks. Reed et al. (2022) took such realization and generalized it beyond text. Using a unified tokenization-based serialization scheme, their model "Gato" is pretrained on hundreds of tasks with different modalities and embodiments, such as image captioning, dialogues, and robotic control. Gato then displays few-shot learning capabilities in brand new tasks of similar categories.

(2) Several flaws of the state-of-the-art paradigms are not to be ignored regarding interpretablity and robustness. It has been noted that deep neural networks can be vulnerable to adversarial attacks. For example, minor phrase modification can easily deceive Google's toxic comment detection systems (Hosseini et al., 2017). In general, simple techniques such as word or pixel substitution that can't confuse humans can be used to fool neural networks (Dong et al., 2020). Safety and controllability are also concerns especially for open-ended generation models. For example, Microsoft's dialogue system Tay learned to produce toxic responses in an online learning scenario (Wolf et al., 2017). In general, generation-based models may fail in unpredictable ways, thus restricting their usability to human-supervised scenarios (Pearce et al., 2021). These problems are related to the black-box nature of neural networks, for which research on interpretability may be important.

6.3 ChatGPT and GPT4

At the end of 2022, dialogue systems phenomenally took off as ChatGPT demonstrated a whole new level of usability.

For the initial version, ChatGPT was based on GPT3 (Brown et al., 2020), a large transformer trained on a large corpus of text in a self-supervised manner, and then further optimized for dialogue by using Reinforcement Learning with Human

Feedback (RLHF, Ouyang et al., 2022), a method that uses human demonstrations and preference comparisons to guide the model toward desired behavior.

RLHF is a technique that trains a "reward model" directly from human feedback and uses the model as a reward function to optimize an agent's policy using reinforcement learning through an optimization algorithm like Proximal Policy Optimization (Schulman et al., 2017). RLHF has enabled language models to align to complex human values without losing raw capabilities learned from massive unstructured text.

The impact of ChatGPT on dialogue systems and AI in general has been significant. By presenting a general easy-to-use text-to-text framework, ChatGPT demonstrated a very successful consumer-facing AI product, and it has been holding people's attention across industries since.

GPT4 (OpenAI, 2023), introduced in March 2023, represents further progress on top of the initial versions of ChatGPT. It demonstrates stronger reasoning capabilities, especially on complicated questions. In addition, it demonstrates multimodal capabilities, i.e., the user may intersperse text and images in the context. As mentioned in Section 6.2, dialogue systems may take on almost any type of text-based tasks. GPT4, when applied to academic and professional tests, is on par with humans in many of them.

6.4 Future Work from GPT4

While GPT4 demonstrated state-of-the-art text understanding and generation capabilities, there are still multiple fronts that researchers can work on to improve it. In this section, we discuss 3 of such directions, including handling longer context, enabling embodied intelligence and reducing hallucinations.

6.4.1 Longer Context

Large language models are usually trained with a fixed maximum context length. And they don't perform well when tested beyond it (Anil et al., 2022).

The motivation for longer context handling is to improve the performance of natural language processing models on tasks that require processing long contexts beyond the traditional length limit of pretrained models, i.e., ranging from a few thousands (Brown et al., 2020) to tens of thousands (OpenAI, 2023).

Two directions can be taken to enable the models to perform well on longer problems.

First, novel attention mechanisms can be created to enable training the models on longer context efficiently, for example, by using sparse attention mechanism (Martins et al., 2020).

Second, prompting techniques such as scratchpad prompting can be utilized (Anil et al., 2022), i.e., by asking the model to output solution steps before producing an answer.

6.4.2 Embodied Intelligence

Embodied intelligence (Francis et al., 2022) is a concept that emphasizes the importance of an agent's physical interaction with its environment for cognitive development and intelligent behavior. The idea behind this approach is that the body and the environment play a crucial role in shaping an organism's or robot's cognitive abilities, rather than just the information processing in the brain.

In embodied intelligence, the body's morphology, sensors, and actuators are considered integral to learning and problem-solving. This perspective contrasts with traditional AI approaches that focus primarily on abstract problem-solving and symbolic reasoning, often disconnected from the real world.

