Towards Humanized Open-Domain Conversational Agents

Peixiang Zhong

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

2021

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.

3 Dec. 2020	
.....	.....
Date	Peixiang Zhong

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.

3 Dec. 2020

.....

Date

.....

Prof. Chunyan Miao

Authorship Attribution Statement

This thesis contains material from 4 papers published in the following peer-reviewed conferences in which I am listed as an author.

Chapter 3 is published as Peixiang, Zhong, Di Wang, and Chunyan Miao. "An Affect-Rich Neural Conversational Model with Biased Attention and Weighted Cross-Entropy Loss." Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2019), Pages 7492-7500.

The contributions of the co-authors are as follows:

- Prof. Chunyan Miao and I discussed the initial research direction.
- I came up with the key idea and methods, designed and conducted all experiments. I prepared the manuscript drafts.
- Dr. Di Wang provided feedback about the initial idea.
- The manuscripts were revised mainly by Dr. Di Wang and Prof. Chunyan Miao.

Chapter 4 is published as Peixiang, Zhong, Di Wang, and Chunyan Miao. "Knowledge-Enriched Transformer for Emotion Detection in Textual Conversations." Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019), Pages 165-176.

The contributions of the co-authors are as follows:

- Prof. Chunyan Miao and I discussed the initial research direction.
- I came up with the key idea and methods, designed and conducted all experiments. I prepared the manuscript drafts.
- The manuscripts were revised mainly by Dr. Di Wang and Prof. Chunyan Miao.

Chapter 5 is published as Peixiang, Zhong, Di Wang, Pengfei Li, Chen Zhang, Hao Wang, and Chunyan Miao. "CARE: Commonsense-Aware Emotional Response Generation with Latent Concepts." Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2021).

The contributions of the co-authors are as follows:

- Prof. Chunyan Miao and I discussed the initial research direction.
- I came up with the key idea and methods, designed and conducted all experiments. I prepared the manuscript drafts.
- Dr. Di Wang and Dr. Pengfei Li provided feedback to the initial idea.

- The manuscripts were revised mainly by Dr. Di Wang, Dr. Chen Zhang, Dr. Hao Wang, and Prof. Chunyan Miao.

Chapter 6 is published as Peixiang, Zhong, Chen Zhang, Hao Wang, Yong Liu, and Chunyan Miao. "Towards Persona-Based Empathetic Conversational Models." Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020), Pages 6556–6566.

The contributions of the co-authors are as follows:

- Prof. Chunyan Miao and I discussed the initial research direction.
- I came up with the key idea and methods, designed and conducted all experiments. I prepared the manuscript drafts.
- Dr. Hao Wang and Dr. Yong Liu provided feedback to the initial idea.
- The manuscripts were revised mainly by Dr. Chen Zhang, Dr. Hao Wang, Dr. Yong Liu, and Prof. Chunyan Miao.

3 Dec. 2020 ..... Date

..... Peixiang Zhong

Acknowledgements

First and foremost, I wish to express my sincere gratitude to my supervisor Prof. Miao Chunyan for the kind support and mentorship throughout my PhD. Her insight and support are what made this research possible. I would like to also thank my committee members Prof. Cyril Leung and Prof. Chng Eng Siong for their guidance.

I wish to thank my collaborators and colleagues who helped and supported me along my PhD journey. I am very glad to have the chance to collaborate with Wang Di, Liu Yong, and Wang Hao, who all contributed much to my development as a critical and independent researcher. I would like to also thank Shen Zhiqi, Yu Han, Nie Zaiqing, Zhou Tianyi, Li Boyang, Zhang Hao, Qian Hangwei, Zeng Zhiwei, Li Pengfei, Zhang Chen, Wu Qiong, Dong Yi, Wang Wei, Zhang Tong, Zhang Yinan, and Guo Xu for their support to my research. I am also grateful for my internship at Alibaba Damo Academy at Beijing, China.

My PhD would be less enjoyable without my friends: Ji Tete, Du Jiawei, Lu Shengliang, He Chaoyue, He Xuerui, Cheng Anqi, Chen Xingyu, Xu Mengxing, Fan Shengxin, Zheng Lulu, Qin Mengqi, Xing Zhe, Chen Yisi, and Lu Ze.

Finally, I would like to thank my parents for all the years of love and support. I would like to also thank my parents-in-law for their care and support for me. Most importantly, I express my deepest gratitude to my wife, Luo Ruijin, who accompanied and supported me with her love throughout my PhD.

"We can see only a short distance ahead, but we can see that much remains to be done."

—Alan, Turing

To my dear family

Abstract

Language is the hallmark of humanity. Conversation or dialogue is a fundamental arena of language and one of the most commonly used forms by humans.

In the field of artificial intelligence (AI) or, more specifically, natural language processing (NLP), a conversational agent (CA), also known as a dialogue system (DS), is an intelligent machine that can converse with humans in natural language. There are primarily three types of CAs: task-oriented CAs, question-answering (QA) systems, and open-domain CAs. In this thesis, we focus on open-domain CAs, also known as chatbots, which are designed to chat with users in any topics engagingly with the aim of establishing long-term relationships. Open-domain CAs are essential in modern conversational user interfaces and have been adopted in numerous business domains such as personal assistant, customer support, education, and healthcare. Building a human-level open-domain CA has been one of the major milestones in AI research.

However, existing open-domain CAs often fail to model the intrinsic traits of humans and exhibit the following limitations: 1) they lack emotional intelligence and cannot generate or recognize emotions in conversations, which often lead to dull or generic responses; 2) they lack commonsense knowledge and often produce incoherent or unrelated responses; 3) they lack persona and often produce inconsistent responses; and 4) they lack empathy and often produce non-empathetic responses.

Addressing the aforementioned limitations is important for bridging the gap between existing CAs and human-level CAs. These intrinsic traits of humans have been empirically shown to improve the performance of CAs on various tasks, e.g., user satisfaction in customer support, user trust and engagement in education, and mental health of participants in healthcare.

Humanization is the process of attributing human traits to an entity. In this thesis, we propose to address the limitations by humanizing open-domain CAs with the

following human traits: emotion, commonsense, persona, and empathy. Our thesis makes a step towards humanized open-domain CAs.

Specifically, to humanize CAs with emotion and commonsense, we first propose an emotional open-domain CA that can generate natural and emotional responses. We then incorporate commonsense into emotional CAs and propose a conversational emotion recognition model and a commonsense-aware emotional response generation model. Experimental results show that both emotion and commonsense improve response quality and human ratings. In addition, emotion and commonsense are shown to have complementary effects in conversational emotion recognition and generation.

To humanize CAs with persona and empathy, we propose a persona-based empathetic CA and investigate the impact of persona and empathy on response quality. Experimental results show that both persona and empathy consistently improve response quality and human ratings. In addition, we investigate the impact of persona on empathetic responding and our results suggest that persona has a larger impact on empathetic conversations than non-empathetic ones.

Finally, we propose a humanized open-domain CA (HCA) that possesses all the proposed human traits simultaneously: emotion, commonsense, persona, and empathy. HCA aims to address the aforementioned limitations altogether. Specifically, we adopt a pretrain-and-finetune paradigm to develop a retrieval-based HCA in a multi-task learning setting. Experimental results show that the multi-task performance of HCA is better than its single-task performance, and our HCA outperforms the state-of-the-art CAs for response retrieval across multiple evaluation datasets. Our case study shows that our proposed HCA can demonstrate multiple human traits and produce consistent, informative, and empathetic responses.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview and Scope

Language is the mark of humanity and sentience, and conversation or dialogue is one of the most fundamental and specially privileged arenas of language. It is the first kind of language we learn as children, and for most of us, it is the kind of language we most commonly indulge in [9].

In the field of artificial intelligence (AI) or, more specifically, natural language processing (NLP), a conversational agent (CA), also known as a dialogue system (DS), is an intelligent machine that can converse with humans in natural language. There are primarily three types of CAs:

- **Task-oriented CAs**: Task-oriented CAs are used to assist users for specific tasks in specific domains. For example, a task-oriented CA can assist a user to book a flight ticket by asking and acknowledging the required information, e.g., destination and departure date, in a conversational manner. Task-oriented CAs are usually measured by task success rate and dialogue efficiency, i.e., number of turns to complete the task, through both automatic and human evaluations.
- **Question-answering (QA) systems**: QA systems are expected to provide accurate and concise answers to users' questions based on rich knowledge sources such as databases, knowledge bases, and web documents. For example, a QA system may be able to answer a user's question "what is the

diameter of Earth?" with "12,742 km". QA systems are usually measured by answer correctness, e.g., exact match (EM) and F1, through both automatic and human evaluations.

- **Open-domain CAs**: Open-domain CAs, also known as chatbots, are designed to chat with users in any topics engagingly with the aim of establishing long-term relationships. The conversations between open-domain CAs and humans are generally chit-chat. For example, given a user message "how are you doing recently?", the open-domain CA may respond with "great, just finished my PhD thesis!". Open-domain CAs are usually measured by fluency, relevancy, consistency, and informativeness through human evaluation.

A large quantity of human conversations centers on socialization and chit-chat [10]. Open-domain CAs are essential in modern conversational user interfaces and have been adopted in numerous business domains such as persona assistant (Apple Siri, Amazon Alexa, Alibaba TmallGenie, etc.), customer support (Alibaba AliMe, etc.), education [11, 12], and healthcare [13, 14]. In recent years, various competitions in both research communities and industries have also been organized to promote the research and development of open-domain CAs, e.g., NeurIPS ConvAI competitions[1] and Amazon Alexa Prize challenges[2]. In this thesis, we focus on open-domain CAs. Moreover, we only focus on text-based open-domain CAs because text is the foundation of other modalities[3].

Building a human-level open-domain CA has been one of the fundamental objectives in AI research [15] and is still far from being solved. Alan Turing, an AI pioneer, proposed Turing Test[4] to test the general intelligence level of a machine. Specifically, a machine is said to have passed the Turing Test if a human talking to it (via text) cannot tell whether the machine is a machine or a human. Early studies in open-domain CAs can be traced back to ELIZA [16], the first CA developed in the 60's, which simulates a Rogerian psychotherapist based on hand-crafted pattern matching rules. In recent years, due to the growing amount of data and computation, and the advances of deep learning [17], end-to-end data-driven neural network based models are emerging and have become the dominant

[1] http://convai.io/

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

[3] For example, a voice-based CA usually comprises an automatic speech recognition (ASR) module, a text-based CA module, and a text-to-speech (TTS) module.

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

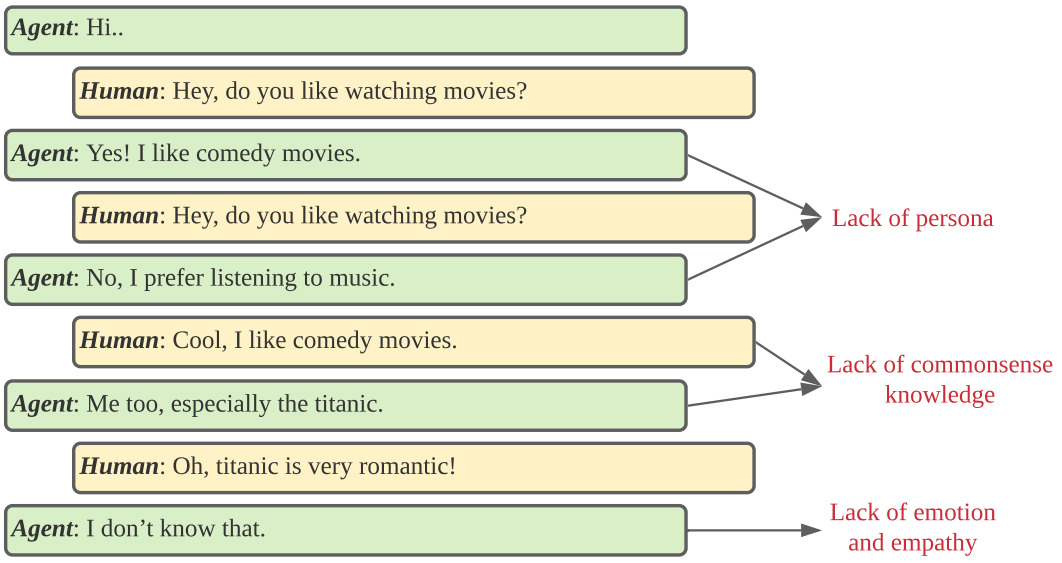


FIGURE 1.1: An example human-agent conversation illustrating the limitations of existing open-domain CAs.

approach in open-domain CAs [18–20]. Neural approaches have been shown to be more effective, scalable, and require less human efforts than traditional rule-based approaches for open-domain CAs [20].

However, despite having large neural networks and being trained on large-scale human-human conversation datasets, existing open-domain CAs often fail to model the traits of humans and exhibit the following limitations [2, 7, 21–25]: 1) they lack emotional intelligence and cannot generate or recognize emotions in conversations, which often lead to dull or generic responses; 2) they lack commonsense knowledge and often produce incoherent or unrelated responses; 3) they lack persona and often produce inconsistent responses; and 4) they lack empathy and often produce non-empathetic responses.

Figure 1.1 illustrates the aforementioned limitations in an example human-agent conversation. First, the agent shows the lack of consistent persona by producing self-contradictory responses regarding whether it likes watching movies. Second, the agent shows the lack of commonsense knowledge by categorizing the movie Titanic into comedy and producing logically incorrect responses. Third, the agent shows the lack of emotion and empathy by producing a dull and generic response "I don't know that." to an emotional user message "Oh, titanic is very romantic!".

Addressing the aforementioned limitations is important for bridging the gap between existing CAs and human-level CAs [23–25]. Humanization is the process of attributing human traits to an entity. In this thesis, we propose to address the aforementioned limitations by humanizing open-domain CAs with the following human traits: emotion, commonsense, personality, and empathy[5]. These human traits have been extensively studied in artificial agents and human-computer interaction (HCI) [26–30]. However, endowing these human traits into open-domain CAs is largely unexplored [23–25]. Our motivations for focusing on these four human traits are as follows:

- **Emotion**: Emotion is a strong feeling deriving from one's circumstances, mood, or relationships with others[6]. Emotions are prevalent in human conversations and crucial for building long-term relationships between interlocutors [31]. Endowing CAs with emotions has been shown to improve the mental health of participants in healthcare [32, 33], and the user satisfaction in customer support [34].

- **Commonsense**: Commonsense involves not only the basic beliefs of a particular society but also the fundamental presuppositions of all human knowledge [35]. Humans ground on socially shared commonsense knowledge to converse with each other. Incorporating commonsense knowledge into CAs has been shown to improve conversation understanding, make responses more reasonable and informative, and therefore improve user satisfaction [22, 36].

- **Persona**: Persona refers to the social face an individual presents to the world [37]. Persona is different from personality[7] but they have been shown to be highly correlated [38]. Humans have different personalities, and the differences in personality reflect on conversations, either implicitly or explicitly [39]. Augmenting CAs with personas has been shown to make the responses more consistent [7, 40], and improve user trust and engagement in education [41] and customer support [42].

[5]There are many other human traits that we do not consider in this thesis, e.g., curiosity, persuasion, and humor, which are left to future work.

[6]The definition is taken from Oxford dictionary.

[7]Persona is considered more subjective because it is the appearance an individual wants to present to the world, whereas personality is considered more objective because it is the characteristics of an individual.

- **Empathy**: Empathy refers to the capacity to understand or feel another's mental states and respond appropriately [43]. Empathy is vital in building good interpersonal relationships during conversations [44]. Endowing CAs with empathy can improve user task performance [45, 46] and build long-term relationships with users [47].

In this thesis, we propose several novels methods to incorporate the aforementioned human traits into open-domain CAs. We further conduct extensive experiments to analyze the impacts of these human traits on response quality and human ratings. All in all, our thesis makes a step towards humanized open-domain CAs.

1.2 Thesis Contributions

The key contributions of this thesis are as follows:

- We study the problem of incorporating emotion into response generation. To this end, we propose an Affect-Rich Seq2Seq model (AR-S2S), which is an emotional open-domain CA that can generate natural and emotional responses. Specifically, we leverage an external word-emotion lexicon, and propose an emotion-aware attention mechanism and a novel objective function to encourage the generation of emotional responses. Human evaluation results show that endowing open-domain CAs with emotion improves both content and emotion qualities of the generated responses, and our AR-S2S outperforms competitive baselines. This work is previously published as [48].
- We investigate the problem of emotion recognition from conversations. We propose a Knowledge-Enriched Transformer (KET) to tackle this problem. Specifically, based on the Transformer [49], we propose a hierarchical self-attention to model the hierarchical structure of conversation and further incorporate commonsense knowledge into emotion recognition. Experimental results show that both context and commonsense knowledge are beneficial to the recognition performance, and our KET outperforms competitive baselines on most of the benchmark datasets in terms of F1. This work is previously published as [50].

- We test the hypothesis that combining commonsense and emotion into open-domain CAs can improve response quality. To this end, we propose a novel model for Commonsense-Aware Response generation with specified Emotions (CARE). Specifically, given a message and the desired emotion, we first construct plausible relational and emotional latent concepts of the response, and then incorporate them into response generation. Experimental results support our hypothesis and show that our CARE can achieve better human ratings than competitive commonsense-aware CAs and emotional CAs. This work is previously published as [51].

- We study the problem incorporating persona and empathy into open-domain CAs and investigate the impact of persona on empathetic responding. To facilitate our study, we propose the first dataset for Persona-based Empathetic Conversations (PEC). We then propose a BERT-based model with multi-hop co-attention (CoBERT) for effective and efficient response retrieval. Experimental results show that our CoBERT can produce persona-based empathetic conversations and outperforms competitive baselines on response retrieval. In addition, human evaluation results show that both persona and empathy are beneficial to response quality and human ratings. Finally, experimental analysis reveals an empirical link between persona and empathy in human conversations, which may suggest that persona has a greater impact on empathetic conversations than non-empathetic ones. This work is previously published as [52].

- We investigate the problem of addressing all the aforementioned limitations of existing open-domain CAs, i.e., lacking emotion, commonsense, persona, and empathy. To this end, we propose a Humanized open-domain CA (HCA) that possesses all proposed human traits simultaneously through multi-task learning (MTL). Specifically, HCA is a retrieval-based CA and follows the pretrain-and-finetune paradigm for MTL. We leverage CoBERT as the base model and propose an improved training strategy for MTL. Experimental results show that the multi-task performance of HCA is often better than its single-task performance, and HCA outperforms competitive baselines on response retrieval across multiple evaluation datasets. In addition, our case study shows that HCA can demonstrate multiple human traits and produce consistent, informative, and empathetic responses.

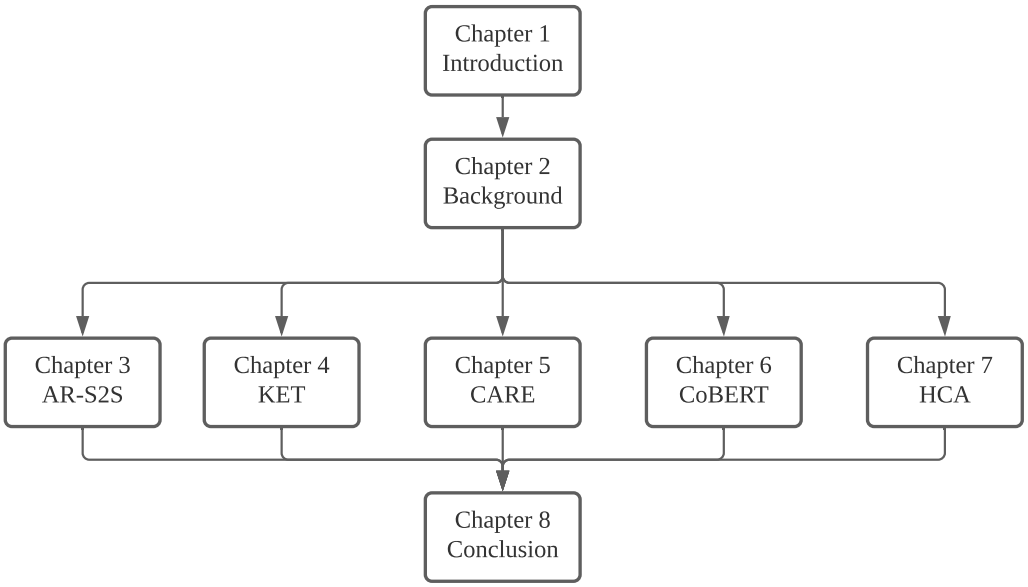


FIGURE 1.2: Thesis organization.