Large language models have the appearance of understanding but do not have experiences. Linking large language models, like GPT4, with embodied intelligence involves integrating the language model into an embodied agent (such as a robot or a virtual agent) that can interact with its environment. The idea is to combine the powerful natural language understanding and generation capabilities of the language model with the sensorimotor and adaptive capabilities of the embodied agent. This can enable more sophisticated, adaptive, and context-aware behavior in real-world environments.

A recent example of such effort is PaLM-E (Driess et al., 2023). The paper embodied PaLM by adding sensory information and robotic control. PaLM-E works by injecting observations into a pre-trained language model. The main architectural idea of PaLM-E is to inject continuous, embodied observations such as images, state estimates, or other sensor modalities into the language embedding space of a pre-trained language model. Therefore, it is shown that PaLM-E, a single large embodied multimodal model, can address a variety of embodied reasoning tasks, from a variety of observation modalities, on multiple robotic embodiments.

6.4.3 Reducing Hallucinations

Hallucinations in large language models refer to mistakes in the generated text that are semantically or syntactically plausible but are in fact incorrect or nonsensical. Hallucinations can have several technical repercussions. For starters, it makes the generated content less trustworthy, rendering it unsuitable for applications that demand factual accuracy, for example, law and healthcare. In addition, hallucinations can lead to biased or unfair decisions when used in decision-making systems (Rowdur, 2023).

Intuitively speaking, hallucination is an artifact of the compression property of large language models, i.e., it occurs when the model tries to recreate information it has not explicitly memorized or seen (Dibia, 2023).

There are practical steps that can be taken to reduce hallucination. One way to reduce hallucination is to use adversarial training. This involves training the model on adversarial examples that are designed to expose the model's weaknesses and improve its robustness (Dibia, 2023). Another way to reduce hallucination is to use task-specific fine-tuning. This involves fine-tuning it on a specific task or domain to improve its performance on that task or domain (Ziegler et al., 2019).

Awards

- 2021 IEEE CIM Outstanding Paper Award

Journal Articles

(Tom Young is the publication name for Yang Tianji.)

- **Tom Young**, Vlad Pandelea, Soujanya Poria, and Erik Cambria. "Dialogue systems with audio context." *Neurocomputing* 388 (2020): 102-109.
- **Tom Young**, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. "Recent trends in deep learning based natural language processing." *IEEE Computational Intelligence Magazine* 13, no. 3 (2018): 55-75.
- Vlad Pandelea, Edoardo Ragusa, **Tom Young**, Paolo Gastaldo, and Erik Cambria. "Toward hardware-aware deep-learning-based dialogue systems." *Neural Computing and Applications* (2021): 1-12.
- Ni Jinjie, **Tom Young**, Vlad Pandelea, Fuzhao Xue, Vinay Adiga, and Erik Cambria. "Recent advances in deep learning based dialogue systems: A systematic survey." under submission to Artificial Intelligence Review.

Conference Proceedings

- **Tom Young**, Erik Cambria, Iti Chaturvedi, Hao Zhou, Subham Biswas, and Minlie Huang. "Augmenting end-to-end dialogue systems with commonsense knowledge." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1. 2018.

- **Tom Young**, Frank Xing, Vlad Pandelea, Jinjie Ni, and Erik Cambria. "Fusing task-oriented and open-domain dialogues in conversational agents." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 10, pp. 11622-11629. 2022.

- Ni Jinjie, Vlad Pandelea, **Tom Young**, Haicang Zhou, and Erik Cambria. "HiTKG: Towards Goal-Oriented Conversations via Multi-Hierarchy Learning." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 10, pp. 11112-11120. 2022.

Bibliography

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. xxi, 6, 7

[2] Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. *arXiv preprint arXiv:1506.08909*, 2015. xxiv, 37, 40, 56, 63, 70

[3] Ryan Lowe, Iulian Vlad Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. On the evaluation of dialogue systems with next utterance classification. *CoRR*, abs/1605.05414, 2016. URL `http://arxiv.org/abs/1605.05414`. xxiv, 63

[4] *https://www.marketdataforecast.com/market-reports/chatbot-market*. 1

[5] Jinjie Ni, Tom Young, Vlad Pandelea, Fuzhao Xue, Vinay Adiga, and Erik Cambria. Recent advances in deep learning based dialogue systems: A systematic survey. *arXiv preprint arXiv:2105.04387*, 2021. 1, 29, 30

[6] *https://www.statista.com/statistics/1029573/us-voice-assistant-users/*. 1

[7] *https://www.smallbizgenius.net/by-the-numbers/chatbot-statistics*. 1

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

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

[10] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014. 2, 9, 38

[11] Oriol Vinyals and Quoc Le. A neural conversational model. *arXiv preprint arXiv:1506.05869*, 2015. 2, 9

[12] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764, 2019. 2

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2, 7, 25, 49, 70, 79

[14] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 2

[15] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020. 2, 49, 91, 93

[16] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 5

[17] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 7, 25

[18] Alan Ritter, Colin Cherry, and William B. Dolan. Data-driven response generation in social media. In *EMNLP*, pages 583–593, 2011. 9

[19] Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C. Courville, and Joelle Pineau. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*, pages 3776–3784, 2016. 9, 49

[20] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*, 2015. 10

[21] Antoine Bordes and Jason Weston. Learning end-to-end goal-oriented dialog. *CoRR*, abs/1605.07683, 2016. 10, 11, 54, 60, 61, 65

[22] Wenya Zhu, Kaixiang Mo, Yu Zhang, Zhangbin Zhu, Xuezheng Peng, and Qiang Yang. Flexible end-to-end dialogue system for knowledge grounded conversation. *CoRR*, abs/1709.04264, 2017. 10

[23] Tiancheng Zhao, Allen Lu, Kyusong Lee, and Maxine Eskenazi. Generative encoder-decoder models for task-oriented spoken dialog systems with chatting capability. *arXiv preprint arXiv:1706.08476*, 2017. 10, 31

[24] Nasrin Mostafazadeh, Chris Brockett, Bill Dolan, Michel Galley, Jianfeng Gao, Georgios P Spithourakis, and Lucy Vanderwende. Image-grounded conversations: Multimodal context for natural question and response generation. *arXiv preprint arXiv:1701.08251*, 2017. 10, 48

[25] Zhou Yu. Attention and engagement aware multimodal conversational systems. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 593–597. ACM, 2015. 10, 49

[26] Minghai Chen, Sen Wang, Paul Pu Liang, Tadas Baltrušaitis, Amir Zadeh, and Louis-Philippe Morency. Multimodal sentiment analysis with word-level fusion and reinforcement learning. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pages 163–171. ACM, 2017. 10, 39, 49

[27] Jason Weston, Antoine Bordes, Sumit Chopra, Alexander M Rush, Bart van Merriënboer, Armand Joulin, and Tomas Mikolov. Towards ai-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*, 2015. 11, 65

[28] Jesse Dodge, Andreea Gane, Xiang Zhang, Antoine Bordes, Sumit Chopra, Alexander Miller, Arthur Szlam, and Jason Weston. Evaluating prerequisite qualities for learning end-to-end dialog systems. *arXiv preprint arXiv:1511.06931*, 2015. 11, 60, 61, 65

[29] Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen-tau Yih, and Michel Galley. A knowledge-grounded neural conversation model. *CoRR*, abs/1702.01932, 2017. URL `http://arxiv.org/abs/1702.01932`. 11, 54, 65

[30] Ryan Lowe, Nissan Pow, Laurent Charlin, Joelle Pineau, and Iulian V. Serban. Incorporating unstructured textual knowledge sources into neural dialogue systems. In *Machine Learning for Spoken Language Understanding and Interaction, NIPS 2015 Workshop*, 2015. URL `http://slunips2015.wixsite.com/slunips2015/accepted-papers`. 11, 65

[31] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 11, 84

[32] Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. *arXiv preprint arXiv:1606.07947*, 2016. 11, 84

[33] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. In *Advances in Neural Information Processing Systems*, pages 742–751, 2017. 12, 84

[34] OpenAI. Chatgpt, 2022. URL `https://beta.openai.com/docs/api-reference/generations/v1/#chat`. 14, 32, 85, 89

[35] OpenAI. Gpt-4, 2023. URL `https://arxiv.org/abs/2303.12712`. 14, 33, 51, 52, 55, 66, 85, 89, 92

[36] Tom Young, Frank Xing, Vlad Pandelea, Jinjie Ni, and Erik Cambria. Fusing task-oriented and open-domain dialogues in conversational agents. In *Thirty-Sixth AAAI Conference on Artificial Intelligence*, 2022. 16

[37] Daniel Adiwardana, Minh-Thang Luong, David R So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, et al. Towards a human-like open-domain chatbot. *arXiv preprint arXiv:2001.09977*, 2020. 16, 29, 66