1.3 Thesis Organization

The organization of this thesis is depicted in Figure 1.2.

Chapter 1 presents the overview and scope of this thesis.

Chapter 2 introduces the preliminaries of this thesis and a brief review of open-domain CAs.

Chapter 3 focuses on emotion and introduces AR-S2S for emotional response generation.

Chapter 4 focuses on emotion and commonsense and introduces KET for emotion recognition from conversations.

Chapter 5 focuses on emotion and commonsense and introduces CARE for commonsense-aware emotional response generation.

Chapter 6 focuses on persona and empathy and introduces CoBERT for persona-based empathetic response retrieval.

Chapter 7 combines emotion, commonsense, persona, and empathy, and proposes HCA for humanized response retrieval.

Chapter 8 concludes this thesis and discusses future research directions.

Chapter 2

Background

In this chapter, we first introduce commonly used neural architectures for open-domain conversational agents (CAs). We then introduce different types of open-domain CAs. Finally, we present a brief literature review on the human traits in open-domain conversational agents.

2.1 Neural Architectures

Neural architectures are prevalent in recent open-domain CAs due to their advantages of being effective and scalable. In this section, we briefly cover commonly used neural architectures for open-domain CAs[1].

2.1.1 Recurrent Neural Networks

Recurrent neural networks (RNN) is a type of neural networks specifically designed for learning the representation of sequential data. A conversation is a sequence of utterances, and each utterance is a sequence of words. Hence, recurrent neural networks are especially suitable and widely used for modeling utterances and conversations.

Formally, given a sequence of word tokens $\{x_1, x_2, ..., x_m\}$, where each x_t, $t = 1, ..., m$ is represented as a k-dimensional dense word vector $\mathbf{x}_t \in \mathbb{R}^k$ [53, 54], the

[1]Note that the bias terms in all equations of this thesis are dropped for brevity.

hidden state $\mathbf{h}_t \in \mathbb{R}^d$ for x_t is recurrently computed by $\mathbf{h}_{t-1}$ and $\mathbf{x_t}$ as follows:

$$\mathbf{h}_t = f(\mathbf{h}_{t-1}, \mathbf{x}_t), \tag{2.1}$$

where f can be any function and $\mathbf{h}_0$ is usually initialized as a zero vector. In a standard RNN, f is a non-linear transformation:

$$f = \phi(\mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{W}_{xh}\mathbf{x}_t), \tag{2.2}$$

where $\mathbf{W}_{hh} \in \mathbb{R}^{d\times d}$, $\mathbf{W}_{xh} \in \mathbb{R}^{d\times k}$ denote model parameters, and ϕ denotes a non-linear activation function such as Sigmoid, Tanh or ReLU. Refer to [55] for a detailed discussion of various activation functions.

LSTM: The standard RNN has been shown to be difficult to learn long-range dependencies between tokens in a sequence [56]. Hochreiter and Schmidhuber [57] proposed Long Short-Term Memory network (LSTM) to address this problem using gates. Specifically, the dynamics of LSTM are as follows:

$$\begin{aligned}
\mathbf{f}_t &= \sigma(\mathbf{W}_f[\mathbf{h}_{t-1}; \mathbf{x}_t] \\
\mathbf{i}_t &= \sigma(\mathbf{W}_i[\mathbf{h}_{t-1}; \mathbf{x}_t] \\
\mathbf{o}_t &= \sigma(\mathbf{W}_o[\mathbf{h}_{t-1}; \mathbf{x}_t] \\
\mathbf{c}_t^{'} &= \tanh(\mathbf{W}_C[\mathbf{h}_{t-1}; \mathbf{x}_t] \\
\mathbf{c}_t &= \mathbf{f}_t * \mathbf{c}_{t-1} + \mathbf{i}_t * \mathbf{c}_t^{'} \\
\mathbf{h}_t &= \mathbf{o}_t * \tanh(\mathbf{c}_t),
\end{aligned} \tag{2.3}$$

where $\mathbf{W}_f, \mathbf{W}_i, \mathbf{W}_o, \mathbf{W}_C \in \mathbb{R}^{d\times(d+k)}$ denote model parameters, σ denotes the Sigmoid function, and $[;]$ denotes vector concatenation.

Intuitively, the cell state $\mathbf{c}_{t-1}$ stores the information along the sequence upon time step $t-1$. When given a new input $\mathbf{x}_t$, LSTM decides what information to forget and what new information to store. These two operations are controlled by the forget gate $\mathbf{f}_t$ and the input gate $\mathbf{i}_t$, respectively. The $\mathbf{c}_t^{'}$ denotes the candidate cell state, which is squeezed to $[-1, 1]$ by Tanh. The new cell state $\mathbf{c_t}$ is jointly controlled by the previous cell state $\mathbf{c_{t-1}}$ and the candidate cell state $\mathbf{c}_t^{'}$. Finally, LSTM outputs $\mathbf{h}_t$, which is a filtered version of $\mathbf{c}_t$, controlled by Tanh and the output gate $\mathbf{f}_o$.

GRU: Another commonly used variant of RNN is the Gated Recurrent Units (GRU) [58]. GRU is also designed to tackle the long-dependency issue of traditional RNN. Specifically, GRU simplifies the connections of LSTM mainly in two ways: (1) it merges the cell state $\mathbf{c}$ and the hidden state $\mathbf{h}$ in LSTM into a single hidden state $\mathbf{h}$, (2) it merges the forget gate $\mathbf{f}$ and the input gate $\mathbf{i}$ in LSTM into a single update gate $\mathbf{z}$. As such, GRU has fewer parameters than LSTM. In practice, GRU has been shown to deliver comparable performance across various sequence modeling tasks [59].

Formally, the dynamics of GRU are as follows:

$$\begin{aligned} \mathbf{z}_t &= \sigma(\mathbf{W}_z[\mathbf{h}_{t-1}; \mathbf{x}_t] \\ \mathbf{r}_t &= \sigma(\mathbf{W}_r[\mathbf{h}_{t-1}; \mathbf{x}_t] \\ \mathbf{h}_t^{'} &= \tanh(\mathbf{W}_h[r_t * \mathbf{h}_{t-1}; \mathbf{x}_t] \\ \mathbf{h}_t &= (1 - \mathbf{z}_t) * \mathbf{h}_{t-1} + \mathbf{z}_t * \mathbf{h}_t^{'} \end{aligned} \tag{2.4}$$

where $\mathbf{W}_z, \mathbf{W}_r, \mathbf{W}_r \in \mathbb{R}^{d \times (d+k)}$ denote model parameters.

Intuitively, given $\mathbf{h}_{t-1}$ and a new input $\mathbf{x}_t$, GRU decides what information to update and reset, controlled by the update gate $\mathbf{z}_t$ and the reset gate $\mathbf{r}_t$, respectively. The $\mathbf{h}_t^{'}$ denotes the candidate hidden state controlled by the reset gate $\mathbf{r}_t$. Finally, GRU outputs $\mathbf{h}_t$, which is a weighted sum of $\mathbf{h}_{t-1}$ and $\mathbf{h}_t^{'}$, controlled by the update gate $\mathbf{z}_t$.

Other RNN Variants: There are also many other RNN variants in the literature [60]. Xingjian et al. [61] proposed convolutional LSTM (ConvLSTM) to better model spatio-temporal correlations. Tai et al. [62] proposed tree-LSTM to learn tree-structured network topologies. Bradbury et al. [63] proposed quasi-recurrent neural networks (QRNNs) for parallel learning across multiple timesteps.

2.1.2 Sequence to Sequence with Attention

Sequence to Sequence neural networks (Seq2Seq) [58, 64] is a type of neural architectures for mapping an input sequence to an output sequence. Seq2Seq was originally proposed for neural machine translation, where a sentence in a source language needs to be translated into a sentence in a target language. After its invention, Seq2Seq becomes widely adopted in many other sequence mapping problems

such as conversation generation and text summarization, etc. Recent open-domain generative CAs are mainly based on Seq2Seq architecture.

Formally, Seq2Seq has an encoder-decoder architecture where the encoder is used to encode a variable length input sequence $X = (x_1, x_2, ..., x_T)$ into a vector of fixed dimension $\mathbf{h_T}$ and the decoder is used to decode $\mathbf{h_T}$ into a variable length output sequence $Y = (y_1, y_2, ..., y_{T'})$. The objective function of Seq2Seq is to maximize

$$\begin{aligned} p(Y|X) &= p(y_1|\mathbf{h}_T) \prod_{t'=2}^{T'} p(y_{t'}|\mathbf{h}_T, y_1, ..., y_{t'-1}), \\ \mathbf{h}_t &= f(\mathbf{h}_{t-1}, x_t), \forall t = 1, 2, ..., T, \end{aligned} \tag{2.5}$$

where $\mathbf{h_t}$ denotes the hidden state of the input sequence X at time step t and $\mathbf{h_0}$ is usually initialized as a zero vector. The function f denotes a non-linear transformation, which usually takes the form of recurrent models such as LSTM in Equation 2.3 or GRU in Equation 2.4[2].

After encoding X as $\mathbf{h_T}$, the decoder updates its decoder hidden state $\mathbf{s_{t'}}$ by taking the previous hidden state $\mathbf{s}_{t'-1}$ and previous output $y_{t'-1}$ as inputs:

$$\mathbf{s}_{t'} = g(\mathbf{s}_{t'-1}, y_{t'-1}), \forall t^{'} = 1, 2, ..., T^{'}, \tag{2.6}$$

where g is usually another recurrent model such as LSTM or GRU for autoregressive decoding, $\mathbf{s}_0 = \mathbf{h_T}$, and y_0 denotes the start of sequence (SOS) token.

Finally, the output word probability in Equation 2.5 is computed by

$$p(y_{t'}) = \mathit{softmax}(\mathbf{W}^o \mathbf{s}_{t'}), \forall t^{'} = 1, 2, ..., T^{'}, \tag{2.7}$$

where $\mathbf{W}^o$ denotes a model parameter and *softmax* denotes the softmax function.

One problem of Seq2Seq is the limited representation power of the encoder representation $\mathbf{h}_T$ on which the entire decoding process is conditioned. The attention mechanisms [65, 66] are proposed to solve this problem. Specifically, at each decoding time step $t^{'}, \forall t^{'} = 1, 2, ..., T^{'}$, the decoder pays attention to different parts of the input sequence by computing a context vector $\mathbf{c}_{t'}$ as the weighted average

[2]Both the encoder and the decoder in Seq2Seq can be non-recurrent models, such as the Transformer [49] (see Section 2.1.3).

of all input hidden states $\mathbf{h_t}, \forall t = 1, 2, ..., T$, as follows:

$$\mathbf{c}_{t'} = \sum_{t=1}^{T} \alpha_{t't} \mathbf{h}_t, \tag{2.8}$$

where the alignment weight $\alpha_{t't}$ is given by

$$\alpha_{t't} = \frac{\exp(e_{t't})}{\sum_{k=1}^{T} \exp(e_{t'k})}, \tag{2.9}$$

where $e_{t't} = \mathit{score}(\mathbf{h}_t, \mathbf{s}_{t'})$ is the energy function that computes the energy or score between the input hidden state $\mathbf{h}_t$ and the output hidden state $\mathbf{s}_{t'}$. This energy function is usually implemented as a dot product or a Multilayer Perceptron (MLP).

The context vector $\mathbf{c}_{t'}$ is then concatenated with the decoder hidden state $\mathbf{s}_{t'}$ to form an attentional hidden state $\hat{\mathbf{s}}_{\mathbf{t}'}$ as follows:

$$\hat{\mathbf{s}}_{\mathbf{t}'} = \tanh(\mathbf{W}^c[\mathbf{c}_{t'}; \mathbf{s}_{t'}]). \tag{2.10}$$

Finally, $\hat{\mathbf{s}}_{\mathbf{t}'}$ replaces $\mathbf{s}_{t'}$ in Equation 2.7 to compute the output word probability.

2.1.3 Transformer

Despite that the recurrent models in Section 2.1.1 use gates to model the information flow along a sequence, they still suffer the problem of being difficult in learning long-range dependencies [49]. Another problem of recurrent models is their large computational complexity, because the recurrence in these models cannot be computed in parallel. To alleviate the aforementioned problems, Vaswani et al. [49] proposed the Transformer, a non-recurrent Seq2Seq model, by replacing all recurrent connections with attention. Specifically, the Transformer uses self-attention to learn dependencies between tokens within a sequence and cross-attention to learn dependencies between tokens in two sequences. The sequential information in tokens is modeled by positional embeddings.

Formally, given a variable length input sequence $X = (x_1, x_2, ..., x_T)$ and a variable length output sequence $Y = (y_1, y_2, ..., y_{T'})$, the Transformer can be decomposed

into an encoder and a decoder. The dynamics of the encoder are as follows:

$$\begin{aligned}
\mathbf{x}_t &= \mathit{Embed}(x_t) + \mathit{Pos}(x_t) \\
\mathbf{X}^{'} &= \mathbf{X} + \mathit{LayerNorm}(\mathit{MultiHead}(\mathbf{X}, \mathbf{X}, \mathbf{X})) \\
\mathbf{X}^{''} &= \mathbf{X}' + \mathit{LayerNorm}(\mathit{FF}(\mathbf{X}')),
\end{aligned} \tag{2.11}$$

where $\mathbf{X} \in \mathbb{R}^{T \times d}$ denotes the sequence of $\mathbf{x}_t \in \mathbb{R}^d, t = 1, ..., T$, $\mathbf{X}'' \in \mathbb{R}^{T \times d}$ denotes the encoder output, *Embed* denotes a word embedding layer, *Pos* denotes a positional embedding layer [49], *MultiHead* denotes a multi-head attention layer described in Equation 2.12 below, *FF* denotes a feedforward layer described in Equation 2.13 below, and *LayerNorm* denotes a layer normalization layer [67].

$$\begin{aligned}
\mathit{MultiHead}(Q, K, V) &= [\mathit{head}_1; ...; \mathit{head}_h]W^O \\
\mathit{head}_i &= \mathit{Attention}(QW_i^Q, KW_i^K, VW_i^V) \\
\mathit{Attention}(Q, K, V) &= \mathit{softmax}(\frac{QK^T}{\sqrt{d_s}})V
\end{aligned} \tag{2.12}$$

$$\mathit{FF}(X) = \mathit{max}(0, XW_1)W_2, \tag{2.13}$$

where $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d \times d_k}$, $W^O \in \mathbb{R}^{d \times d}$, $W_1 \in \mathbb{R}^{d \times d_{ff}}$, and $W_2 \in \mathbb{R}^{d_{ff} \times d}$ denote model parameters. The multi-head attention layer models word-word dependencies in multiple subspaces using multiple heads.

The dynamics of the decoder are as follows:

$$\begin{aligned}
\mathbf{y}_t &= \mathit{Embed}(y_t) + \mathit{Pos}(y_t) \\
\mathbf{Y}^{'} &= \mathbf{Y} + \mathit{LayerNorm}(\mathit{MultiHead}(\mathbf{Y}, \mathbf{Y}, \mathbf{Y})) \\
\mathbf{Y}^{''} &= \mathbf{Y}^{'} + \mathit{LayerNorm}(\mathit{MultiHead}(\mathbf{Y}^{'}, \mathbf{X}^{''}, \mathbf{X}^{''})) \\
\mathbf{Y}^{'''} &= \mathbf{Y}^{''} + \mathit{LayerNorm}(\mathit{FF}(\mathbf{Y}^{''})) \\
\mathbf{O} &= \mathit{softmax}(\mathbf{Y}^{'''}W_3),
\end{aligned} \tag{2.14}$$

where $\mathbf{Y} \in \mathbb{R}^{T^{'} \times d}$ denotes the sequence of $\mathbf{y}_t \in \mathbb{R}^d, t = 1, ..., T^{'}$, $\mathbf{Y}''' \in \mathbb{R}^{T^{'} \times d}$ denotes the decoder output before softmax, $W_3 \in \mathbb{R}^{d \times V}$ denotes a model parameter and $\mathbf{O} \in \mathbb{R}^{T' \times V}$ denotes the output token probabilities.

Both the encoder and the decoder can be stacked multiple times to form a multi-layer Transformer. The entire Transformer can be trained using cross-entropy loss between $\mathbf{O}$ and Y to maximize the probabilities of ground-truth tokens. Note that

1) the Y in Equation 2.14 is shifted one position to the right during training to learn the next token prediction, and 2) at each decoding step, the tokens at the right side are masked to prevent the attention from attending future tokens.

Compared to RNN-based Seq2Seq models, the Transformer has the advantages of the shorter path of dependencies between tokens, smaller computational complexity[3], and better parallelization.

In recent years, extensive research efforts have been made to improve the Transformer [68–74]. Dai et al. [68] proposed Transformer-XL to model longer sequences with faster speed than the Transformer. Kitaev et al. [70] proposed Reformer for more memory-efficient and faster learning on long sequences. [71] proposed Longformer with an attention mechanism that scales linearly with sequence length to model long sequences.

2.1.4 BERT

Due to the computational advantage of the Transformer and its competitive performance in various sequence modeling tasks such as machine translation [49] and language modeling [68], researchers have recently investigated pretraining the Transformer on a large collection of corpus to achieve better finetuning results [75–77]. One notable example is BERT [76], which is one of the most popular Transformer-based pretrained language models.

BERT is based on the Transformer encoder (see Equation 2.11) and pretrained using the masked language model (MLM) and the next sentence prediction (NSP) objectives on the BooksCorpus (800M words) [78] and English Wikipedia (2,500M words) datasets. The finetuned BERT has achieved the state-of-the-art results on multiple language understanding tasks and QA tasks, suggesting that BERT has successfully learned powerful text representation and is easy to transfer to downstream tasks. BERT started a new era of NLP where the paradigm of pretraining on a large corpus and then finetuning on small downstream tasks becomes standard practice. Recently, numerous BERT-inspired pretrained language models are proposed for better performance [79, 80], smaller model size [81, 82], faster inference [83, 84], cross-lingual support [85, 86], and language generation [87], etc.

[3] This is true for short sequences where the sequence length is smaller than the word embedding size.

2.2 Open-Domain Conversational Agents

Open-domain conversational agents (CAs) are generally classified into three types: rule-based, retrieval-based, and generation-based. In this section, we briefly introduce these three types of open-domain CAs and then summarize their advantages and disadvantages.

2.2.1 Rule-Based Models

CAs in the early stages are mainly based on rules. For example, the first chatbot ELIZA [16] in the 60's maintains a dictionary of ranked keywords and their associated patterns. If a keyword is identified in a given user input, the corresponding transform will be applied to the keyword and subsequently incorporated into the response. If no keyword matches, ELIZA will output a generic response such as "PLEASE GO ON". Later in the 70's, another rule-based chatbot PARRY [88], used for studying schizophrenia, included additional rules to maintain a mental state of the chatbot. For example, certain user inputs may lead PARRY to an angry mental state, and PARRY may output hostile responses. Both ELIZA and PARRY achieved great success as they were able to deliver human-like responses in a few specific scenarios. After the success of ELIZA and PARRY, more sophisticated rules and logic were developed into chatbots, such as Jabberwacky[4], ALICE[5], and cleverbot[6].

In recent years, rule-based chatbots are less popular due to their following drawbacks: 1) the number of rules grows dramatically as the conversation scenarios become more complex; 2) significant human efforts are required to rewrite the rules for new languages or domains; 3) their performance is often surpassed by recent retrieval-based and generation-based models.