[38] Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M Smith, et al. Recipes for building an open-domain chatbot. *arXiv preprint arXiv:2004.13637*, 2020. 16, 66

[39] Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. Dialogpt: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536*, 2019. 16, 24, 66

[40] Donghoon Ham, Jeong-Gwan Lee, Youngsoo Jang, and Kee-Eung Kim. End-to-end neural pipeline for goal-oriented dialogue systems using gpt-2. ACL, 2020. 16, 25, 27, 66

[41] Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Inigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. Multiwoz–a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *arXiv preprint arXiv:1810.00278*, 2018. 16, 19

[42] Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8689–8696, 2020. 16

[43] Bill Byrne, Karthik Krishnamoorthi, Chinnadhurai Sankar, Arvind Neelakantan, Daniel Duckworth, Semih Yavuz, Ben Goodrich, Amit Dubey, Andy Cedilnik, and Kyu-Young Kim. Taskmaster-1: Toward a realistic and diverse dialog dataset. *arXiv preprint arXiv:1909.05358*, 2019. 18

[44] Fanghua Ye, Jarana Manotumruksa, and Emine Yilmaz. Multiwoz 2.4: A multi-domain task-oriented dialogue dataset with essential annotation corrections to improve state tracking evaluation. *arXiv preprint arXiv:2104.00773*, 2021. 18

[45] Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. Personalizing dialogue agents: I have a dog, do you have pets too? *arXiv preprint arXiv:1801.07243*, 2018. 20, 32, 79

[46] Eric Michael Smith, Mary Williamson, Kurt Shuster, Jason Weston, and Y-Lan Boureau. Can you put it all together: Evaluating conversational agents' ability to blend skills. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2021–2030, 2020. 20, 31

[47] Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. Wizard of wikipedia: Knowledge-powered conversational agents. *arXiv preprint arXiv:1811.01241*, 2018. 20

[48] Jun Quan, Shian Zhang, Qian Cao, Zizhong Li, and Deyi Xiong. Risawoz: A large-scale multi-domain wizard-of-oz dataset with rich semantic annotations for task-oriented dialogue modeling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 930–940, 2020. 22

[49] Kurt Shuster, Da Ju, Stephen Roller, Emily Dinan, Y-Lan Boureau, and Jason Weston. The dialogue dodecathlon: Open-domain knowledge and image grounded conversational agents. *arXiv preprint arXiv:1911.03768*, 2019. 24, 31

[50] Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. A simple language model for task-oriented dialogue. *arXiv preprint arXiv:2005.00796*, 2020. 25

[51] Andrea Madotto, Zhaojiang Lin, Yejin Bang, and Pascale Fung. The adapter-bot: All-in-one controllable conversational model. *arXiv preprint arXiv:2008.12579*, 2020. 25, 30

[52] Kai Sun, Seungwhan Moon, Paul Crook, Stephen Roller, Becka Silvert, Bing Liu, Zhiguang Wang, Honglei Liu, Eunjoon Cho, and Claire Cardie. Adding chit-chats to enhance task-oriented dialogues. *arXiv preprint arXiv:2010.12757*, 2020. 25, 31

[53] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR, 2019. 30

[54] Andrea Madotto, Zhaojiang Lin, Chien-Sheng Wu, Jamin Shin, and Pascale Fung. Attention over parameters for dialogue systems. *arXiv preprint arXiv:2001.01871*, 2020. 30

[55] Zhiyu Chen, Bing Liu, Seungwhan Moon, Chinnadhurai Sankar, Paul Crook, and William Yang Wang. Ketod: Knowledge-enriched task-oriented dialogue. *arXiv preprint arXiv:2205.05589*, 2022. 31

[56] Miaoran Li, Baolin Peng, Jianfeng Gao, and Zhu Zhang. Opera: Harmonizing task-oriented dialogs and information seeking experience. *arXiv preprint arXiv:2206.12449*, 2022. 31

[57] Vojtěch Hudeček and Ondřej Dušek. Are llms all you need for task-oriented dialogue? *arXiv preprint arXiv:2304.06556*, 2023. 32

[58] Seokhwan Kim, Mihail Eric, Karthik Gopalakrishnan, Behnam Hedayatnia, Yang Liu, and Dilek Hakkani-Tur. Beyond domain apis: Task-oriented conversational modeling with unstructured knowledge access. *arXiv preprint arXiv:2006.03533*, 2020. 32