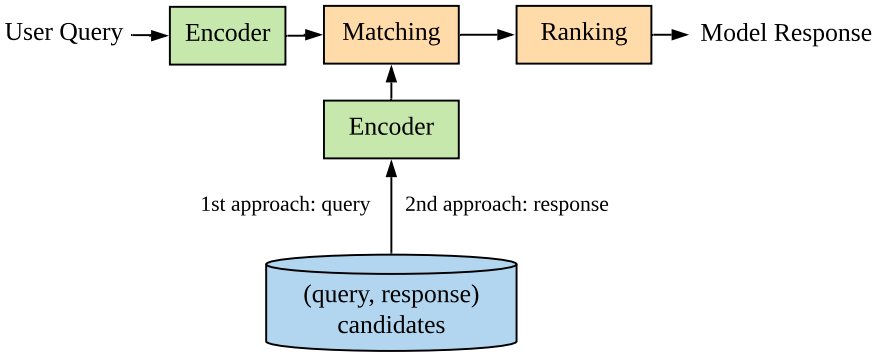


FIGURE 2.1: Illustration of a retrieval-based CA.

2.2.2 Retrieval-Based Models

Given a user query, retrieval-based open-domain CAs rely on information retrieval (IR) techniques to select a response from a candidate pool of {query, response} pairs. The representation learning and the matching algorithm of queries and responses are two central research problems of retrieval-based models. Various models can be applied to represent queries and responses, e.g., the bag of words model (BoW) or neural networks such as RNN, convolutional neural networks (CNN), Transformer, and BERT. The matching algorithms between query and response representations are based on ranking algorithms in IR or neural attention-based fine-grained matching. The matching output is usually computed by relevancy measures such as cosine similarity, dot product, or MLP [20]. Given a representation learning method and a matching algorithm, there are generally two types of approaches to response retrieval: 1) select the response of the most relevant query and 2) select the most relevant response. An illustration of retrieval-based models is shown in Figure 2.1.

In the first approach, the model first selects the most relevant query with respect to the user query from a list of candidate queries, and then return the response to the selected query as the model response [89, 90].

In the second approach, the model directly selects the response that is most relevant to the user query from a list of candidate responses [19, 91].

[4]https://en.wikipedia.org/wiki/Jabberwacky
[5]https://en.wikipedia.org/wiki/Artificial_Linguistic_Internet_Computer_Entity
[6]https://en.wikipedia.org/wiki/Cleverbot

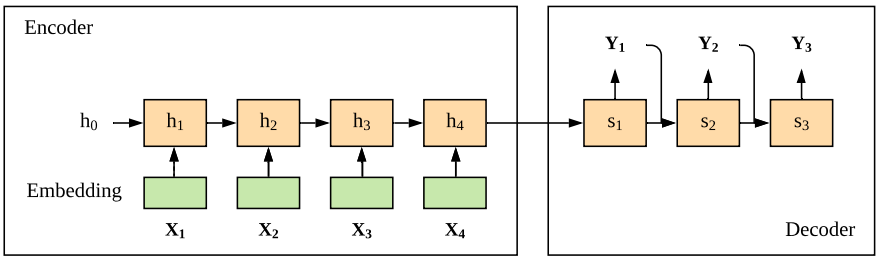


FIGURE 2.2: Illustration of a generation-based CA using the Seq2Seq architecture. $X = X_1, X_2, X_3, X_4$ denotes a user query, and $Y = Y_1, Y_2, Y_3$ denotes a model response.

The first approach seems more intuitive; however, in practice, the second approach often performs better [91], possibly because the two-step selection in the first approach introduces more noises than the direct selection in the second approach [9]. Modern neural retrieval-based open-domain CAs [19, 92–94] are mainly based on the second approach.

2.2.3 Generation-Based Models

Generation-based open-domain CAs are largely inspired by neural machine translation where a sentence in a target language is generated given this sentence in a source language. Ritter et al. [91] presented the first attempt to address response generation under the framework of phrase-based machine translation. Later, with the invention of Seq2Seq [58, 64] (see Section 2.1.2) for neural machine translation, neural network based response generation models are becoming dominant [18, 20, 95, 96]. An illustration of a Seq2Seq-based generative CA is shown in Figure 2.2. In recent years, many attempts have been made to improve the qualities of generated responses such as more diversity [97, 98], more informativeness [99, 100], and less repetitions [101].

One major limitation of Seq2Seq-based response generation models is the incapability of modeling the inherent one-to-many relationships between query and responses in human conversations. In other words, there are often many appropriate responses to a query. Researchers have recently proposed conditional variational autoencoders (CVAE) [102–104] to model such relationships using latent variables over potential conversational intents. CVAE-based response generation models have been shown to improve response diversity over Seq2Seq-based models [102].

2.2.4 Summary

We present a summary of the advantages and disadvantages of rule-based, retrieval-based, and generation-based models.

Compared to rule-based models, both retrieval-based and generation-based models have the advantages of being end-to-end trainable, scalable, and language-independent. However, rule-based models are more robust and interpretable. Hence, hand-crafted rules are still actively used in commercial CAs.

Compared to retrieval-based models, generation-based models have the advantage of being able to generate novel responses. In addition, generation-based models do not require a candidate pool to work. However, in the industry, retrieval-based models are often preferred because 1) they always produce grammatical responses because the responses are selected from a candidate pool of human responses and 2) developers have more control over retrieval-based models because they can always filter out unwanted responses, e.g., toxic responses, from the candidate pool. In human evaluations, generation-based models are usually favored less than retrieval-based models. However, recent transformer-based generative models began to achieve better human ratings than retrieval-based models when the generative models are first pretrained on a large collection of generic conversations and then finetuned on a small number of domain-specific conversations [105, 106].

2.3 Human Traits in Open-Domain Conversational Agents

In this section, we present a brief literature review on the human traits in open-domain conversational agents (CA), such as emotion, commonsense, persona, empathy, etc.

2.3.1 Emotion

Emotion is a strong feeling deriving from one's circumstances, mood, or relationships with others. The incorporation of emotion into CAs dates back to rule-based

CAs [107–109]. For example, Ochs et al. [107] designed a CA that can express emotions based on cognitive appraisal theories [110], which require numerous event-handling rules to be implemented. In recent years, several studies [3, 21, 111, 112] proposed to incorporate emotion into generation-based CAs. For example, [21] proposed ECM, a Seq2Seq conversational model, to generate responses with user-specified emotions. In these studies, emotion has been observed to improve the engagingness and user satisfaction of CAs.

Besides emotional response generation, another line of research in emotional CAs is conversational emotion recognition. Early related studies focus on call center dialogs using lexicon-based methods and audio features [113–115]. In recent years, neural networks are becoming the dominant approaches [116–120]. RNNs and CNNs are often used for modelling the sequential structure and extracting discriminative features of conversations, respectively. For example, Poria et al. [121] proposed an LSTM-based model to capture contextual information extracted by CNN for sentiment analysis in conversational videos.

2.3.2 Commonsense

Commonsense involves not only the basic beliefs of a particular society but also the fundamental presuppositions of all human knowledge [35]. CAs often require commonsense to produce logical and relevant responses. In recent years, several studies investigated the problem of incorporating commonsense into CAs [22, 36, 122–126]. The commonsense knowledge are usually represented by knowledge graphs or unstructured text documents, and often directly incorporated into model training as additional input. For example, Zhou et al. [22] proposed CCM to incorporate commonsense knowledge by applying attention mechanisms on 1-hop knowledge triplets for response generation. In these studies, commonsense has been observed to improve the relevancy and informativeness of the responses of CAs.

2.3.3 Persona

Persona refers to the social face an individual presents to the world [37]. CAs often require persona to produce consistent responses. In recent years, personalized CAs are emerging [7, 40, 127–129]. The personas of CAs are usually represented by

descriptive sentences of the speakers, e.g., "I live in Singapore" or "I like dogs". Similar to commonsense, a common approach to incorporating persona into CAs is to leverage persona sentences as additional input during model training. For example, Li et al. [40] presented an early attempt to learn persona embeddings on textual personal information and then incorporate them into a response generation model. In these studies, persona has been observed to improve the consistency of the responses of CAs.

2.3.4 Empathy

Empathy refers to the capacity to understand or feel another's mental states and respond appropriately [43]. Empathetic responses are helpful in building good interpersonal relationships in conversations [44]. Early empathetic CAs are primarily based on rules [45, 46, 130]. In recent years, more neural network models [2, 131–136] are proposed to produce empathetic responses. For example, Lin et al. [135] proposed a mixture of empathetic listeners that first captures the user emotion distribution at each turn and then softly combines the output states of each emotional listener to generate empathetic responses. In these studies, empathy has been observed to improve the user satisfaction of CAs.

2.3.5 Others

There are many other desirable human traits that are not studied in this thesis but left to future work, e.g., curiosity, persuasion, humor, etc. Curious CAs have been studied in online education to help students learn by teaching [137, 138]. Persuasive CAs are expected to persuade people to change their attitudes or behaviors through conversation. They have been applied to conversational recommendation scenarios to improve purchase rate [139] or encourage charity donations for social good [140]. Finally, humorous CAs have been shown to improve user satisfaction and task enjoyment through humorous responses [141, 142].

Chapter 3

Emotional Conversational Agents

3.1 Overview

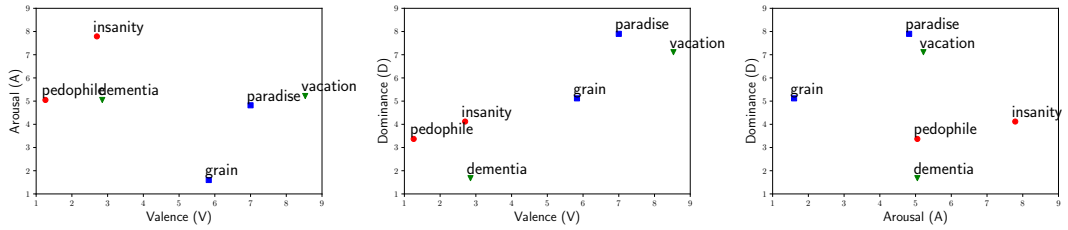
Emotion or affect[1] is a strong feeling deriving from one's circumstances, mood, or relationships with others. As a vital part of human intelligence, having the capability to recognize, understand, and express emotions like humans has been arguably one of the major milestones in artificial intelligence [143]. In this chapter, we investigate the problem of incorporating emotion into response generation[2].

Open-domain conversational agents (CAs) aim to produce natural, coherent, and engaging responses when given user messages. In recent years, Seq2Seq based generative conversational models [58, 64] (see Section 2.1.2) have been widely adopted due to the simplicity and scalability of Seq2Seq in modeling sequence mappings. To make neural conversational models more engaging, various techniques have been proposed, such as using latent variable [98] to promote response diversity and encoding topic [144] into conversational models to produce more coherent responses.

However, incorporating emotion into neural conversational models has seldom been explored, despite that it has many benefits such as improving user satisfaction [145], fewer breakdowns [146], and more engaged conversations [147]. For real-world applications, Fitzpatrick et al. [148] developed a rule-based empathetic chatbot to

[1] Affect encompasses a broad range of feelings, including emotion and mood, etc. In this chapter, we slightly abuse the definitions and use emotion and affect interchangeably.

[2] This chapter is published as *An Affect-Rich Neural Conversational Model with Biased Attention and Weighted Cross-Entropy Loss*, Proceedings of AAAI 2019 [48].



(A) V-A ratings (B) V-D ratings (C) A-D ratings

FIGURE 3.1: 2D plot of words with either highest or lowest ratings in valence (V), arousal (A) or dominance (D) in the corpus.

deliver cognitive behavior therapy to young adults with depression and anxiety, and obtained significant results on depression reduction.

Despite these benefits, there are a few challenges in the incorporation of emotion into neural conversational models that existing approaches fail to address: 1) It is difficult to capture the emotion of a sentence, partly because negators and intensifiers often change its polarity and strength. Handling negators and intensifiers properly still remains a challenge in sentiment analysis, and 2) It is difficult to embed emotions naturally in responses with correct grammar and semantics [149].

In this chapter, we propose a single-turn open-domain neural conversational model to address the aforementioned challenges to produce responses that are both natural and emotional. Our model extends the Seq2Seq with attention [66] (see Section 2.1.2). Specifically, we first leverage an external corpus [150] to provide affect knowledge for each word in the Valence, Arousal, and Dominance (VAD) dimensions [151]. We incorporate the affect knowledge into the embedding layer of our model. VAD notation has been widely used as a dimensional representation of human emotions in psychology. A 2D plot of selected words with extreme VAD values are shown in Figure 3.1. To capture the effect of negators and intensifiers, we then propose a novel biased attention mechanism that explicitly considers negators and intensifiers in attention computation. Finally, to maintain correct grammar and semantics, we train our Seq2Seq model with a weighted cross-entropy loss that encourages the generation of emotional words without degrading language fluency.

In summary, our main contributions in this chapter are as follows:

- For the first time, we propose a novel affective attention mechanism to incorporate the effect of negators and intensifiers in conversation modeling. Our

mechanism introduces only a small number of additional parameters.

- For the first time, we apply weighted cross-entropy loss in conversation modeling. Our affect-incorporated weights achieve a good balance between language fluency and emotion quality of model responses. Our empirical study does not show performance degradation in language fluency while producing emotional words.

- Overall, we propose *Affect-Rich Seq2Seq* (AR-S2S), a novel end-to-end affect-rich open-domain neural conversational model. Human preference tests show that our model is preferred over the state-of-the-art model in terms of both content quality and emotion quality by a large margin. In addition, our result analysis shows that incorporating emotion into open-domain conversational models have a positive impact on response quality and human ratings.

3.2 Related Work

Prior studies on emotional conversational agents mainly focused on rule-based systems, which require an extensive hand-crafted rule base. For example, Ochs et al. [107] designed an empathetic virtual agent that can express emotions based on cognitive appraisal theories [110], which require numerous event-handling rules to be implemented. Another example is the Affect Listeners [152], which are conversational systems aiming to detect and adapt to the affective states of users. However, their detection and adaptation mechanisms heavily rely on hand-crafted features such as letter capitalization, punctuation, and emoticons, and thus difficult to scale.

In recent years, there is an emerging research trend in end-to-end neural network based generative conversational models [18, 95]. To improve the content quality of neural conversational models, many techniques have been proposed, such as improving response diversity using Conditional Variational Autoencoders (CVAE) [102] and encoding commonsense knowledge using external facts corpus [153].

However, few studies investigated the problems in improving the emotion quality of neural conversational models. Emotional Chatting Machine (ECM) [21] is a Seq2Seq conversational model that generates responses with user-specified

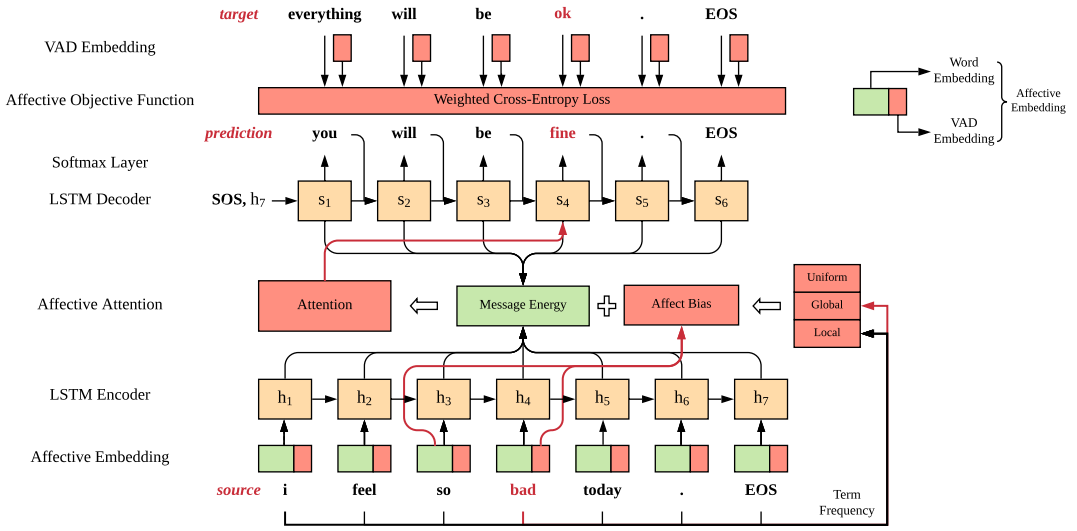


FIGURE 3.2: Overall architecture of our proposed AR-S2S. This diagram illustrates decoding "fine" and affect bias for "bad".

emotions. It employs an internal memory module to model implicit emotional changes and an external memory module to help generate more explicit emotional words. The main objective of ECM is to produce responses according to explicit user-specified emotions, while our model focuses on enriching affect in generated responses. Similar to ECM, Mojitalk [112] presents a few generative models, including Seq2Seq, CVAE, and Reinforced CVAE, to generate responses according to explicit user-specified emojis. Both ECM and Mojitalk do not explicitly consider emotions in input sentences when generating emotional responses. In comparison, our model pays extra attention to emotional words in input sentences during generation. The most relevant work to our study is [154], which introduces a Seq2Seq model with three extensions to incorporate affect into conversations. Similar to their work, we also adopt the approach of using VAD embedding to encode affects. However, we perform extra preprocessing on VAD embedding to improve model performance. In addition, we specifically consider the effect of negators and intensifiers via a novel affective attention mechanism when generating affect-rich responses.

3.3 Affect-Rich Seq2Seq (AR-S2S)

In this section, we present our proposed AR-S2S model, which extends the Seq2Seq with attention (see Section 2.1.2), as illustrated in Figure 3.2.

TABLE 3.1: Interpretations of clipped VAD embeddings.

Dimensions	Values	Interpretations
Valence	3 - 7	pleasant - unpleasant
Arousal	3 - 7	low intensity - high intensity
Dominance	3 - 7	submissive - dominant

3.3.1 Affective Embedding

Our model adopts Valence, Arousal, and Dominance (VAD) [151] attributes to encode word affects as vectors of size 3 from an annotated lemma-VAD pairs corpus [150]. This corpus comprises 13,915 lemmas with VAD values annotated in the $[1, 9]$ scale. To leverage this corpus, we assign VAD values to words based on their lemmas. To increase coverage, we extend the corpus to 23,825 lemmas using synonym expansion, i.e., assigning the average VAD values of their synonyms to absent lemmas. Furthermore, we empirically clip VAD values of all words to the $[3, 7]$ interval to prevent words with extreme VAD values from repeatedly showing in the generated responses, as observed in our preliminary experiments. The interpretations of clipped VAD embedding are presented in Table 3.1. For example, the word "nice" is associated with the clipped VAD values: (V: 6.95, A: 3.53, D: 6.47). For words whose lemmas are not in the extended corpus, comprising approximately 10% of the entire training vocabulary, we assign them VAD values of $[5, 3, 5]$, which are the clipped values of a neutral word. Note that a value of 3 in arousal (A) dimension is regarded as neutral because it has zero emotional intensity.

Finally, to remove bias, we normalize VAD embedding as $\overline{VAD}(x_t) = VAD(x_t) - [5, 3, 5]$, where $VAD(x_t) \in \mathbb{R}^3$ is the VAD embedding of word x_t. We incorporate VAD embedding by concatenation as follows:

$$\mathbf{e}(x_t) = [\mathbf{x_t}; \lambda \overline{VAD}(x_t)], \tag{3.1}$$

where $\mathbf{x_t} \in \mathbb{R}^m$ denotes the word embedding of x_t, $\mathbf{e}(x_t) \in \mathbb{R}^{m+3}$ denotes the final affective embedding of x_t, m denotes the dimensionality of word vectors, and $\lambda \in \mathbb{R}_+$ denotes the affect embedding strength hyper-parameter to tune the strength of VAD embeddings.

3.3.2 Affective Attention

To incorporate affect into attention naturally, we make the intuitive assumption that humans pay extra attention on affect-rich words during conversations. Specifically, our model pays biased attention to affect-rich words in the input sentences, and considers the effect of negators and intensifiers. Specifically, our model employs an affect bias η augmenting the affective strength of each word in the input sentences into the energy function (see Equation 2.9) as follows:

$$e_{t't} = \mathbf{h}_t{}^T \mathbf{s}_{t'} + \eta_t, \tag{3.2}$$

where $\mathbf{h}_t{}^T \mathbf{s}_{t'}$ denotes the conventional dot product energy function and η_t is defined as

$$\begin{aligned} \eta_t &= \gamma \, ||\mu(x_t)(1+\beta) \otimes \overline{VAD}(x_t)||_2^2, \\ \beta &= \tanh(\mathbf{W}^b \mathbf{x_{t-1}}), \end{aligned} \tag{3.3}$$

where $\otimes$ denotes element-wise multiplication, $||.||_k$ denotes l_k norm, $\mathbf{W}^b \in \mathbb{R}^{3\times m}$ denotes a model parameter, $\beta \in \mathbb{R}^3$ denotes a scaling factor in V, A and D dimensions in the $[-1, 1]$ interval to scale the normalized VAD values of the current input word, $\gamma \in \mathbb{R}_+$ denotes the affective attention coefficient controlling the magnitude of affect bias towards affect-rich words in the input sentence, and $\mu(x_t) \in \mathbb{R}$ in the $[0, 1]$ interval denotes a measure of term importance of x_t (see the following paragraph).

Term Importance

The introduction of term importance $\mu(x_t)$ as weights in computing affective attention is inspired by the sentence embedding work [155], where a simple weighted sum of word embedding algorithm with weights being smoothed inverse term frequency can achieve good performance in textual similarity tasks. Term frequency has been widely adopted in information retrieval (IR) to compute the importance of a word. In our model, we propose three approaches, namely "uniform importance" (ui),

"global importance" (gi), and "local importance" (li) to compute $\mu(x_t)$:

$$\mu(x_t) = \begin{cases} 1 & \text{ui} \\ a/(a + p(x_t)) & \text{gi} \\ \frac{\log(1/(p(x_t)+\epsilon))}{\sum_{t=1}^{t=T} \log(1/(p(x_t)+\epsilon))} & \text{li} \end{cases}, \tag{3.4}$$

where $p(x_t)$ denotes the term frequency of x_t in the training corpus, a denotes a smoothing constant that is usually set to 10^{-3} as suggested by Arora et al. [155], and ϵ denotes another small smoothing constant with value 10^{-8}. We take the log function in $\mu_{li}(x_t)$ to prevent rare words from dominating the importance.

Modeling Negators and Intensifiers

The introduction of β in Equation 3.3 is to model the affect changes caused by negators and intensifiers. Generally, negators make positive words negative but with much lower intensity, and make negative words less negative [156]. Thus, β is expected to be negative for negators because negators tend to reduce the affect intensity of the following word (e.g., "not bad"). Intensifiers usually adjust the intensities of positive words and negative words but do not flip their polarities [157]. As a result, β for extreme intensifiers (e.g., "extremely") is expected to be larger than β for less extreme intensifiers (e.g., "very"). To specifically consider these phenomena, β is modeled to be a nonlinear transformation through the word vector of x_{t-1}. This idea is inspired by the observation that common negators and intensifiers share some common underlying properties in their word vector representations. Figure 3.3 shows that several common negators and intensifiers tend to cluster together in the 2D plot of their GloVe embeddings [158] after applying Principle Component Analysis (PCA).

Note that our affective attention only considers unigram negators and intensifiers, which are empirically found as the majority of all negators and intensifiers. For example, the statistics based on our training set indicate that the unigram intensifier "very" occurs 364,913 times; in comparison, the composite intensifier "not very" only occurs 2,838 times.

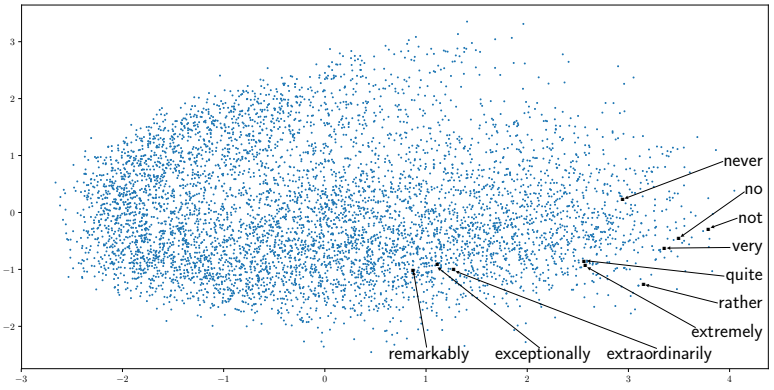


FIGURE 3.3: 2D plot of the most frequent 30,000 words in our training dataset in GloVe embedding after PCA. Selected common negators and intensifiers are annotated in text.

3.3.3 Affective Objective Function

The conventional objective function of seq2seq is to maximize the probability of target response Y given input sequence X, measured by cross-entropy loss. To encourage the generation of affect-rich words, we incorporate the VAD embedding of words into cross-entropy loss as follows:

$$\Psi_{t'} = -|V| \frac{1 + \delta || \overline{VAD}(y_{t'}) ||_2}{\sum_{\hat{y}_{t'} \in V} (1 + \delta || \overline{VAD}(\hat{y}_{t'}) ||_2)} \log(p(y_{t'})), \tag{3.5}$$

where $t^{'} = 1, 2, ..., T^{'}$, $\Psi_{t'}$ denotes the affective loss at decoding time step $t^{'}$, $y_{t'}$ denotes the target token at decoding time step $t^{'}$, V denotes the dataset vocabulary, and δ denotes a hyper-parameter named affective loss coefficient, which regulates the contribution of VAD embedding.

Our proposed affective loss is essentially a weighted cross-entropy loss. The weight of each word is constant and positively correlated with the VAD strength of the word in l_2 norm. The weight normalization is applied to ensure that our introduced weights do not alter the overall learning rate during optimization. Intuitively, our affective loss encourages affect-rich words to obtain higher output probability, which effectively introduces a probability bias into the decoder language model towards affect-rich words. This bias is controlled by our affective loss coefficient δ. When $\delta = 0$, our affective loss falls back to the conventional unweighted cross-entropy loss.

It is worth noting that our weighted cross-entropy loss incorporating external word knowledge, i.e., VAD in our case, is simple but effective in controlling the response style. Our loss function has many other potential application areas, such as controlled neural text generation.

3.4 Experiments

In this section, we present the datasets, evaluation metrics, baselines, model settings, experimental results and analysis.

3.4.1 Datasets and Evaluation Metrics

We use the OpenSubtitles dataset [159], a collection of movie subtitles, as our training dataset due to its large size. We use the relatively less noisy Cornell Movie Dialog Corpus dataset [160] as our validation dataset for more reliable tuning. We use the DailyDialog dataset [1], a collection of clean human written conversations, as our testing dataset to examine model generalizations in different corpus domains. All these datasets are in English.

The sentences in OpenSubtitles are not segmented. Hence, to extract {message, response} pairs, we follow a simple rule that the input sentence ends with a question mark and the time interval between the pair of input and output sentences is less than 20 seconds. In addition, sound sequences such as "*BANG*" are removed. These pairs are then expanded (e.g., isn't $\rightarrow$ is not) and tokenized. Special symbols and numbers were removed. Finally, the pairs with either input or output sentences longer than 20 words are removed. The validation and testing datasets are preprocessed by word expansion, tokenization, and removal of special symbols and numbers. Since we are modeling single-turn conversations, only the first two utterances from each dialogue session in the testing dataset are extracted because using utterances in the middle would require context to respond.

After data preprocessing, we randomly select 5 million pairs from OpenSubtitles as the training dataset with a vocabulary comprising the most 30,000 frequent words, covering 98.89% of all words. We randomly sample 100K pairs from Cornell Movie Dialog Corpus for validation and 10K pairs from DailyDialog for testing.

We adopt perplexity as the metric to measure the language fluency of a conversational model, as it is the only well-established automatic evaluation method in conversation modeling. Other metrics such as BLEU [161] do not correlate well with human judgments [162]. A model with lower perplexity indicates that it is more confident about the generated responses. Note that a model with low perplexity is not guaranteed to be a good conversational model because it may achieve so by always generating generic responses.

To qualitatively examine model performance, we conduct widely adopted human evaluations. We randomly sample 100 input sentences from the testing dataset, and each comparison model produces 100 responses. The responses are randomly ordered during evaluation. For each response, five human annotators are asked to evaluate the following two aspects:

- **+2**: (content) The response has correct grammar and is relevant and natural / (emotion) The response has adequate and appropriate emotions conveyed.
- **+1**: (content) The response has correct grammar but is too universal / (emotion) The response has inadequate but appropriate emotions conveyed.
- **0**: (content) The response has either grammar errors or is completely irrelevant / (emotion) The response has either no or inappropriate emotions conveyed.

3.4.2 Baselines and Model Settings

We first compare our approach with the following baselines to examine the performance of our proposed affective attention and affective objective function on model perplexity and human evaluations. We then compare our model with a state-of-the-art model in a preference test in Section 3.4.4.

S2S: The standard Seq2Seq model with attention.

S2S-UI, **S2S-GI**, **S2S-LI**: The standard Seq2Seq model with our proposed affective attention using μ_{ui}, μ_{gi} and μ_{li} (see Equation 3.4), respectively.

S2S-AO: The standard Seq2Seq model with attention and our proposed affective objective function (see Equation 3.5).

AR-S2S: our best model, which incorporates both μ_{li} and affective objective function.

All models have a word embedding of size 1027 (1024 + 3) and a hidden size of 1024. Both encoder and decoder have two layers of LSTM. All models implement affective embedding. Parameters λ, δ and a are set to 0.1, 0.15 and 10^{-3}, respectively. Parameter γ for S2S-UI, S2S-GI, and S2S-LI are set to 0.5, 1, and 5, respectively. We use beam search for decoding, where the beam size is set to 20. Note that all models implement the maximum mutual information (MMI) objective function [97] during inference to alleviate the problem of generic responses (e.g., "I don't know"). For all models, a simple re-rank operation is applied during inference to rank the generated responses $\hat{Y}$ based on their affective strength computed as $\frac{1}{|\hat{Y}|}\sum_{y\in\hat{Y}}||\overline{VAD}(y)||_2$. All model weights are initialized with a uniform distribution in the $[-0.08, 0.08]$ interval, using the same seed. We train all models using Adam [163] for 5 epochs with a batch size of 64 and a learning rate of 0.0001 throughout the training process.

3.4.3 Comparison with Baselines

Table 3.2 presents the results on model test perplexity. To analyze model generalization in different domains, we additionally report test perplexity on the in-domain test dataset, which is created using 10K test pairs from the OpenSubtitles dataset. All models have comparable perplexity on both in-domain and out-domain test datasets, empirically showing that our proposed methods do not cause performance degradation in language fluency. One note is that the out-domain test perplexity for all models is much worse than in-domain perplexity. One possible reason is that our testing dataset is different from the training dataset in terms of both vocabulary and linguistic distributions (the former was created from daily conversations, whereas the latter was created from movie subtitles). As a result, the models may not generalize well.

Tables 3.3 and 3.4 present the evaluation results by five human annotators on the content quality and emotion quality, respectively. The values in brackets denote relative performance improvement in percentage. The Fleiss' kappa [4] for measuring inter-rater agreement is included as well. All models have "moderate agreement" or "substantial agreement". For content quality, all models except

TABLE 3.2: Model test perplexity. Symbol † indicates in-domain perplexity obtained on 10K test pairs from the OpenSubtitles dataset. Symbol ‡ indicates out-domain perplexity obtained on 10K test pairs from the DailyDialog dataset.

Model	**#Params**	**PPL†**	**PPL‡**
S2S	99M	42.5	124.3
S2S-UI	99M	40.4	116.4
S2S-GI	99M	40.7	120.3
S2S-LI	99M	40.4	117.0
S2S-AO	99M	40.2	115.7
AR-S2S	99M	**39.8**	**113.7**

TABLE 3.3: Human evaluations on content quality.

Model (%)	**+2**	**+1**	**0**	**Score**	**Kappa**
S2S	22.4	47.0	30.6	0.918	0.544
S2S-UI	**30.0**	48.6	21.4	**1.086 (+18.3%)**	0.458
S2S-GI	28.6	46.6	24.8	1.038 (+13.1%)	0.413
S2S-LI	29.4	47.2	23.4	1.060 (+15.5%)	0.525
S2S-AO	25.0	46.0	29.0	0.960 (+4.3%)	0.482
AR-S2S	29.6	44.8	25.6	1.040 (+13.3%)	0.487

TABLE 3.4: Human evaluations on emotion quality.

Model (%)	**+2**	**+1**	**0**	**Score**	**Kappa**
S2S	19.0	33.2	47.8	0.712	0.613
S2S-UI	23.6	36.0	40.4	0.832 (+16.9%)	0.483
S2S-GI	26.0	34.2	39.8	0.862 (+21.1%)	0.652
S2S-LI	24.6	36.4	39.0	0.856 (+20.2%)	0.706
S2S-AO	22.6	37.6	39.8	0.828 (+16.3%)	0.602
AR-S2S	**26.8**	37.2	36.0	**0.908 (+27.5%)**	0.625

S2S-AO have noticeably more +2 ratings than S2S. For emotion quality, it is clear that all of our proposed affective models have substantial improvement over S2S. Among the three affective attention mechanisms, S2S-LI achieves the best overall performance. Note that the improvement gained by affective attention and affective objective function are partially orthogonal. One explanation is that by actively paying attention to affect-rich words in the input sentence, our model is able to produce more accurate affect-rich words during decoding. Therefore, combining both techniques (AR-S2S) results in maximum improvement in emotion quality. Table 3.5 presents some sample test responses.

TABLE 3.5: Sample responses. Text in bold denote affect-rich words.

Message	Model	Response
Mommy, can i stay up until eleven of the clock?	S2S	Of course you can stay up late.
	AR-S2S	Of course you can, **sweetheart**.
You are home late today, david. How was school?	S2S	It was fine.
	AR-S2S	**Great fun** today.

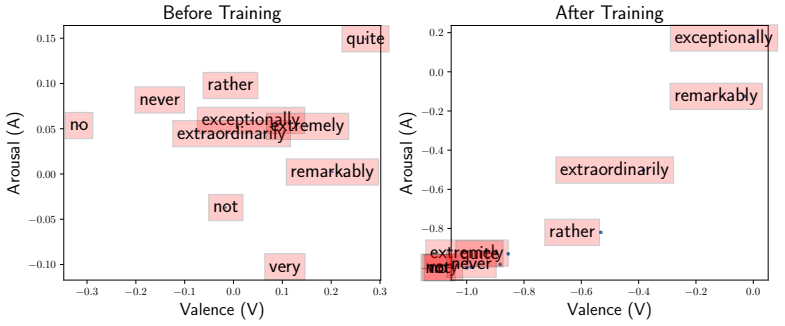


FIGURE 3.4: Learned parameter β (see Equation 3.3) in Valence (V) and Arousal (A) dimensions for several common negators and intensifiers. Left sub-figure: before AR-S2S is trained. Right sub-figure: after AR-S2S is trained.

Analysis of Affective Attention

To examine our hypothesis that our affective attention mechanism can correctly capture the effect of negators and intensifiers, we plot the learned parameter β (see Equation 3.3) in the Valence and Arousal dimensions in Figure 3.4. It is obvious that our model successfully learned to make β negative for negators. In addition, several extreme intensifiers such as "exceptionally" and "remarkably" have higher β than less extreme intensifiers such as "very" and "quite", which is consistent with our hypothesis. One note is that our model does not learn well for some intensifiers such as "extremely", whose β is comparable to less extreme intensifiers such as "very". This result is not surprising because the impact of intensifiers are difficult to be completely captured as they tend to vary depending on the words that follow them [156].

Figure 3.5 shows the attention strength over a sample input sentence in the testing dataset. As expected, our proposed affective attention models place extra attention on affect-rich words, i.e., "good" in this case. In addition, S2S-UI and S2S-LI have larger strengths than S2S-GI. This result is aligned with our model's assumption because different "term importance" have different impact on the attention

FIGURE 3.5: Learned attention on a sample input sentence from the testing dataset. From top to bottom, the models are S2S, S2S-UI, S2S-GI and S2S-LI, respectively. Darker colors indicate larger strength.

TABLE 3.6: Number of distinct affect-rich words.

	Threshold for l_2 Norm of VAD		
Model	3	2	1
S2S	25	104	190
S2S-AO ($\delta = 0.5$)	36	138	219
S2S-AO ($\delta = 1$)	50	154	234
S2S-AO ($\delta = 2$)	69	177	256

strengths and the word "good" here is quite common ($p(\text{"}good\text{"}) = 0.00143$), which leads to the lower strength in S2S-GI.

Analysis of Affective Objective Function

We analyze the capability of our proposed affective objective function in producing affect-rich words. Table 3.6 presents the number of distinct affect-rich words in randomly selected 1K test responses produced by different models. Affect-rich words are defined as words with VAD strength in l_2 norm exceeding the given threshold. It is clear that all S2S-AO models can produce more affect-rich words than S2S. In addition, the number of affect-rich words for every threshold increases steadily as the affective objective coefficient δ increases, showing a good controllability of our model via δ.

3.4.4 Preference Test

We conduct human preference test to compare our **AR-S2S** with the state-of-the-art baseline **S2S-Asghar**, the best model proposed in [154]. To the best of our knowledge, S2S-Asghar is the only model in the neural conversational model literature that aims to produce affect-rich responses in an end-to-end manner (i.e., without explicit user-specified emotions). We also include **S2S** for comparison.

TABLE 3.7: Model test perplexity. Symbol † indicates in-domain perplexity obtained on 10K test pairs from the OpenSubtitles dataset. Symbol ‡ indicates out-domain perplexity obtained on 10K test pairs from the DailyDialog dataset.

Model	**#Params**	**PPL†**	**PPL‡**
S2S	66M	41.2	130.6
S2S-Asghar	66M	46.4	137.2
AR-S2S	66M	**40.3**	**121.0**

TABLE 3.8: Number of distinct affect-rich words.

	Threshold for l_2 Norm of VAD		
Model	3	2	1
S2S	21	83	157
S2S-Asghar	31	120	217
AR-S2S	52	173	319

To make comparisons fair, we follow the specifications of S2S-Asghar and keep the number of parameters in all models comparable by reducing the size of our model. We use a smaller training dataset with 3 million random pairs and a vocabulary of size 20,000 due to the reduced model size. Note that our training dataset is still significantly larger than the original dataset used in [154], which comprises only 300K pairs and a vocabulary size of 12,000. All models have a word embedding of size 1027, a single-layer LSTM encoder, and a single-layer LSTM decoder. All training settings remain the same as in Section 3.4.2 except that S2S-Asghar is trained for 4 epochs with conventional cross-entropy loss and 1 more epoch with their proposed objective function, which includes a term to maximize affective content.

For human evaluation, we follow the same procedures as adopted in Section 3.4.1 except that five human annotators were asked to choose their preferred responses based on content quality and emotion quality, respectively, instead of annotating +2, +1, and 0. Ties are allowed.

Table 3.7 presents the results on model test perplexity. Our model achieves slightly better perplexity than S2S-Asghar in both in-domain and out-domain tests, suggesting that our model is more fluent than S2S-Asghar.

Table 3.8 shows the number distinct of affect-rich words in randomly selected 1K responses produced by S2S, S2S-Asghar, and our model. It is clear that our model produces significantly more affect-rich words than both S2S-Asghar and S2S.

TABLE 3.9: Human preference test.

Model (%)	Content	Emotion	Kappa
S2S	64	26	0.522/0.749
S2S-Asghar	66 (+3.1%)	32 (+23.1%)	0.554/0.612
AR-S2S	**80 (+25.0%)**	**49 (+88.5%)**	0.619/0.704

TABLE 3.10: Sample responses. Text in bold are affect-rich words.

Message	Model	Response
Do you like singing?	S2S	Yes, i do.
	S2S-Asghar	I do not know.
	AR-S2S	I **love** music.
I'm pretty sure that jim will turn out to be a good lawyer.	S2S	He will turn out to be a good lawyer.
	S2S-Asghar	I'm sure he will.
	AR-S2S	The **best** lawyer in the world.

Table 3.9 shows the result of human evaluation. The Fleiss' kappa scores for content/emotion qualities are included in the last column. All models have "moderate agreement" or "substantial agreement". For content preference, our model scores relatively 21% higher than S2S-Asghar. For emotion preference, our model scores relatively 50% higher than S2S-Asghar. These findings show that our model is capable of producing better responses that are not only more appropriate in syntax and content, but also significantly more affect-rich than the state-of-the-art model.