[59] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 33

[60] Tom Young, Vlad Pandelea, Soujanya Poria, and Erik Cambria. Dialogue systems with audio context. *Neurocomputing*, 388:102–109, 2020. 36

[61] Björn Schuller, Stefan Steidl, Anton Batliner, Alessandro Vinciarelli, Klaus Scherer, Fabien Ringeval, Mohamed Chetouani, Felix Weninger, Florian Eyben, Erik Marchi, et al. The interspeech 2013 computational paralinguistics challenge: social signals, conflict, emotion, autism. In *Proceedings INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France*, 2013. 37, 41

[62] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*, 2017. 37

[63] Alan Ritter, Colin Cherry, and Bill Dolan. Unsupervised modeling of twitter conversations. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 172–180. Association for Computational Linguistics, 2010. 40

[64] Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N Chang, Sungbok Lee, and Shrikanth S Narayanan. Iemocap: Interactive emotional dyadic motion capture database. *Language resources and evaluation*, 42(4):335, 2008. 40, 49

[65] Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. Meld: A multimodal multi-party dataset for emotion recognition in conversations. *arXiv preprint arXiv:1810.02508*, 2018. 40, 41

[66] Dom Lachowicz. pyenchant. `https://github.com/rfk/pyenchant`. 41

[67] Robert Ochshorn and Max Hawkins. Gentle: A forced aligner, 2016. 41

[68] Florian Eyben, Martin Wöllmer, and Björn Schuller. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1459–1462. ACM, 2010. 41

[69] Haishu Xianyu, Xinxing Li, Wenxiao Chen, Fanhang Meng, Jiashen Tian, Mingxing Xu, and Lianhong Cai. Svr based double-scale regression for dynamic emotion prediction in music. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 549–553. IEEE, 2016. 41

[70] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015. 42

[71] pushshift.io. Reddit comments data. `https://www.reddit.com/r/datasets/comments/3bxlg7/i_have_every_publicly_available_reddit_comment/`. 42, 78

[72] Minh-Thang Luong, Eugene Brevdo, and Rui Zhao. Neural machine translation (seq2seq) tutorial. *https://github.com/tensorflow/nmt*, 2017. 42

[73] Shuman Liu, Hongshen Chen, Zhaochun Ren, Yang Feng, Qun Liu, and Dawei Yin. Knowledge diffusion for neural dialogue generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1489–1498, 2018. 43

[74] Marc Schröder and Jürgen Trouvain. The German Text-to-Speech Synthesis System MARY: A Tool for Research, Development and Teaching. In *4th ISCA Workshop on Speech Synthesis*, 2001. URL `../../pdf/schroeder_trouvain2001.pdf`. 45

[75] James A Russell. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161, 1980. 45

[76] Huda Alamri, Vincent Cartillier, Raphael Gontijo Lopes, Abhishek Das, Jue Wang, Irfan Essa, Dhruv Batra, Devi Parikh, Anoop Cherian, Tim K Marks, et al. Audio visual scene-aware dialog (avsd) challenge at dstc7. *arXiv preprint arXiv:1806.00525*, 2018. 48

[77] Chiori Hori, Huda Alamri, Jue Wang, Gordon Winchern, Takaaki Hori, Anoop Cherian, Tim K Marks, Vincent Cartillier, Raphael Gontijo Lopes, Abhishek Das, et al. End-to-end audio visual scene-aware dialog using multimodal attention-based video features. *arXiv preprint arXiv:1806.08409*, 2018. 48

[78] Amrita Saha, Mitesh M Khapra, and Karthik Sankaranarayanan. Towards building large scale multimodal domain-aware conversation systems. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 49

[79] Shubham Agarwal, Ondrej Dusek, Ioannis Konstas, and Verena Rieser. Improving context modelling in multimodal dialogue generation. *arXiv preprint arXiv:1810.11955*, 2018. 49

[80] Yue Gu, Xinyu Li, Kaixiang Huang, Shiyu Fu, Kangning Yang, Shuhong Chen, Moliang Zhou, and Ivan Marsic. Human conversation analysis using attentive multimodal networks with hierarchical encoder-decoder. In *2018 ACM Multimedia Conference on Multimedia Conference*, pages 537–545. ACM, 2018. 49