3.4.5 Sensitivity Analysis

We examine the impact of affect embedding strength λ, affective attention hyper-parameter γ, and affective loss hyper-parameter δ on model perplexity and the number of affect-rich words produced.

Due to the large number of experiments required for this analysis, we conduct the sensitivity analysis using 1 million pairs and a vocabulary of size 20,000. All training settings remain the same as in Section 3.4.2 except that the number of LSTM layers is 1, the hidden layer size is 512, and the embedding layer size is 303.

Figure 3.6 shows the plots of model test perplexity versus λ, γ, and δ. Our model is fairly robust to a wide range of λ, γ and δ, regardless of the type of term importance. It is worth noting that the generated responses tend to become shorter with $\gamma \in [20, \infty]$, which may be caused by excessive attention placed on affect-rich words during decoding. Another interesting finding is that our affective objective

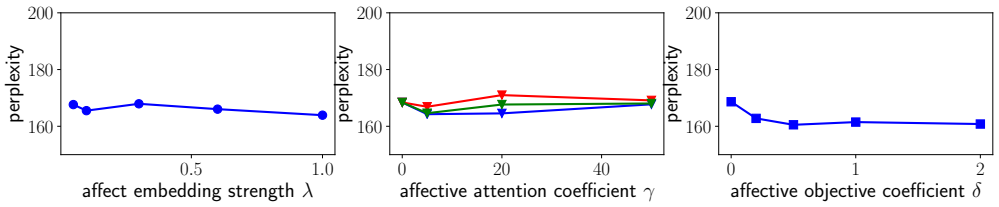


FIGURE 3.6: Sensitivity analysis for affect embedding strength λ, affective attention coefficient γ, and affective objective coefficient δ on model perplexity. The blue, red and green curves (*best viewed in color*) in the middle sub-figure denote μ_{ui}, μ_{gi} and μ_{li} (see Equation 3.4), respectively.

function slightly improves test perplexity. One possible explanation is that affect-rich words are less common than generic words in our training corpus. As a result, our weighted cross-entropy loss placing extra weights on them improves the overall prediction performance.

Figure 3.7 shows the plots of the number of distinct affect-rich words in randomly selected 1K test responses versus λ, γ and δ. The number of distinct words increases slightly when λ increases from 0 to 0.3, and then gradually decreases and stabilizes as λ increases from 0.3 to 1. For γ in all three term importance, there is an initial boost in the number of distinct words when γ is small, i.e., $\gamma \in [0, 5]$. However, as γ further increases, the number of distinct words gradually decreases, which may be caused by the limited word choices during decoding due to excessive attention on affect-rich words. Among the three proposed term importance, local importance (μ_{li}) is slightly more robust against γ than the other two methods. Finally, the number of distinct words consistently increases with δ, which is similar to our findings from Table 3.6. Note that the numbers in this sensitivity analysis are much smaller than Table 3.6, which can be attributed to smaller models and fewer training examples.

3.5 Summary

In this chapter, we propose AR-S2S, an end-to-end open-domain neural conversational agent that can produce affect-rich/emotional responses without performance degradation in language fluency. AR-S2S leverages external word-VAD knowledge to encode affect information. In addition, AR-S2S captures user emotions

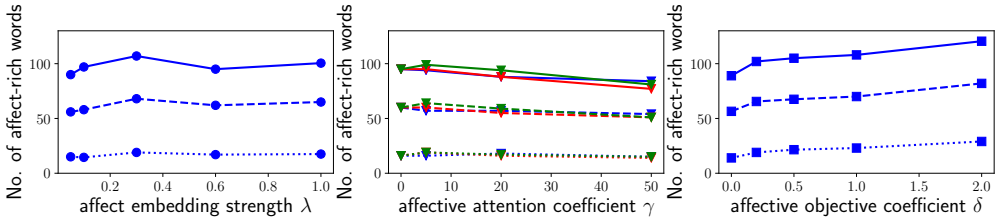


FIGURE 3.7: Sensitivity analysis for affect embedding strength λ, affective attention coefficient γ, and affective objective coefficient δ on the number of distinct affect-rich words in randomly selected 1K test responses. The solid, dashed and dotted curves correspond to l_2 norm threshold of 1, 2 and 3, respectively. The blue, red and green curves (*best viewed in color*) in the middle sub-figure denote μ_{ui}, μ_{gi} and μ_{li} (see Equation 3.4), respectively.

by paying extra attention to affect-rich words in input sentences and considering the effect caused by negators and intensifiers. Lastly, AR-S2S is trained with an affect-incorporated weighted cross-entropy loss to encourage the generation of affect-rich words. Empirical studies on both model perplexity and human evaluations show that AR-S2S outperforms the state-of-the-art model in producing natural and affect-rich responses. Our study suggests that incorporating emotion into open-domain CAs improves response quality and human ratings.

It is worth noting that many other languages such as Spanish, Dutch, Finish, etc., also have lemma-VAD pairs corpus, although in smaller sizes. Hence, our proposed AR-S2S has great potential to be directly applied to other languages.

Chapter 4

Commonsense-Aware Conversational Emotion Recognition

4.1 Overview

In the last chapter, we studied the problem of endowing conversational agents (CA) with emotions for response generation. However, a humanized CA should be able to not only produce appropriate emotional responses but also recognize human emotions at each turn during conversations. In this chapter, we investigate the problem of recognizing human emotions in natural conversations[1]. Specifically, we address the task of recognizing emotions (e.g., happy, sad, and angry, etc.) in multi-turn textual conversations, where the emotion of an utterance is to be detected in the conversational context. Being able to automatically and effectively detect emotions in conversations enables a wide range of applications ranging from opinion mining in social media platforms [119] to building emotion-aware CAs [21].

However, building machines that can analyze emotions in human conversations is challenging, partly because humans often express emotions by relying on the context and commonsense knowledge, which are difficult to be captured by machines.

[1]This chapter is published as *Knowledge-Enriched Transformer for Emotion Detection in Textual Conversations*, Proceedings of EMNLP-IJCNLP 2019 [50].

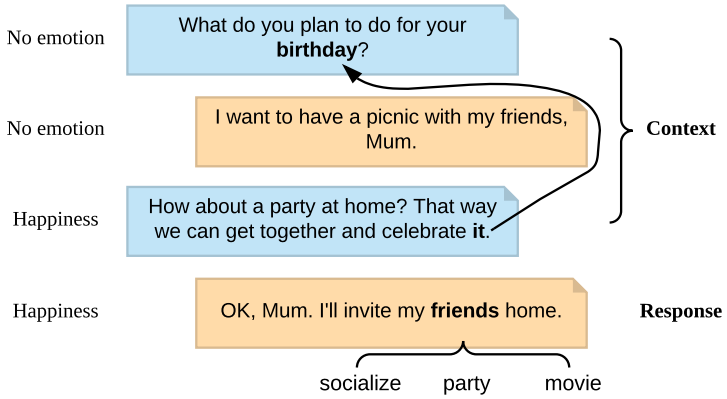


FIGURE 4.1: An example conversation with annotated labels from the DailyDialog dataset [1]. By referring to the context, "it" in the third utterance is linked to "birthday" in the first utterance. By leveraging an external knowledge base, the meaning of "friends" in the fourth utterance is enriched by associated knowledge entities, namely "socialize", "party", and "movie". Thus, the implicit "happiness" emotion in the fourth utterance can be inferred more easily via its enriched meaning.

Figure 4.1 shows an example conversation demonstrating the importance of context and commonsense knowledge in understanding conversations and detecting implicit emotions.

There are several recent studies that model contextual information to detect emotions in conversations. Poria et al. [121] and Majumder et al. [164] leveraged RNNs (see Section 2.1.1) to model the contextual utterances in sequence, where each utterance is represented by a feature vector extracted by a pretrained CNN. Similarly, Hazarika et al. [116, 117] proposed to use extracted CNN features in memory networks to model contextual utterances. However, these methods require separate feature extraction and tuning, which may not be ideal for real-time applications. In addition, to the best of our knowledge, no attempts have been made in the literature to incorporate commonsense knowledge from external knowledge bases to detect emotions in textual conversations. Commonsense knowledge is fundamental to understanding conversations and generating appropriate responses [22].

To this end, we propose a Knowledge-Enriched Transformer (KET) to effectively incorporate contextual information and external knowledge bases to address the aforementioned challenges. The Transformer [49] (see Section 2.1.3) has been shown to be a powerful representation learning model in many NLP tasks such as machine translation [49] and language understanding [165]. The self-attention [166]

and cross-attention [65] modules in the Transformer capture the intra-sentence and inter-sentence correlations, respectively. The shorter path of information flow in these two modules compared to gated RNNs and CNNs allows KET to model contextual information more efficiently. In addition, we propose a hierarchical self-attention mechanism allowing KET to model the hierarchical structure of conversations. Our model separates context and response into encoder and decoder, respectively, which is different from other Transformer-based models, e.g., BERT [165], which directly concatenate context and response and train language models using only the encoder part.

Moreover, to exploit commonsense knowledge, we leverage external knowledge bases to facilitate the understanding of each word in the utterances by referring to related knowledge entities. The referring process is dynamic and balances between relatedness and affectiveness of the retrieved knowledge entities using a context-aware affective graph attention mechanism.

In summary, our main contributions in this chapter are as follows:

- For the first time, we apply the Transformer to analyze conversations and detect emotions. Our hierarchical self-attention and cross-attention modules allow our model to exploit contextual information more efficiently than existing gated RNNs and CNNs.
- We derive dynamic, context-aware, and emotion-related commonsense knowledge from external knowledge bases and emotion lexicons to facilitate emotion detection in conversations.
- We conduct extensive experiments and demonstrate that both contextual information and commonsense knowledge are beneficial to emotion detection performance. In addition, our proposed KET model outperforms the state-of-the-art models on most of the tested datasets across different domains.

4.2 Related Work

Emotion Detection in Conversations: Early studies on emotion detection in conversations focus on call center dialogs using lexicon-based methods and audio features [113, 114]. Devillers et al. [115] annotated and detected emotions in

call center dialogs using unigram topic modeling. Devillers and Vidrascu [114] investigated emotion detection with lexical and paralinguistic cues on dialogs in a medical emergency call center. In recent years, there is an emerging research trend on emotion detection in conversational videos and multi-turn Tweets using deep learning methods [116–120, 167]. Poria et al. [121] proposed an LSTM [57] based model to capture contextual information for sentiment analysis in user-generated videos. Hazarika et al. [116] proposed a memory network to capture context and inter-speaker dependencies in conversational videos. Later on, Hazarika et al. [117] extended further to model self- and inter-speaker emotional influence. Majumder et al. [164] proposed the DialogueRNN model that uses three GRU [58] to model the speaker, the context from the preceding utterances, and the emotions of the preceding utterances, respectively. DialogueRNN achieved state-of-the-art performance on several conversational video datasets.

Knowledge Base in Conversations: Recently, there is a growing number of studies on incorporating knowledge base in generative conversation systems, such as open-domain dialogue systems [36, 48, 111, 122, 168–172], task-oriented dialogue systems [173–175] and question answering systems [176–179]. Zhou et al. [22] adopted structured knowledge graphs to enrich the interpretation of input sentences and help generate knowledge-aware responses using graph attentions. The graph attention in the knowledge interpreter [22] is static and only related to the recognized entity of interest. In contrast, our graph attention mechanism is dynamic and selects context-aware knowledge entities that balance between relatedness and affectiveness.

Emotion Detection in Text: There is a trend moving from traditional machine learning methods [180–182] to deep learning methods [183, 184] for emotion detection in text. Khanpour and Caragea [185] investigated the emotion detection from health-related posts in online health communities using both deep learning features and lexicon-based features.

Incorporating Knowledge in Sentiment Analysis: Traditional lexicon-based methods detect emotions or sentiments from a piece of text based on the emotions or sentiments of words or phrases that compose it [186–188]. Some commonly used lexicons include WordNet-Affect [189], SentiWordNet [190], and NRC-VAD [191]. Few studies investigated the usage of knowledge bases in deep learning methods.

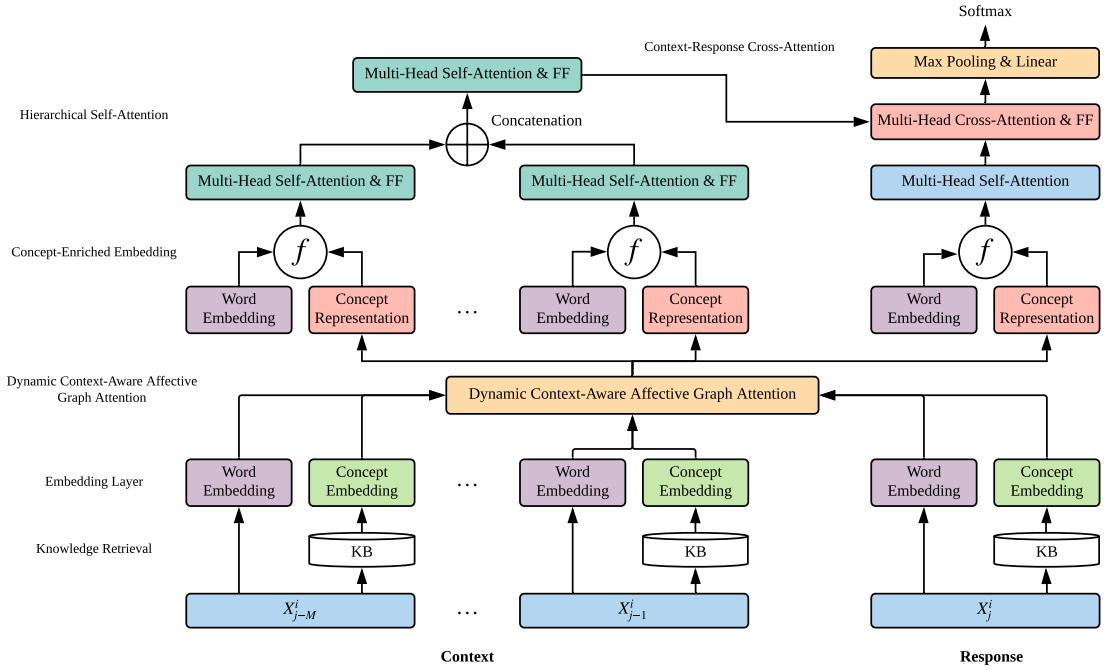


FIGURE 4.2: Overall architecture of our proposed KET model. The positional encoding, residual connection, and layer normalization are omitted in the illustration for brevity.

Kumar et al. [192] proposed to use knowledge from WordNet [193] to enrich the text representations produced by LSTM and obtained improved performance.

Transformer: The Transformer (see Section 2.1.3) has been applied to many NLP tasks due to its rich representation and fast computation, e.g., document machine translation [194], response matching in dialogue system [93], language modeling [68] and understanding [76]. The current state-of-the-art models on natural language understanding tasks, e.g., GLUE [195], are dominated by pretrained Transformer models. Recently, incorporating knowledge into pretrained Transformer models has become an active research topic [196–198]. Different from their approaches, our KET is not a pretrained Transformer model and is specifically designed for modeling conversations.

4.3 Knowledge-Enriched Transformer (KET)

In this section, we present the task definition and our proposed KET model.

4.3.1 Task Definition

Let $\{X_j^i, Y_j^i\}, i = 1, ...N, j = 1, ...N_i$ be a collection of $\{utterance, label\}$ pairs in a given dialogue dataset, where N denotes the number of conversations and N_i denotes the number of utterances in the ith conversation. The objective of the task is to maximize the following function:

$$\Phi = \prod_{i=1}^{N} \prod_{j=1}^{N_i} p(Y_j^i | X_j^i, X_{j-1}^i, ..., X_1^i; \theta), \tag{4.1}$$

where $X_{j-1}^i, ..., X_1^i$ denote contextual utterances and θ denotes the model parameters we want to optimize.

We limit the number of contextual utterances to M. Discarding early contextual utterances may cause information loss, but this loss is negligible because they only contribute the least amount of information [199]. This phenomenon can be further observed in our model analysis regarding context length (see Section 4.4.4). Similar to [121], we clip and pad each utterance X_j^i to a fixed m number of tokens. The overall architecture of our KET model is illustrated in Figure 4.2.

4.3.2 Knowledge Retrieval

We use a commonsense knowledge base ConceptNet [200] and an emotion lexicon NRC_VAD [191] as knowledge sources in our model.

ConceptNet is a large-scale multilingual semantic graph that describes general human knowledge in natural language. The nodes in ConceptNet are concepts and the edges are relations. Each $\langle$*concept1*, *relation*, *concept2*$\rangle$ triplet is an assertion. Each assertion is associated with a confidence score. An example assertion is $\langle$*friends*, *CausesDesire*, *socialize*$\rangle$ with confidence score of 3.46. Usually assertion confidence scores are in the $[1, 10]$ interval. Currently, for English, ConceptNet comprises 5.9M assertions, 3.1M concepts and 38 relations.

NRC_VAD is a list of English words and their VAD scores, i.e., valence (negative-positive), arousal (calm-excited), and dominance (submissive-dominant) scores in the $[0, 1]$ interval. The VAD measure of emotion is culture-independent and widely

adopted in Psychology [151]. Currently, NRC_VAD comprises around 20K words, which is a newer and larger lexicon than the one used in Chapter 3.

In general, for each non-stopword token t in X_j^i, we retrieve a connected knowledge graph $g(t)$ comprising its immediate neighbors from ConceptNet. For each $g(t)$, we remove concepts that are stopwords or not in our vocabulary. We further remove concepts with confidence scores less than 1 to reduce annotation noises. For each concept, we retrieve its VAD values from NRC_VAD. The final knowledge representation for each token t is a list of tuples: $(c_1, s_1, VAD(c_1))$, $(c_2, s_2, VAD(c_2))$, ..., $(c_{|g(t)|}, s_{|g(t)|}, VAD(c_{|g(t)|}))$, where $c_k \in g(t)$ denotes the kth connected concept, s_k denotes the associated confidence score, and $VAD(c_k)$ denotes the VAD values of c_k. The treatment for tokens that are not associated with any concept and concepts that are not included in NRC_VAD are discussed in Section 4.3.4. We leave the treatment on relations as future work.

4.3.3 Embedding Layer

We use a word embedding layer to convert each token t in X^i into a vector representation $\mathbf{t} \in \mathbb{R}^d$, where d denotes the size of word embedding. To encode positional information, the position encoding [49] is added as follows:

$$\mathbf{t} = Embed(t) + Pos(t). \tag{4.2}$$

Similarly, we use a concept embedding layer to convert each concept c into a vector representation $\mathbf{c} \in \mathbb{R}^d$ but without position encoding.

4.3.4 Dynamic Context-Aware Affective Graph Attention

To enrich word embedding with concept representations, we propose a dynamic context-aware affective graph attention mechanism to compute the concept representation for each token. Specifically, the concept representation $\mathbf{c}(t) \in \mathbb{R}^d$ for token t is computed as

$$\mathbf{c}(t) = \sum_{k=1}^{|g(t)|} \alpha_k * \mathbf{c_k}, \tag{4.3}$$

where $\mathbf{c_k} \in \mathbb{R}^d$ denotes the concept embedding of c_k and α_k denotes its attention weight. If $|g(t)| = 0$, we set $\mathbf{c}$(t) to the average of all concept embeddings. The attention α_k in Equation 4.3 is computed as

$$\alpha_k = softmax(w_k), \tag{4.4}$$

where w_k denotes the weight of c_k.

The derivation of w_k is crucial because it regulates the contribution of $\mathbf{c}_k$ towards enriching $\mathbf{t}$. A standard graph attention mechanism [201] computes w_k by feeding $\mathbf{t}$ and $\mathbf{c}_k$ into a single-layer feedforward neural network. However, not all related concepts are equal in detecting emotions given the conversational context. In our model, we make the assumption that important concepts are those that relate to the conversational context and have strong emotional intensity. To this end, we propose a context-aware affective graph attention mechanism by incorporating two factors when computing w_k, namely relatedness and affectiveness.