[81] Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 49

[82] Jie Lu, Ryan Kiros, and Richard S. Zemel. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*, 2019. 50

[83] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, pages 5583–5594. PMLR, 2021. 50

[84] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *Advances in Neural Information Processing Systems*, 34:24206–24221, 2021. 50

[85] Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. data2vec: A general framework for self-supervised learning in speech, vision and language. 2022. 50

[86] Hisashi Kamezawa, Noriki Nishida, Nobuyuki Shimizu, Takashi Miyazaki, and Hideki Nakayama. A visually-grounded first-person dialogue dataset with verbal and non-verbal responses. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3299–3310, 2020. 50

[87] Seungwhan Moon, Satwik Kottur, Paul A Crook, Ankita De, Shivani Poddar, Theodore Levin, David Whitney, Daniel Difranco, Ahmad Beirami, Eunjoon Cho, et al. Situated and interactive multimodal conversations. *arXiv preprint arXiv:2006.01460*, 2020. 50

[88] Hung Le and Steven CH Hoi. Video-grounded dialogues with pretrained generation language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5842–5848, 2020. 50, 51

[89] Zekang Li, Zongjia Li, Jinchao Zhang, Yang Feng, and Jie Zhou. Bridging text and video: A universal multimodal transformer for audio-visual scene-aware dialog. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:2476–2483, 2021. 51

[90] Shiquan Yang, Rui Zhang, Sarah M Erfani, and Jey Han Lau. Unimf: A unified framework to incorporate multimodal knowledge bases into end-to-end task-oriented dialogue systems. In *IJCAI*, pages 3978–3984, 2021. 51, 67

[91] Satwik Kottur, Seungwhan Moon, Alborz Geramifard, and Babak Damavandi. Simmc 2.0: a task-oriented dialog dataset for immersive multimodal conversations. *arXiv preprint arXiv:2104.08667*, 2021. 51

[92] Robert Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In *AAAI*, pages 4444–4451, 2017. 53

[93] Erik Cambria, Qian Liu, Sergio Decherchi, Frank Xing, and Kenneth Kwok. Senticnet 7: a commonsense-based neurosymbolic ai framework for explainable sentiment analysis. 2022. 53

[94] Tom Young, Erik Cambria, Iti Chaturvedi, Hao Zhou, Subham Biswas, and Minlie Huang. Augmenting end-to-end dialogue systems with commonsense knowledge. In *AAAI*, pages 4970–4977, 2018. 54

[95] Marvin Minsky. *The Society of Mind*. Simon and Schuster, New York, 1986. 55

[96] Erik Cambria, Amir Hussain, Catherine Havasi, and Chris Eckl. Common sense computing: From the society of mind to digital intuition and beyond. In Julian Fierrez, Javier Ortega, Anna Esposito, Andrzej Drygajlo, and Marcos Faundez-Zanuy, editors, *Biometric ID Management and Multimodal Communication*, volume 5707 of *Lecture Notes in Computer Science*, pages 252–259. Springer, Berlin Heidelberg, 2009. 55

[97] Ha-Nguyen Tran, Erik Cambria, and Amir Hussain. Towards gpu-based common-sense reasoning: Using fast subgraph matching. *Cognitive Computation*, 8(6):1074–1086, 2016. 55

[98] Robert Speer and Catherine Havasi. Representing general relational knowledge in conceptnet 5. In *LREC*, pages 3679–3686, 2012. 55

[99] Erik Cambria, Daniel Olsher, and Dheeraj Rajagopal. Senticnet 3: a common and common-sense knowledge base for cognition-driven sentiment analysis. In *Twenty-eighth AAAI conference on artificial intelligence*, 2014. 55

[100] Push Singh, Thomas Lin, Erik T Mueller, Grace Lim, Travell Perkins, and Wan Li Zhu. Open mind common sense: Knowledge acquisition from the general public. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 1223–1237. Springer, 2002. 55

[101] Chia-Wei Liu, Ryan Lowe, Iulian V Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023*, 2016. 55

[102] Ryan Lowe, Iulian V Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. On the evaluation of dialogue systems with next utterance classification. *arXiv preprint arXiv:1605.05414*, 2016. 55