Relatedness:

Relatedness measures the strength of the relation between c_k and the conversational context. The relatedness factor in w_k is computed as

$$rel_k = min\text{-}max(s_k) * abs(\cos(\mathbf{CR}(X^i), \mathbf{c_k})), \tag{4.5}$$

where s_k is the confidence score introduced in Section 4.3.2, *min-max* denotes min-max scaling for each token t, *abs* denotes the absolute function, cos denotes the cosine similarity function, and $\mathbf{CR}(X^i) \in \mathbb{R}^d$ denotes the context representation of the ith conversation X^i. Here we compute $\mathbf{CR}(X^i)$ as the average of all sentence representations in X^i as follows:

$$\mathbf{CR}(X^i) = avg(\mathbf{SR}(X^i_{j-M}), ..., \mathbf{SR}(X^i_j)), \tag{4.6}$$

where $\mathbf{SR}(X^i_j) \in \mathbb{R}^d$ denotes the sentence representation of X^i_j. We compute $\mathbf{SR}(X^i_j)$ via hierarchical pooling [202] where n-gram ($n \leq 3$) representations in X^i_j are first computed by max-pooling and then all n-gram representations are averaged. The hierarchical pooling mechanism preserves word order information

to certain degree and has demonstrated superior performance than average pooling or max-pooling on sentiment analysis tasks [202].

Affectiveness:

Affectiveness measures the emotional intensity of c_k. The affectiveness factor in w_k is computed as

$$\mathit{aff}_k = \mathit{min\text{-}max}(||[V(c_k) - 1/2, A(c_k)/2]||_2), \tag{4.7}$$

where $||.||_k$ denotes l_k norm, $V(c_k) \in [0, 1]$ and $A(c_k) \in [0, 1]$ denote the valence and arousal values of $VAD(c_k)$, respectively. Intuitively, aff_k considers the deviations of valence from neutral and the level of arousal from calm. There is no established method in the literature to compute the emotional intensity based on VAD values, but empirically we found that our method correlates better with an emotion intensity lexicon comprising 6K English words [203] than other methods such as taking dominance (D) into consideration or taking l_1 norm. For concept c_k not in NRC_VAD, we set aff_k to the mid value of 0.5.

Combining both rel_k and aff_k, we define the weight w_k as follows:

$$w_k = \lambda_k * \mathit{rel}_k + (1 - \lambda_k) * \mathit{aff}_k, \tag{4.8}$$

where λ_k is a model parameter balancing the impacts of relatedness and affectiveness on computing concept representations. Parameter λ_k can be fixed as *a prior* or learned during training. The analysis of λ_k is discussed in Section 4.4.4.

Finally, the concept-enriched word representation $\hat{\mathbf{t}}$ can be obtained via a linear transformation:

$$\hat{\mathbf{t}} = \mathbf{W}[\mathbf{t}; \mathbf{c}(t)], \tag{4.9}$$

where $[;]$ denotes concatenation and $\mathbf{W} \in \mathbb{R}^{d \times 2d}$ denotes a model parameter. All m tokens in each X_j^i then form a concept-enriched utterance embedding $\hat{\mathbf{X}}_j^i \in \mathbb{R}^{m \times d}$.

4.3.5 Hierarchical Self-Attention

We propose a hierarchical self-attention mechanism to exploit the structural representation of conversations and learn a vector representation for the contextual utterances $X^i_{j-1}, ..., X^i_{j-M}$. Specifically, the hierarchical self-attention follows two steps: 1) each utterance representation is computed using an utterance-level self-attention layer, and 2) a context representation is computed from the M learned utterance representations using a context-level self-attention layer.

At step 1, for each utterance X^i_n, n=$j-1$, ..., $j-M$, its representation $\hat{\mathbf{X}}'^i_n \in \mathbb{R}^{m \times d}$ is learned as follows:

$$\hat{\mathbf{X}}'^i_n = FF(MultiHead(\hat{\mathbf{X}}^i_n, \hat{\mathbf{X}}^i_n, \hat{\mathbf{X}}^i_n)), \tag{4.10}$$

where FF denotes the feed-forward layer defined in Section 2.13 and $MultiHead$ denotes the multi-head attention layer defined in Section 2.12. The *MultiHead* layer enables our model to jointly attend to information from different representation subspaces [49]. Similar to Equation 2.11, both *MultiHead* and *FF* layers are followed by residual connection and layer normalization, which are omitted in Equation 4.10 for brevity.

At step 2, to effectively combine all utterance representations in the context, the context-level self-attention layer is proposed to hierarchically learn the context-level representation $\mathbf{C}^i \in \mathbb{R}^{M \times m \times d}$ as follows:

$$\mathbf{C}^i = FF(MultiHead(\hat{\mathbf{X}}^i, \hat{\mathbf{X}}^i, \hat{\mathbf{X}}^i)), \tag{4.11}$$

where $\hat{\mathbf{X}}^i$ denotes $[\hat{\mathbf{X}}'^i_{j-M}; ...; \hat{\mathbf{X}}'^i_{j-1}]$, which is the concatenation of all learned utterance representations in the context.

4.3.6 Context-Response Cross-Attention

Finally, a context-aware concept-enriched response representation $\mathbf{R}^i \in \mathbb{R}^{m \times d}$ for conversation X^i is learned by cross-attention [65], which selectively attends to the concept-enriched context representation as follows:

$$\mathbf{R}^i = FF(MultiHead(\hat{\mathbf{X}}'^i_j, \mathbf{C}^i, \mathbf{C}^i)), \tag{4.12}$$

TABLE 4.1: Dataset descriptions.

Dataset	Domain	#Conv.	#Utter.	#Classes
EC	Tweet	30160/2755/5509	90480/8265/16527	4
DailyDialog	Daily Chat	11118/1000/1000	87170/8069/7740	7
MELD	TV Show Scripts	1038/114/280	9989/1109/2610	7
EmoryNLP	TV Show Scripts	659/89/79	7551/954/984	7
IEMOCAP	Emotional Chat	100/20/31	4810/1000/1523	6

where the response utterance representation $\hat{\mathbf{X}}_j'^i \in \mathbb{R}^{m \times d}$ is obtained via the *MultiHead* layer:

$$\hat{\mathbf{X}}_j'^i = \mathit{MultiHead}(\hat{\mathbf{X}}_j^i, \hat{\mathbf{X}}_j^i, \hat{\mathbf{X}}_j^i), \tag{4.13}$$

The resulted representation $\mathbf{R}^i \in \mathbb{R}^{m \times d}$ is then fed into a max-pooling layer to learn discriminative features among the positions in the response and derive the final representation $\mathbf{O} \in \mathbb{R}^d$:

$$\mathbf{O} = \mathit{max\_pool}(\mathbf{R}^i). \tag{4.14}$$

The output probability p is then computed as

$$p = \mathit{softmax}(\mathbf{O}W_3 + b_3), \tag{4.15}$$

where $W_3 \in \mathbb{R}^{d \times q}$ and $b_3 \in \mathbb{R}^q$ denote model parameters, and q denotes the number of classes. The entire KET model is optimized in an end-to-end manner as defined in Equation 4.1.

4.4 Experiments

In this section, we present our datasets, evaluation metrics, baselines, model settings, experimental results and analysis.

4.4.1 Datasets and Evaluation Metrics

Following prior studies, we evaluate our model on an extensive set of five English emotion detection datasets of various sizes and domains. The statistics are reported in Table 4.1.

EC [119]: Three-turn Tweets. The emotion labels include happiness, sadness, anger, and other.

DailyDialog [1]: Human written daily chats. The emotion labels include neutral and Ekman's six basic emotions [204], namely happiness, surprise, sadness, anger, disgust, and fear.

MELD [205]: TV show scripts collected from *Friends*. The emotion labels are the same as the ones used in DailyDialog.

EmoryNLP [118]: TV show scripts collected from *Friends* as well. However, its size and annotations are different from MELD. The emotion labels include neutral, sad, mad, scared, powerful, peaceful, and joyful.

IEMOCAP [206]: Emotional chats. The emotion labels include neutral, happiness, sadness, anger, frustrated, and excited.

All datasets follow the default train/val/test splits. For IEMOCAP, whose validation split is not given, we randomly selected 20 conversations from the training data as the validation data.

In terms of the evaluation metric, for EC and DailyDialog, we follow [119] to use the micro-averaged F1 excluding the majority class (neutral), due to their extremely unbalanced labels (the percentage of the majority class in the test set is over 80%). For the rest relatively balanced datasets, we follow [164] to use the weighted macro-F1.

4.4.2 Baselines and Model Settings

For a comprehensive performance evaluation, we compare our model with the following baselines:

cLSTM: A contextual LSTM model. An utterance-level bidirectional LSTM is used to encode each utterance. A context-level unidirectional LSTM is used to encode the context.

CNN [207]: A single-layer CNN with strong empirical performance. This model is trained on the utterance-level without context.

TABLE 4.2: Hyper-parameter settings for KET. M: context length. m: number of tokens per utterance. d: word embedding size. p: hidden size in FF layer. h: number of heads.

Dataset	M	m	d	p	h
EC	2	30	200	100	4
DailyDialog	6	30	300	400	4
MELD	6	30	200	100	4
EmoryNLP	6	30	100	200	4
IEMOCAP	6	30	300	400	4

CNN+cLSTM [121]: An CNN is used to extract utterance features. An cLSTM is then applied to learn context representations.

BERT_BASE [165]: Base version of the state-of-the-art model for sentiment classification. We treat each utterance with its context as a single document. We limit the document length to the last 100 tokens to allow larger batch sizes. We do not experiment with the large version of BERT due to the memory constraint of our GPU.

DialogueRNN [164]: The state-of-the-art model for emotion detection in textual conversations. It models both context and speaker information. The CNN features used in DialogueRNN are extracted from the carefully tuned CNN model. For datasets without speaker information, i.e., EC and DailyDialog, we use two speakers only. For MELD and EmoryNLP, which have 260 and 255 speakers, respectively, we additionally experimented with clipping the number of speakers to the most frequent ones (6 main speakers + a universal speaker representing all other speakers) and reported the best results.

KET_SingleSelfAttn: We replace the hierarchical self-attention by a single self-attention layer to learn context representations. Contextual utterances are concatenated together prior to the single self-attention layer.

KET_StdAttn: We replace the dynamic context-aware affective graph attention by the standard graph attention [201].

We preprocessed all datasets by lower-casing and tokenization using Spacy[2]. We keep all tokens in the vocabulary[3]. We use the released code for BERT_BASE and

[2]https://spacy.io/

[3]We keep tokens with a minimum frequency of 2 for DailyDialog due to its large vocabulary size

TABLE 4.3: Performance comparisons on the five test sets. The bottom three rows are from our models. Best values are highlighted in bold.

Model	EC	DailyDialog	MELD	EmoryNLP	IEMOCAP
cLSTM	0.6913	0.4990	0.4972	0.2601	0.3484
CNN [207]	0.7056	0.4934	0.5586	0.3259	0.5218
CNN+cLSTM [121]	0.7262	0.5024	0.5687	0.3289	0.5587
BERT_BASE [165]	0.6946	0.5312	0.5621	0.3315	0.6119
DialogueRNN [164]	0.7405	0.5065	0.5627	0.3170	**0.6121**
KET_SingleSelfAttn	0.7285	0.5192	0.5624	0.3251	0.5810
KET_StdAttn	**0.7413**	0.5254	0.5682	0.3353	0.5861
KET	0.7348	**0.5337**	**0.5818**	**0.3439**	0.5956

DialogueRNN. For each dataset, all models are fine-tuned based on their performance on the validation set.

For our model in all datasets, we use Adam optimization [163] with a batch size of 64 and a learning rate of 0.0001 throughout the training process. We use GloVe embedding [158] for initialization in the word and concept embedding layers[4]. For the class weights in cross-entropy loss for each dataset, we set them as the ratio of the class distribution in the validation set to the class distribution in the training set. Thus, we can alleviate the problem of unbalanced dataset. The detailed hyper-parameter settings for KET are presented in Table 4.2.

4.4.3 Comparison with Baselines

We compare the performance of KET against that of the baseline models on the five afore-introduced datasets. The results are reported in Table 4.3. Note that our results for CNN, CNN+cLSTM, and DialogueRNN on EC, MELD, and IEMOCAP are slightly different from the reported results in [120, 164].

cLSTM performs reasonably well on short conversations (i.e., EC and DailyDialog), but the worst on long conversations (i.e., MELD, EmoryNLP, and IEMOCAP). One major reason is that learning long dependencies using gated RNNs may not be effective enough because the gradients are expected to propagate back through inevitably a huge number of utterances and tokens in sequence, which can easily lead to the vanishing gradient problem [56]. In contrast, when the utterance-level LSTM in cLSTM is replaced by features extracted by CNN, i.e., the CNN+cLSTM,

[4]We use GloVe embeddings from https://github.com/plasticityai/magnitude

the model performs significantly better than cLSTM on long conversations, which further validates that modeling long conversations using only RNN models may not be sufficient.

BERT_BASE achieves very competitive performance on all datasets except EC due to its strong representational power via bi-directional context modeling using the Transformer. Note that BERT_BASE has considerably more parameters than other baselines and our model (110M parameters for BERT_BASE versus 4M parameters for our model), which can be a disadvantage when deployed to devices with limited computing power and memory. The state-of-the-art DialogueRNN model performs the best overall among all baselines. In particular, DialogueRNN performs better than our model on IEMOCAP, which may be attributed to its detailed speaker information for modeling the emotion dynamics in each speaker as the conversation flows.

It is encouraging to see that our KET model outperforms the baselines on most of the datasets tested. This finding indicates that our model is robust across datasets with varying training sizes, context lengths and domains. Our KET variants KET_SingleSelfAttn and KET_StdAttn perform comparably with the best baselines on all datasets except IEMOCAP. However, both variants perform noticeably worse than KET on all datasets except EC, validating the importance of our proposed hierarchical self-attention and dynamic context-aware affective graph attention mechanism. One observation worth mentioning is that these two variants perform on a par with the KET model on EC. Possible explanations are that 1) the hierarchical self-attention may not be critical for modeling short conversations in EC, and 2) the informal linguistic styles of Tweets in EC, e.g., misspelled words and slangs, hinder the context representation learning in our graph attention mechanism.

4.4.4 Model Analysis

We analyze the impact of different settings on the validation performance of KET. All results in this section are averaged over 5 random seeds.

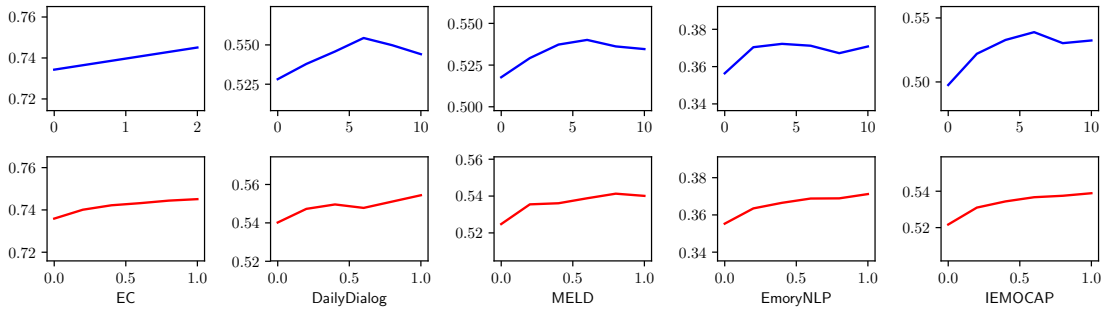


FIGURE 4.3: Validation performance by KET. Top: different context length (M). Bottom: different sizes of random fractions of ConceptNet.

Analysis of context length:

We vary the context length M and plot model performance in Figure 4.3 (top portion). Note that EC has only a maximum number of 2 contextual utterances. It is clear that incorporating context into KET improves performance on all datasets. However, adding more context is contributing to diminishing performance gain or even making a negative impact on some datasets. This phenomenon has been observed in a prior study [199]. One possible explanation is that incorporating long contextual information may introduce additional noises, e.g., polysemes expressing different meanings in different utterances of the same context. A more thorough investigation of this diminishing return phenomenon is a worthwhile direction in the future.

Analysis of the size of ConceptNet:

We vary the size of ConceptNet by randomly keeping only a fraction of the concepts in ConceptNet when training and evaluating our model. The results are illustrated in Figure 4.3 (bottom portion). Adding more concepts consistently improves model performance before reaching a plateau, validating the importance of commonsense knowledge in detecting emotions. We may expect the performance of our KET model to improve with the growing size of ConceptNet in the future.

TABLE 4.4: Analysis of the relatedness-affectiveness tradeoff on the validation sets. Each column corresponds to a fixed λ_k for all concepts (see Equation 4.8).

Dataset	**0**	**0.3**	**0.7**	**1**
EC	0.7345	0.7397	**0.7426**	0.7363
DailyDialog	0.5365	0.5432	**0.5451**	0.5383
MELD	0.5321	**0.5395**	0.5366	0.5306
EmoryNLP	0.3528	**0.3624**	0.3571	0.3488
IEMOCAP	0.5344	**0.5367**	0.5314	0.5251

TABLE 4.5: Ablation study for KET on the validation sets.

Dataset	KET	-context	-knowledge
EC	**0.7451**	0.7343	0.7359
DailyDialog	**0.5544**	0.5282	0.5402
MELD	**0.5401**	0.5177	0.5248
EmoryNLP	**0.3712**	0.3564	0.3553
IEMOCAP	**0.5389**	0.4976	0.5217

Analysis of the relatedness-affectiveness tradeoff:

We experiment with different values of $\lambda_k \in [0, 1]$ (see Equation 4.8) for all k and report the results in Table 4.4. It is clear that λ_k makes a noticeable impact on the model performance. Discarding relatedness or affectiveness completely will cause significant performance drop on all datasets, with one exception of IEMOCAP. One possible reason is that conversations in IEMOCAP are emotional dialogues, therefore, the affectiveness factor in our proposed graph attention mechanism can provide more discriminative power.

Ablation Study:

We conduct ablation study to investigate the contribution of context and knowledge as reported in Table 4.5. It is clear that both context and knowledge are essential to the strong performance of KET on all datasets. Note that removing context has a greater impact on long conversations than short conversations, which is expected because more contextual information is lost in long conversations.

4.4.5 Error Analysis

Despite the strong performance of our model, it still fails to detect certain emotions on certain datasets. We rank the F1 score of each emotion per dataset and investigate the emotions with the worst scores. We found that disgust and fear are generally difficult to detect and differentiate. For example, the F1 score of fear emotion in MELD is as low as 0.0667. One possible cause is that these two emotions are intrinsically similar. The VAD representation of both emotions has low valence, high arousal, and low dominance [151]. Another cause is the small amount of data available for these two emotions. How to differentiate intrinsically similar emotions and how to effectively detect emotions using limited data are two challenging directions in this field.

4.5 Summary

In this chapter, we present a knowledge-enriched transformer (KET) to detect emotions in textual conversations. Unlike the dominant approach of using pre-trained language models or RNNs and CNNs for text classification, we extend the Transformer and analyze conversations under the framework of machine translation. Our model learns structured conversation representations via hierarchical self-attention and dynamically refers to external, context-aware, and emotion-related knowledge entities from knowledge bases. Our experimental analysis demonstrates that both contextual information and commonsense knowledge are beneficial to model performance. The tradeoff between relatedness and affectiveness plays an important role as well. In addition, our model outperforms the state-of-the-art models on most of the tested datasets of varying sizes and domains.

Given that there are similar emotion lexicons to NRC_VAD in other languages and ConceptNet is a multilingual knowledge base, our model can be easily adapted to other languages. In addition, given that NRC_VAD is the only emotion-specific component, our model can be adapted as a generic model for conversation analysis.

Chapter 5

Commonsense-Aware Emotional Conversational Agents

5.1 Overview

In Chapter 3, we studied the problem of endowing response generation with emotion. In Chapter 4, we investigated the impact of commonsense knowledge on conversational emotion recognition and found that commonsense is beneficial to conversational emotion recognition. In this chapter, we take a step further and investigate the impact of combining commonsense and emotion on response generation[1]. Commonsense and emotion are two fundamental elements of humans and indispensable to our social interactions [208, 209]. In recent years, there is an emerging research trend in endowing response generation with commonsense [22, 123, 126] or emotion [3, 21]. Commonsense-aware conversational models are able to leverage commonsense knowledge to reason and have been shown to produce more appropriate and informative responses [22]. Emotional conversational models are able to generate appropriate emotional responses, leading to improved user satisfaction [21, 210] (see Chapter 3) and long-term relationships with users [47].