[103] Dheeraj Rajagopal, Erik Cambria, Daniel Olsher, and Kenneth Kwok. A graph-based approach to commonsense concept extraction and semantic similarity detection. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 565–570. ACM, 2013. 57

[104] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. 60

[105] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448, 2015. 60, 65

[106] Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. *CoRR*, abs/1410.3916, 2014. 60

[107] Ryan Lowe, Michael Noseworthy, Iulian V Serban, Nicolas Angelard-Gontier, Yoshua Bengio, and Joelle Pineau. Towards an automatic turing test: Learning to evaluate dialogue responses. *arXiv preprint arXiv:1708.07149*, 2017. 62

[108] Jason Weston, Sumit Chopra, Alexander M. Rush, Antoine Bordes, and Tomas Mikolov. Towards AI-Complete question answering: A set of prerequisite toy tasks. *CoRR*, abs/1502.05698, 2015. URL `http://arxiv.org/abs/1502.05698`. 62

[109] Qianzhou Zhou, Nan Hu, Weiwei Zhao, and Chengqing Zong. Emotional chatting machine: Emotional conversation generation with internal and external memory. *CoRR*, abs/1611.08986, 2016. URL `http://arxiv.org/abs/1611.08986`. 62

[110] Robert Speer and Catherine Havasi. ConceptNet 5: A large semantic network for relational knowledge. In Eduard Hovy, M Johnson, and G Hirst, editors, *Theory and Applications of Natural Language Processing*, chapter 6. Springer, 2012. 62

[111] Juan Ramos et al. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, 2003. 63

[112] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–43, 2014. 63

[113] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. URL `http://arxiv.org/abs/1412.6980`. 63

[114] Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. Topic augmented neural response generation with a joint attention mechanism. *CoRR*, abs/1606.08340, 2016. URL `http://arxiv.org/abs/1606.08340`. 65

[115] Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. Commonsense knowledge aware conversation generation with graph attention. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 4623–4629, 2018. 65

[116] Xuhui Zhou, Yue Zhang, Leyang Cui, and Dandan Huang. Evaluating commonsense in pre-trained language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9733–9740, 2020. 66

[117] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019. 66

[118] Mojtaba Komeili, Kurt Shuster, and Jason Weston. Internet-augmented dialogue generation. *arXiv preprint arXiv:2107.07566*, 2021. 66

[119] Kurt Shuster, Mojtaba Komeili, Leonard Adolphs, Stephen Roller, Arthur Szlam, and Jason Weston. Language models that seek for knowledge: Modular search & generation for dialogue and prompt completion. *arXiv preprint arXiv:2203.13224*, 2022. 67

[120] Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*, 2022. 67

[121] Google. Bard: A large language model from google ai, 2023. URL `https://bard.google.com/`. 67

[122] Microsoft. Bing chat: A new ai-powered chat experience from microsoft, 2023. URL `https://www.bing.com/new`. 67

[123] Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Sebastian Ruder, Dani Yogatama, et al. Pitfalls of static language modelling. *arXiv preprint arXiv:2102.01951*, 2021. 67

[124] Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. Qa-gnn: Reasoning with language models and knowledge graphs for question answering. *arXiv preprint arXiv:2104.06378*, 2021. 68

[125] Basma El Amel Boussaha, Nicolas Hernandez, Christine Jacquin, and Emmanuel Morin. Deep retrieval-based dialogue systems: A short review. *arXiv preprint arXiv:1907.12878*, 2019. 70

[126] Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring, 2019. 70, 80

[127] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016. 78

[128] Patrick H Chen, Si Si, Sanjiv Kumar, Yang Li, and Cho-Jui Hsieh. Learning to screen for fast softmax inference on large vocabulary neural networks. *arXiv preprint arXiv:1810.12406*, 2018. 78, 85

[129] Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. Dailydialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 986–995, 2017. 78

[130] Emily Dinan, Varvara Logacheva, Valentin Malykh, Alexander Miller, Kurt Shuster, Jack Urbanek, Douwe Kiela, Arthur Szlam, Iulian Serban, Ryan Lowe, et al. The second conversational intelligence challenge (convai2). *arXiv preprint arXiv:1902.00098*, 2019. 79

[131] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771, 2019. 79

[132] Raphael Tang, Yao Lu, Linqing Liu, Lili Mou, Olga Vechtomova, and Jimmy Lin. Distilling task-specific knowledge from bert into simple neural networks. *arXiv preprint arXiv:1903.12136*, 2019. 80, 84

[133] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 80

[134] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang. Structured knowledge distillation for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2604–2613, 2019. 84

[135] Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling bert for natural language understanding. *arXiv preprint arXiv:1909.10351*, 2019. 84

[136] Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. Patient knowledge distillation for bert model compression. *arXiv preprint arXiv:1908.09355*, 2019. 84

[137] Minjia Zhang, Wenhan Wang, Xiaodong Liu, Jianfeng Gao, and Yuxiong He. Navigating with graph representations for fast and scalable decoding of neural language models. In *Advances in Neural Information Processing Systems*, pages 6308–6319, 2018. 85

[138] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022. 85, 92

[139] Guangxuan Xiao, Ji Lin, Mickael Seznec, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. *arXiv preprint arXiv:2211.10438*, 2022. 86

[140] Yuvanesh Anand, Zach Nussbaum, Brandon Duderstadt, Benjamin Schmidt, and Andriy Mulyar. Gpt4all: Training an assistant-style chatbot with large scale data distillation from gpt-3.5-turbo. 86

[141] Nan Yang, Tao Ge, Liang Wang, Binxing Jiao, Daxin Jiang, Linjun Yang, Rangan Majumder, and Furu Wei. Inference with reference: Lossless acceleration of large language models. *arXiv preprint arXiv:2304.04487*, 2023. 86

[142] Zhoujun Cheng, Jungo Kasai, and Tao Yu. Batch prompting: Efficient inference with large language model apis. *arXiv preprint arXiv:2301.08721*, 2023. 86

[143] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity, 2021. 87

[144] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, et al. A generalist agent. *arXiv preprint arXiv:2205.06175*, 2022. 91

[145] Hossein Hosseini, Sreeram Kannan, Baosen Zhang, and Radha Poovendran. Deceiving google's perspective api built for detecting toxic comments. *arXiv preprint arXiv:1702.08138*, 2017. 91

[146] Xinshuai Dong, Anh Tuan Luu, Rongrong Ji, and Hong Liu. Towards robustness against natural language word substitutions. In *International Conference on Learning Representations*, 2020. 91

[147] Marty J Wolf, Keith W Miller, and Frances S Grodzinsky. Why we should have seen that coming: comments on microsoft's tay "experiment," and wider implications. *The ORBIT Journal*, 1(2):1–12, 2017. 91

[148] Hammond Pearce, Baleegh Ahmad, Benjamin Tan, Brendan Dolan-Gavitt, and Ramesh Karri. An empirical cybersecurity evaluation of github copilot's code contributions. *arXiv e-prints*, pages arXiv–2108, 2021. 91

[149] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 92

[150] Cem Anil, Yuhuai Wu, Anders Andreassen, Aitor Lewkowycz, Vedant Misra, Vinay Ramasesh, Ambrose Slone, Guy Gur-Ari, Ethan Dyer, and Behnam Neyshabur. Exploring length generalization in large language models. In *36th Conference on Neural Information Processing Systems (NeurIPS 2022)*. Google Research, Blueshift Team; Google Research; University of Toronto, Vector Institute, 2022. 92, 93

[151] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. URL `https://arxiv.org/abs/2303.08774`. 93

[152] André F. T. Martins, António Farinhas, Marcos Treviso, Vlad Niculae, Pedro M. Q. Aguiar, and Mário A. T. Figueiredo. Sparse and continuous attention mechanisms. *arXiv preprint arXiv:2006.07214*, 2020. URL `https://arxiv.org/abs/2006.07214`. 93

[153] Jonathan Francis, Nariaki Kitamura, Felix Labelle, Xiaopeng Lu, Ingrid Navarro, and Jean Oh. Core challenges in embodied vision-language planning. *Journal of Artificial Intelligence Research*, 74:459–515, 2022. 93

[154] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023. 94

[155] Srinivas Rowdur. Llm hallucinations. `https://www.linkedin.com/pulse/llm-gotchas-1-hallucinations-srinivas-rowdur/`, 2023. Accessed: April 10, 2023. 94

[156] Victor Dibia. Practical steps to reduce hallucination and improve performance of systems built with large language models. `https://newsletter.victordibia.com/p/practical-steps-to-reduce-hallucination`, 2023. Accessed: April 12, 2023. 94

[157] Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019. 94