Commonsense and emotion are not independent for humans [211]. In fact, emotions are often rational and commonsense-aware in social interactions [212]. For

[1]This chapter is published as *CARE: Commonsense-Aware Emotional Response Generation with Latent Concepts*, Proceedings of AAAI 2021 [51].

TABLE 5.1: Sample responses from EmoDS [3] and our model. EmoDS generates generic or unrelated emotional responses. Our model extracts the message concept "starbucks", and generates more commonsense-aware emotional responses by referring to our constructed relational (in **bold**) and emotional (in *italic*) latent concepts, e.g., **company**, **coffee** and *bitter*.

Model	**Message**	Why do you avoid starbucks?
	Gold	I don't like their coffee.
EmoDS	Neutral	I'm not a gamer so it's just that I'm a bit more tired of them.
	Joy	Yeah, that's pretty cool. I like that.
	Sadness	That sucks.
Ours	Neutral	It's just a general dislike of the **company**.
	Joy	**Tea** is my *favorite*.
	Sadness	I'm not a big fan of **coffee**. It 's too *bitter* for me.

example, our findings in Chapter 4 suggest that humans often rely on commonsense knowledge to express emotions in conversations. However, most existing conversational models only specialize in either commonsense or emotion but not both[2], which may lead to dull or unrelated responses. For example, as shown in Table 5.1, the state-of-the-art emotional dialogue system (EmoDS) [3] produces generic or unrelated emotional responses due to the lack of specific modeling of commonsense. Moreover, existing commonsense-aware conversational models, e.g., CCM [22], are not able to generate emotional responses, rendering them difficult to build long-term relationships with users [47].

A recent work [106] proposed to blend several human skills such as knowledge, personality, and empathy into a conversational agent and obtained the state-of-the-art performance in human evaluations. Their experimental analysis suggests that blending these skills is critical for achieving good human ratings. Motivated by the facts that 1) commonsense and emotion are two fundamental qualities of humans, and that 2) empirical performance improvement has been achieved via combining several human qualities [106], we hypothesize that combining commonsense and emotion into conversational agents can improve response quality and their human ratings.

To test our hypothesis, we propose a novel model for **C**ommonsense-**A**ware **R**esponse generation with specified **E**motions (**CARE**) and assess its empirical performance. Two major challenges to this task are 1) the lack of relevant datasets or resources that can provide such supervision and 2) how to generate appropriate

[2]One exception is XiaoIce [47], however, it has no public API and only supports Mandarin.

commonsense-aware emotional words. We tackle the first challenge by building an emotion-aware commonsense knowledge graph (**EA-CKG**) to integrate commonsense and emotion knowledge. We tackle the second challenge by incorporating both relational and emotional latent concepts constructed from EA-CKG into response generation. Specifically, we first build EA-CKG by augmenting an external CKG with emotional triplets extracted from emotional conversations (see Section 5.3.2). We then construct latent concepts using learned EA-CKG embeddings, endowing the response with commonsense and emotion by reasoning over the EA-CKG (see Section 5.3.2). Finally, we propose three methods to sequentially and collaboratively incorporate the latent concepts during attention, optimization, and sampling (see Section 5.3.3). CARE is illustrated in Figure 5.1.

In summary, our main contributions in this chapter are as follows:

- We identify the problem of lacking either commonsense or emotion in existing conversational models, which often leads to dull or unrelated responses. We hypothesize that combining commonsense and emotion into conversational agents can improve response quality.
- We propose CARE, the first commonsense-aware emotional response generation model, to address the aforementioned problem.
- We conduct extensive automatic and human evaluations, and show that CARE can produce better commonsense-aware emotional responses than state-of-the-art models that only specialize in one aspect. The experimental results support our hypothesis.

5.2 Related Work

Commonsense-Aware Response Generation: Existing commonsense-aware response generation models usually rely on knowledge bases. Extensive studies in this direction are emerging, such as open-domain response generation [36, 122–125, 168, 170, 171], task-oriented response generation [173, 213, 214] and question answering [177, 178, 215]. Zhou et al. [22] proposed CCM to incorporate commonsense knowledge by applying attention mechanisms on 1-hop knowledge triplets

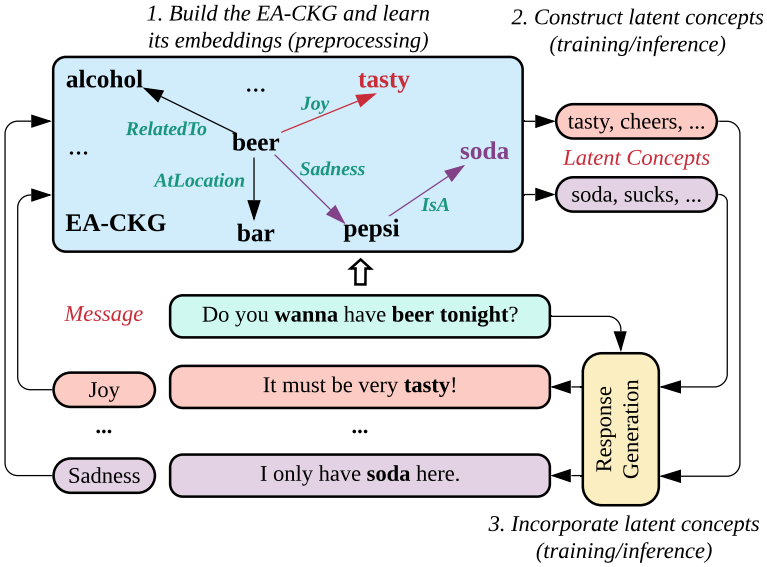


FIGURE 5.1: Illustration of CARE. Given the message "Do you wanna have **beer** tonight?" (**beer** is a message concept) and the learned EA-CKG embeddings, CARE first constructs latent concepts depending on the specified emotions of the response. For example, **tasty** is constructed for "joy" and **soda** is constructed for "sadness", because **tasty** is linked to **beer** via the "joy" relation, and **soda** is linked to **beer** via a composite of "sadness" and "IsA" relations. Then CARE leverages the proposed three methods to incorporate the latent concepts, e.g., **tasty**, into response generation.

for open-domain response generation. Zhang et al. [126] proposed ConceptFlow to extend CCM to multi-hop knowledge triplets. Different from CCM and ConceptFlow, our model is not restricted by the coverage of the CKG and can learn novel knowledge triplets for response generation.

Emotional Response Generation: Early approaches to emotional response generation primarily focus on rule-based methods [107]. Polzin and Waibel [108] employed different discourse strategies depending on the expressed emotions of the user, which require extensive hand-crafted rules to implement. Ochs et al. [107] designed an empathetic conversational agent that can express emotions based on the cognitive appraisal theories [109], which require extensive hand-crafted rules to implement. In recent years, neural emotional conversational models [2, 48, 111, 112, 135, 216] are also emerging. Zhou et al. [21] extended the Seq2Seq model by proposing an internal memory module to capture emotional state changes and an external memory module to generate emotional words. Song et al. [3] addressed the problem of dataset bias, i.e., the tendency to express the emotion category having the most number of training samples, by using an emotion classifier to guide

the response generation. In contrast, our model generates emotional responses by leveraging emotional latent concepts constructed from KG embeddings.

Controlled Text Generation: Recent controlled text generation methods are primarily based on generative adversarial networks (GAN) [217–219], language models [149] and Seq2Seq models [144, 220]. Keskar et al. [221] trained a Transformer-based conditional language model on a large collection of corpora with control codes that govern style, content, and task-specific behavior. Li and Sun [222] and Peng et al. [223] proposed topic-aware emotional response generation models. In contrast, we focus on commonsense, i.e., the semantic network of words, instead of topics, i.e., word clusters.

Commonsense Knowledge Base Completion: Different from knowledge entities in knowledge base completion [224] or relation extraction [225], commonsense entities can be arbitrary words or phrases and may not fit into a schema precisely. Several studies investigated how to construct new commonsense triplets using knowledge graph embeddings [226], Seq2Seq models [227] and language models [228]. Bosselut et al. [228] transferred a large pretrained language model on a seed set of triplets to generate novel triplets of high quality. However, the aforementioned methods cannot be trivially adapted to extract emotional CCP or build an emotion-aware CKG. Instead, we extract emotional triplets from emotional message-response pairs based on co-occurrence statistics.

5.3 Commonsense-Aware Response Generation with Specified Emotions (CARE)

In this section, we introduce the task definition and our CARE model, which includes a framework for constructing latent concepts and three methods to incorporate the latent concepts.

5.3.1 Task Definition

We denote $\{X_i, Y_i, e_i\}, i = 1, ..., N$, as a collection of $\{message, response, emotion\}$ tuples, where e_i is chosen from a predefined set of emotions and denotes the emotion

category of Y_i, and N denotes the number of conversations in the training dataset. Our task can be formulated as follows: given a new message X_{new} and an emotion category e, generate a natural and commonsense-aware response Y_{new} that has emotion e.

5.3.2 Latent Concepts Construction Framework

In this framework, we first build an emotion-aware commonsense knowledge graph (EA-CKG) and then construct latent concepts from EA-CKG.

EA-CKG

We extract emotional triplets from emotional conversations and augment them into an external CKG to obtain EA-CKG. We use ConceptNet [200] as our CKG[3]. Each triplet in ConceptNet follows the {*head*, *relation*, *tail*} format, e.g., {*beer*, *AtLocation*, *bar*}. Note that we use n-gram matching with ConceptNet to extract concepts from utterances, and ignore stopwords and n-grams that are formed entirely by stopwords. We define an emotional triplet as in the {*msg_concept*, *emotion*, *res_concept*} format, representing an emotional link from a message concept to a response concept. For example, given a message "I heard there is a bar nearby with nice beer." and its response "I love tasty beer." with joy emotion, the triplet {*beer*, *joy*, *tasty*} is a valid emotional triplet because there is a commonly expressed emotional link, i.e., *joy*, from *beer* in the message to *tasty* in the response.

We propose a two-step approach based on the pointwise mutual information (PMI) [229] to extract such emotional triplets from emotional conversations. PMI can measure the association between two words in a corpus. We extend the smoothed positive PMI, i.e., PPMI_α [230], as follows:

$$\text{PPMI}_\alpha(w_1, w_2) = \max\left(\log_2 \frac{P(w_1, w_2)}{P_\alpha(w_1) P_\alpha(w_2)}, 0\right), \tag{5.1}$$

where (w_1, w_2) denotes the word pair, $P_\alpha(w) = \frac{\text{count}(w)^\alpha}{\sum_x \text{count}(x)^\alpha}$ denotes the smoothed probability of w, and α denotes a smoothing factor set to 0.75 [230] to alleviate the bias towards rare words.

[3]We remove non-English and rare concepts.

TABLE 5.2: EA-CKG statistics. Reddit and Twitter are two conversation datasets used in our experiments.

CKG	#entity	#relation	#triplet
ConceptNet	182K	36	1.48M
EA-CKG (Reddit)	182K	42	1.58M
EA-CKG (Twitter)	182K	42	1.80M

In our two-step approach, we first construct a PPMI matrix between concepts in messages and in the corresponding responses to extract strongly associated concept pairs in conversations[4], denoted as conversational concept pairs (**CCP**). Note that in this case, w_1 refers to a message concept and w_2 refers to a response concept in Equation 5.1. We then construct a second PPMI matrix between CCP and their expressed emotions and extract CCP that statistically express certain emotions more often than other emotions[5]. Note that in this case, w_1 refers to a CCP and w_2 refers to its expressed emotion in Equation 5.1. We do not smooth $P(w_2)$. By using this two-step approach, we can effectively extract conversational triplets that are commonly expressed with certain emotions. The statistics of EA-CKG are presented in Table 5.2. Our approach shares similarities with commonsense knowledge base completion methods [226–228]; however, they cannot be trivially adapted to extract emotional CCP.

Latent Concepts Construction

During training and inference, given a message X_i and a desired emotion e_i, we construct the latent concepts of the response based on EA-CKG embeddings. Specifically, we first train a well-established knowledge embedding model, i.e., TransE [231][6], on the entire EA-CKG to learn global concept and relation embeddings. The embeddings in TransE are learned such that the score $-||\mathbf{h}+\mathbf{r}-\mathbf{t}||_2$ for a correct triplet (h, r, t) is much higher than a corrupted one, where $\mathbf{h}, \mathbf{r}, \mathbf{t}$ denote the TransE embeddings of h, r, t, respectively, and $||\mathbf{h}||_2 = 1$ and $||\mathbf{t}||_2 = 1$ [231]. Hence, given a message concept h, a relation r and a response concept t, we can

[4]We consider concept pairs whose frequency ≥ 5 and PPMI ≥ 1 as strongly associated pairs (CCP).

[5]We associate a CCP $\{w_1, w_2\}$ with emotion e if $\text{PPMI}(\{w_1, w_2\}, e) - \max_{e_i \neq e}\text{PPMI}(\{w_1, w_2\}, e_i) \geq 1$.

[6]We adopt TransE because it achieves only marginally worse performance than RotatE [224], a state-of-the-art knowledge graph embedding model, for triplet classification on ConceptNet, but much faster in inference.

estimate the relatedness between h and t via r as follows:

$$\text{score}(h, r, t) = (\mathbf{h} + \mathbf{r})^\top \mathbf{t}. \tag{5.2}$$

We then obtain the top m related latent concepts of the response from EA-CKG, i.e., $\{t\}_1^m$, as follows:

$$\{t_i\}_1^m = \underset{t}{\text{top}}(\text{score}(h, r, t)), \tag{5.3}$$

where $h \in C_{X_i}$, $r \in R \cup \{e_i\}$, C_{X_i} denotes all concepts in X_i and R denotes all 36 relations in ConceptNet, and t is searched over the concept vocabulary of EA-CKG. For messages without any concepts[7], we use a null message concept whose embedding is the average of all concept embeddings.

Framework Analysis

Our framework constructs plausible relational ($r \in R$) and emotional ($r = e_i$) concepts for the response. By leveraging the EA-CKG embeddings, our framework inherits the ideas from knowledge base completion and has two major advantages over the graph search methods used in existing models [22, 126] to find related concepts: 1) our framework can find concepts that are both commonsense-aware and emotional due to the incorporation of emotional triplets in EA-CKG, e.g., *tasty* is found given *beer* and *joy* whereas *bland* is found given *beer* and *sadness*; and 2) our framework can not only traverse through the EA-CKG to find related concepts in a multi-hop neighborhood but also discover an arbitrary number of novel related concepts using Equation 5.3, without being limited by the CKG coverage.

5.3.3 Incorporating Latent Concepts

After obtaining the latent concepts, we propose three methods to collaboratively incorporate them into our Transformer-based conversational model [49], as illustrated in Figure 5.2. Note that similar to the idea of persona embedding [40], we additionally employ an emotion embedding layer in our decoder.

[7] Around 3% messages do not have any concepts.

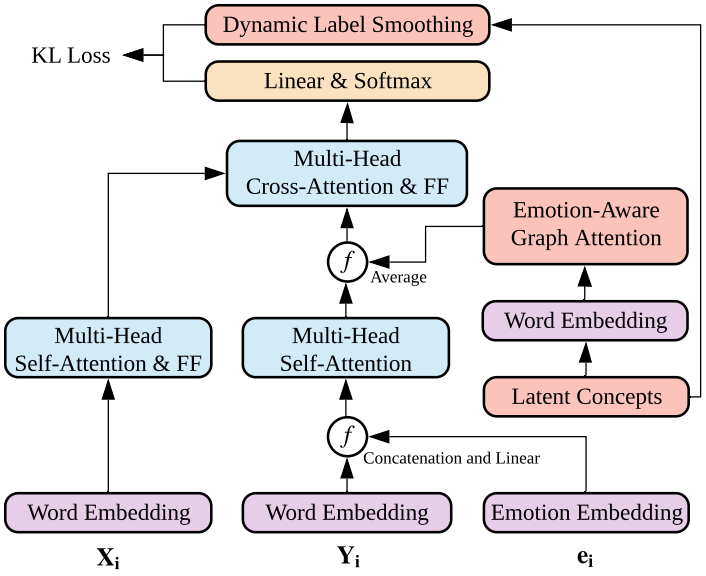


FIGURE 5.2: Architecture of our Transformer-based conversational model. The positional encoding, residual connection, and layer normalization are omitted in the illustration for brevity.

Emotion-Aware Graph Attention

We incorporate latent concepts into the decoder using an emotion-aware graph attention (**EAGA**) prior to the cross-attention layer, similar to our proposed graph attention in Chapter 4. Specifically, we assume that important latent concepts are those related to the message concepts and have strong emotional intensity. The relatedness is obtained from Equation 5.2. The emotional intensity is computed based on an emotion lexicon NRC_VAD [191] and an emotional intensity computation method (see Equation 4.7). We expand the size of NRC_VAD from 20K to 34K using synonym expansion for better coverage[8].

Formally, let $\{t_1, t_2, ..., t_m\}$ be the constructed latent concepts of response Y_i obtained from Equation 5.3, $\{s_1, s_2, ..., s_m\}$ be their relatedness scores obtained from Equation 5.2, and $\{q_1, q_2, ..., q_m\}$ be their emotion intensities based on NRC_VAD, we compute the latent concept embedding of Y_i, i.e., $\mathbf{C}_{Y_i}$, as follows:

$$\mathbf{C}_{Y_i} = \sum_{i=1}^{m} \beta_i \mathbf{t}_i, \tag{5.4}$$

[8]The expanded NRC_VAD covers more than 97% tokens in the datasets used in our experiments.

where $\mathbf{t}_i$ denotes the word embedding of t_i and β_i is computed as follows:

$$\beta_i = \lambda_i \frac{\exp(\delta_{1i} s_i)}{\sum_j \exp(\delta_{1j} s_j)} + (1 - \lambda_i) \frac{\exp(\delta_{2i} q_i)}{\sum_j \exp(\delta_{2j} q_j)}, \tag{5.5}$$

where λ_i denotes the trade-off coefficient between relatedness and emotional intensity, and δ_{1i}, δ_{2i} denote softmax temperatures. Note that λ_i, δ_{1i} and δ_{2i} are concept-specific and can be fixed *a prior* or learned during training. The obtained latent concept embedding $\mathbf{C}_{Y_i}$ is then averaged with the response representation prior to being fed to the cross-attention layer[9]. Compared with the graph attention in [22], our EAGA measures concept relatedness using translation-based distance in TransE instead of MLP and additionally considers the emotion property of concepts.

Dynamic Label Smoothing

Label smoothing is conventionally adopted in the Transformer [49] to improve translation quality. We propose a dynamic label smoothing (**DLS**) method to explicitly enforce the supervision of latent concepts in producing concept-related responses, as well as to stabilize the learning process. Specifically, starting from the conventional label smoothing, we linearly increase the smoothing values for latent concepts according to the training step and decrease the smoothing values for other words in the vocabulary. Note that the smoothing value of the target word remains unchanged. The maximum of the total smoothing value for latent concepts is a hyper-parameter to be tuned in experiments. We optimize model parameters to minimize the Kullback-Leibler (KL) loss [232].

Concept-Aware Top-K Decoding

During inference, we propose a concept-aware top-K decoding (**CATD**) method to encourage the generation of words that are more related to the associated latent concepts. Formally, given the conventional top-K unnormalized token probabilities $P(w_1), ..., P(w_k)$, our concept-aware token probability $P^{'}$ for $w_i, i = 1, ..., k$, is

[9]We experimented with feeding the concept embedding into other layers using linear transformation or averaging but observed inferior performance.

computed as follows:

$$P^{'}(w_i) = P(w_i) * P_c^{\gamma}(w_i), \tag{5.6}$$

where γ denotes a trade-off hyper-parameter between fluency and relatedness to latent concepts, and $P_c(w_i)$ is computed as follows:

$$P_c(w_i) = \frac{\exp(\mathbf{C}_Y^{\top}\mathbf{w}_i)}{\sum_{i=1}^{k}\exp(\mathbf{C}_Y^{\top}\mathbf{w}_i)}, \tag{5.7}$$

where $\mathbf{C}_Y$ denotes the latent concept embedding obtained from Equation 5.4 during inference. One merit of the CATD is that it only reorders top-K tokens by additionally considering their relatedness to latent concepts and thus does not introduce unlikely tokens into the sampling process.

5.4 Experiments

In this section, we present our datasets, evaluation metrics, baselines, model settings, experimental results and analysis.

5.4.1 Datasets and Evaluation Metrics

Following prior studies on social media datasets [21, 22], we conduct experiments on two large-scale English datasets, namely Reddit and Twitter. The Reddit dataset is obtained from comments on the CasualConversation subreddit[10] discussing a variety of casual topics[11]. The Twitter dataset is obtained from question-answer pairs on twitter.com[12].

For Reddit, we follow [233] to extract {*message*, *response*} pairs. For each extracted sentence, we apply lower-casing and remove URLs, newlines, "[]" and "()". The latter two often appear around URLs and image captions. Finally, we tokenize each sentence using the Spacy[13] tokenizer.

[10]https://www.reddit.com/r/CasualConversation/
[11]https://files.pushshift.io/reddit/comments/
[12]https://github.com/Marsan-Ma/chat corpus/
[13]https://spacy.io/

TABLE 5.3: Dataset statistics.

		Reddit	**Twitter**
Training	Neutral	268K	649K
	Joy	232K	308K
	Sadness	236K	302K
	Surprise	551K	543K
	Fear	156K	325K
	Anger	132K	373K
	Total	1.58M	2.50M
Validation	Total	49K	50K
Testing	Total	49K	50K

For Twitter, we apply lower-casing and use the tweet preprocessor[14] to remove mentions, hashtags, URLs, reserved words, and numbers for each tweet. Each tweet is then tokenized using the Spacy tokenizer.

For both datasets, we truncate each sentence to a maximum of 30 tokens. We use the most frequent 30K words as the vocabulary for each dataset.

To obtain the ground-truth emotion labels for each response, similar to [3, 21], we train an emotion classifier on emotional conversations. Specifically, we use the emotional tweets released by Mohammad [234] and Mohammad et al. [235] to train the emotion classifier. We consider neutral and Ekman's six basic emotions [204]: joy, sadness, surprise, fear, and anger, but exclude disgust due to its small amount of training samples in the emotional tweets. We propose an emotion classifier based on DeepMoji embeddings [236] followed by a linear layer and a softmax layer (included in the supplementary material). Our classifier achieves an accuracy of 0.562 on a balanced test dataset, outperforming several competitive baselines such as BiLSTM (0.446), CNN (0.547), BERT [76] (0.530) and XLNet [79] (0.522). We then use our trained emotion classifier to annotate the responses in the datasets. The statistics of the datasets are presented in Table 5.3.

We conduct both automatic and human evaluations. Automatic evaluations include 1) **Fluency**: perplexity (PPL), which measures the confidence of the generated responses; 2) **Diversity**: distinct-1 (dist-1) and distinct-2 (dist-2) [97], which measure the percentage of unique unigrams and bigrams in the generated responses, respectively; 3) **Emotion Accuracy (EA)**: the emotion accuracy of the generated responses measured by our trained emotion classifier; and 4) **Commonsense**

[14] https://github.com/s/preprocessor

Awareness (CA): the average number of commonsense triplets in one pair of message and generated response, measured by ConceptNet.

Following [21], we conduct human evaluations to measure both the **content quality** (rating scale in {0, 1, 2}) and **emotion quality** (rating scale in {0, 1}) of the generated responses. Content quality measures whether the response is natural and related to the message, as well as how commonsense-aware the response is. Emotion quality measures whether the response expresses the desired emotion appropriately and accurately. We randomly sample 200 test messages and emotions to generate 200 responses for each model. Each response is then annotated by three annotators.

5.4.2 Baselines and Model Settings

We compare CARE with the following baselines:

Vanilla Models: Seq2Seq [18] and Transformer [49].

Commonsense-Aware Models: CCM [22] and ConceptFlow [126]. ConceptFlow leverages multi-hop knowledge triplets and is a state-of-the-art model for commonsense-aware response generation.

Emotional Models: ECM [21] and EmoDS [3]. EmoDS is a state-of-the-art model for emotional response generation.

Pre-trained Model: CTRL [221]. CTRL is a large pre-trained conditional language model with 1.6 billion parameters trained on 140GB of text. We fine-tune CTRL on our training conversations such that it is able to produce emotional responses. CTRL has also been shown to contain commonsense knowledge [228, 237].

All baselines except CTRL have a vocabulary size of 30K and are initialized with GloVe [158] embeddings of size 300. We use top-10 decoding [238, 239] for all baselines[15] except CTRL, which has a specially designed decoding algorithm. All baselines except CTRL are implemented using OpenNMT-py [240] in PyTorch on NVIDIA V100 GPUs.

[15]Top-10 decoding performs much better than beam search with beam sizes ranging from 5 to 40 in our manual inspection. In particular, beam search suffers from producing generic responses.

TABLE 5.4: Automatic evaluation results on Reddit. Size denotes model size. IT denotes inference time relative to Seq2Seq.

Models	PPL	Dist-1	Dist-2	EA	CA	Size	IT
Seq2Seq	**57.2**	0.0035	0.0347	-	0.1349	38M	**1.0x**
Transformer	63.8	0.0032	0.0371	-	0.1224	**20M**	1.5x
CCM	62.3	0.0046	0.0469	-	0.1222	74M	5.9x
ConceptFlow	60.1	0.0047	0.0458	-	0.1375	33M	21.8x
ECM	65.6	0.0044	**0.0506**	0.5893	0.1105	40M	2.0x
EmoDS	76.6	0.0030	0.0455	0.6186	0.1107	46M	1.5x
CTRL	-	**0.0068**	0.0447	0.3425	0.1502	1.6B	1876.7x
Ours	70.4	0.0049	0.0460	**0.6840**	**0.1538**	20M	1.9x

TABLE 5.5: Automatic evaluation results on Twitter.

Models	PPL	Dist-1	Dist-2	EA	CA
Seq2Seq	**79.7**	0.0047	0.0522	-	0.1653
Transformer	90.1	0.0053	0.0563	-	0.1728
CCM	82.5	0.0060	0.0663	-	0.1835
ConceptFlow	89.1	0.0051	0.0556	-	0.1893
ECM	91.3	0.0056	0.0630	0.5619	0.1650
EmoDS	113.5	0.0030	0.0450	0.5950	0.1599
CTRL	-	**0.0108**	**0.0851**	0.3995	0.1958
Ours	100.1	0.0064	0.0775	**0.6693**	**0.2304**

We use the same hyper-parameters on both datasets. Our TransE embeddings have a dimension of 100 and achieve an accuracy of 0.89 for triplet classification on EA-CKG. Our Transformer model has 1 layer and 4 attention heads. We initialize the word embedding layer with pre-trained GloVe embeddings [158] of size 300. The emotion embedding and feedforward layers have sizes of 50 and 512, respectively. We train our model using Adam [163] with a learning rate of 1, a batch size of 64 and dropout of 0.1 for 80K steps, including 6K steps for warmup. We empirically construct 30 relational latent concepts and 10 emotional latent concepts for each response using Equation 5.3. We use label smoothing of 0.1, total smoothing value of 0.08 for latent concepts in DLS, and top-10 decoding with $\gamma = 1$ in CATD.

5.4.3 Comparison with Baselines

We present the results of automatic evaluations in Table 5.4 and 5.5. Seq2Seq achieves the lowest perplexity while Transformer achieves slightly better diversity than Seq2Seq. Commonsense-aware models, i.e., CCM and ConceptFlow, obtain slightly better diversity and CA; however, they are unable to generate responses

TABLE 5.6: Human evaluation results for content quality. The inter-annotator agreement, measured by Fleiss' Kappa [4], are 0.441 and 0.479 for Reddit and Twitter, respectively. Both datasets obtain "moderate agreement" and "substantial agreement".

	Models	Neutral	Joy	Sadness	Surprise	Fear	Anger	Total
Reddit	Seq2Seq	0.62	0.79	0.69	0.78	**0.72**	0.74	0.73
	ConceptFlow	0.82	0.96	0.81	0.89	0.70	0.76	0.83
	EmoDS	0.76	0.89	0.86	0.71	0.63	0.68	0.75
	CTRL	**0.92**	**1.08**	**1.03**	0.79	0.66	**0.93**	**0.90**
	Ours	0.78	0.98	0.88	**0.92**	0.63	0.81	0.84
Twitter	Seq2Seq	0.92	0.76	0.79	0.85	0.81	0.99	0.86
	ConceptFlow	0.97	0.91	0.98	1.03	0.87	0.85	0.93
	EmoDS	0.82	0.78	0.91	0.93	0.79	0.84	0.84
	CTRL	**1.08**	**1.05**	**1.16**	**1.21**	0.92	**1.12**	**1.09**
	Ours	0.87	0.83	1.13	1.15	**0.93**	0.94	0.96

TABLE 5.7: Human evaluation results for emotion quality. The inter-annotator agreement, measured by Fleiss' Kappa [4], are 0.626 and 0.673 for Reddit and Twitter, respectively. Both datasets obtain "moderate agreement" and "substantial agreement".

	Models	Neutral	Joy	Sadness	Surprise	Fear	Anger	Total
Reddit	Seq2Seq	0.34	0.32	0.15	0.35	0.19	0.08	0.24
	ConceptFlow	0.45	0.35	0.17	0.31	0.16	0.15	0.26
	EmoDS	0.66	0.72	**0.67**	0.52	0.41	0.38	0.56
	CTRL	0.50	0.63	0.42	0.34	0.24	0.38	0.42
	Ours	**0.68**	**0.75**	0.63	**0.76**	**0.44**	**0.42**	**0.62**
Twitter	Seq2Seq	0.33	0.23	0.21	0.17	0.25	0.29	0.25
	ConceptFlow	0.42	0.28	0.22	0.19	0.21	0.26	0.27
	EmoDS	0.46	0.48	0.56	0.63	0.65	**0.65**	0.57
	CTRL	0.54	**0.62**	0.50	0.68	0.71	0.61	0.61
	Ours	**0.57**	0.58	**0.62**	**0.71**	**0.74**	0.63	**0.64**

with specified emotions. Emotional models, i.e., ECM and EmoDS, achieve the highest EA among all baselines but the worst in perplexity and CA, suggesting that they only specialize in emotion and neglect commonsense. It is not surprising that CTRL achieves the highest diversity among all models, partially due to its large vocabulary size of 250K. However, it achieves inferior EA. Our model achieves better EA and CA than all baselines, especially CTRL[16], which is also capable of producing commonsense-aware emotional responses, suggesting that our model can generate more commonsense-aware responses while achieving higher EA.

[16]We initially suspected that this is due to underfitting; however, low EA persists even we gradually increased the number of fine-tuning steps until observing noticeable overfitting.

We present the results of human evaluations in Table 5.6 and 5.7. The responses of all non-emotional models are generated via top-10 decoding six times. ConceptFlow obtains similar emotion quality but noticeably better content quality than Seq2Seq due to its incorporation of multi-hop triplets. EmoDS achieves comparable content quality but much better emotion quality than Seq2Seq. CTRL obtains the best content quality among all models but only mediocre emotion quality, especially on Reddit. Our model performs best in emotion quality (t-test, $p < 0.01$). In addition, our model achieves significantly better content quality than EmoDS (t-test, $p < 0.01$), showing that our model can produce better commonsense-aware emotional responses than EmoDS. Finally, our model outperforms ConceptFlow in content quality. One possible reason is that the graph search method in ConceptFlow heavily relies on the coverage of ConceptNet to extract knowledge triplets, but ConceptNet only has an average coverage of 27% on Reddit and Twitter. In contrast, our model has less such restriction and can construct an arbitrary number of latent concepts given any input message.

We further analyze the space and time complexity of all models in the rightmost columns of Table 5.4. Notably, our model has comparable space and time complexity with respect to vanilla baselines. In contrast, despite the competitive performance of CTRL, it is around 80x larger in model size and 1,000x slower in inference than our model, rendering it intractable for real-time applications.

5.4.4 Model Analysis

We conduct ablation study and present the results in Table 5.8 and 5.9. It is clear that the removal of any component except EAGA leads to much worse performance in both EA and CA, validating the importance of these components in our model. In particular, we observe that 1) our approach of constructing latent concepts performs better than alternatives (-ET+EL and -TransE); and 2) the removal of EAGA leads to significantly higher perplexity, diversity, and CA. The higher perplexity may be attributed to the additional supervisions of DLS on latent concepts, which are not explicitly incorporated into the model due to the lack of EAGA. The higher diversity and CA may be attributed to the untrained λ, δ_1, and δ_2 (see Equation 5.5), which sometimes leads to ungrammatical but diverse latent

TABLE 5.8: Ablation study on Reddit. **-ET+EL**: replace the tails of the extracted emotional triplets (ET) by randomly sampled corresponding emotional words from an emotion lexicon (EL) [5]. **-TransE**: instead of using TransE, search neighbors with a growing neighborhood size (up to 3) on EA-CKG to find latent concepts based on the message and emotion. **-EAGA**: remove the emotion-aware graph attention. **-DLS**: remove the dynamic label smoothing. **-DLS+LS**: replace the dynamic label smoothing by conventional label smoothing (LS) of 0.1. **-CATD**: replace the concept-aware top-K decoding by the conventional top-K decoding.

Models	PPL	Dist-1	Dist-2	EA	CA
Ours (CARE)	**70.4**	**0.0049**	0.0460	**0.6840**	0.1538
-ET+EL	72.2	0.0040	0.0428	0.6518	0.1332
-TransE	72.8	0.0039	0.0430	0.6595	0.1261
-EAGA	79.6	0.0045	**0.0484**	0.6258	**0.1635**
-DLS	72.6	0.0038	0.0441	0.6497	0.1277
-DLS + LS	72.5	0.0040	0.0443	0.6421	0.1318
-CATD	**70.4**	0.0036	0.0373	0.6094	0.1394

TABLE 5.9: Ablation study on Twitter. Refer to Table 5.8 for details of ablated models.

Models	PPL	Dist-1	Dist-2	EA	CA
Ours (CARE)	**100.1**	0.0064	0.0775	**0.6693**	0.2304
-ET+EL	100.8	0.0057	0.0669	0.6266	0.2077
-TransE	101.8	0.0057	0.0660	0.6391	0.1960
-EAGA	116.3	**0.0080**	**0.1303**	0.4775	**0.3512**
-DLS	100.7	0.0056	0.0682	0.6162	0.2050
-DLS + LS	101.2	0.0055	0.0675	0.6194	0.2013
-CATD	**100.1**	0.0059	0.0630	0.5848	0.1903

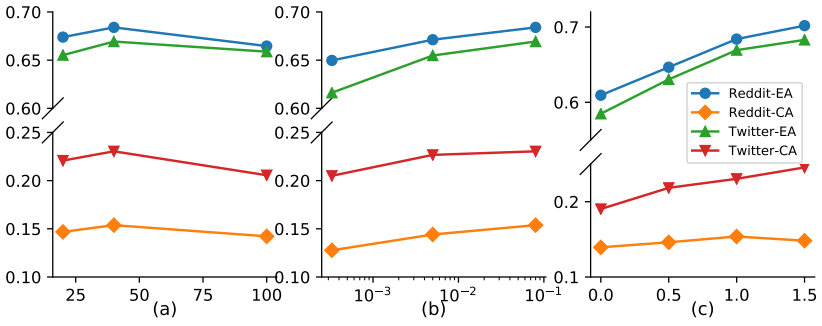


FIGURE 5.3: Hyper-parameter analysis on EA and CA. (a) Different number of latent concepts for each response (see m in Equation 5.3), where 1/4 latent concepts are emotional. (b) Different total smoothing values for latent concepts in DLS. (c) Different γ (see Equation 5.6) in CATD.

concepts during decoding. Our observation validates the importance of EAGA in attending more related latent concepts.

TABLE 5.10: Case studies. Words in **bold** and *italic* denote relational and emotional latent concepts, respectively.

Message		Why do you avoid starbucks?
Seq2Seq		It's not bad.
Ours (CARE)	Neutral	It 's just a general dislike of the **company**.
	Joy	**Tea** is my *favorite*.
	Sadness	I'm not a big fan of **coffee**. It 's too *bitter* for me.
	Surprise	I'm not sure. Maybe because of the **coffee shop** thing?
	Fear	I'm not sure. It's just a *weird* feeling.
	Anger	I hate the taste of **coffee**. It's just so *bitter* and *gross*.
Message		I'd like to confess that I'm a cat person way more then a dog person.
Seq2Seq		I hope you get the chance.
Ours (CARE)	Neutral	I'm curious to hear the story behind this!
	Joy	Yes! I *love* their **eyes**, I *love* them.
	Sadness	I don't understand the difference though :(
	Surprise	What kind of **cat** you like ?
	Fear	That's the *worst*!
	Anger	That's a little *annoying*!

We analyze the impact of different model hyper-parameters on EA and CA, as shown in Figure 5.3. Using 40 latent concepts achieves the sweet spot for model complexity. Regarding DLS, increasing the total smoothing values for latent concepts in the $[0, 0.08]$ range improves model performance. However, we do observe degraded fluency when using larger smoothing values, which is expected because the true learning signal is weakened. Increasing γ in CATD consistently improves EA and CA for our model. However, models using larger γ, e.g., 1.5, sometimes produce unfluent long responses due to its overemphasizes on latent concepts.

5.4.5 Case Study and Error Analysis

We present two sample cases in Table 5.10. Given a message and desired emotions, our model produces commonsense-aware responses with the desired emotions, guided by both relational and emotional latent concepts. For example, given the message concept "starbucks" and the anger emotion, the relational latent concept "coffee" and the emotional latent concept "gross" are constructed and incorporated into response generation. However, we do observe bad cases where the latent concepts overemphasize on emotional intensity, and the response becomes unnatural.

5.4.6 Limitation

One major limitation of our work is the mediocre accuracy of our trained emotion classifier, which can be attributed to the unavailability of large-scale datasets for emotional conversations and sentences. Nevertheless, our proposed lightweight classifier obtains better performance than the best models reported in [3, 21] and BERT. A potential solution to this limitation is to leverage few-shot learning on BERT-like models.

5.5 Summary

In this chapter, we propose CARE as the first attempt to test the hypothesis that combing commonsense and emotion into conversational agents can improve response quality and human ratings. Specifically, we build an EA-CKG and leverage its TransE embeddings to allow CARE to reason over the EA-CKG and construct both relational and emotional latent concepts. We further propose three methods to collaboratively incorporate the latent concepts into response generation. Extensive ablation studies show that our methods of constructing and incorporating latent concepts outperform alternative methods. In addition, both automatic and human evaluations show that CARE can produce more accurate and commonsense-aware emotional responses than state-of-the-art commonsense-aware models and emotional models. Finally, our work provides empirical evidence for our hypothesis.

Chapter 6

Persona-Based Empathetic Conversational Agents

6.1 Overview

In the previous chapters, we have proposed several methods of endowing open-domain conversational agents (CAs) with emotion and commonsense. In this chapter, we focus on incorporating other two human traits, i.e., persona and empathy, into open-domain CAs[1]. In addition, we study the impact of persona and empathy on CAs. In particular, we investigate the link between persona and empathy in human conversations.

Empathy refers to the capacity to understand or feel another's mental states and respond appropriately [43]. Empathy is vital in building good interpersonal relationships [44]. In NLP, empathetic conversational models have been shown to improve user satisfaction and task outcomes in numerous domains [45–47, 130, 148]. For example, empathetic agents received more positive user ratings, including greater likeability and trustworthiness than controls [241].

Early studies in empathetic conversational models are primarily based on rules. Klein [45] designed scripted dialogues to handle user frustration and found that empathetic dialogues significantly helped people handle the frustration than two

[1]This chapter is published as *Towards Persona-Based Empathetic Conversational Models*, Proceedings of EMNLP 2020 [52].

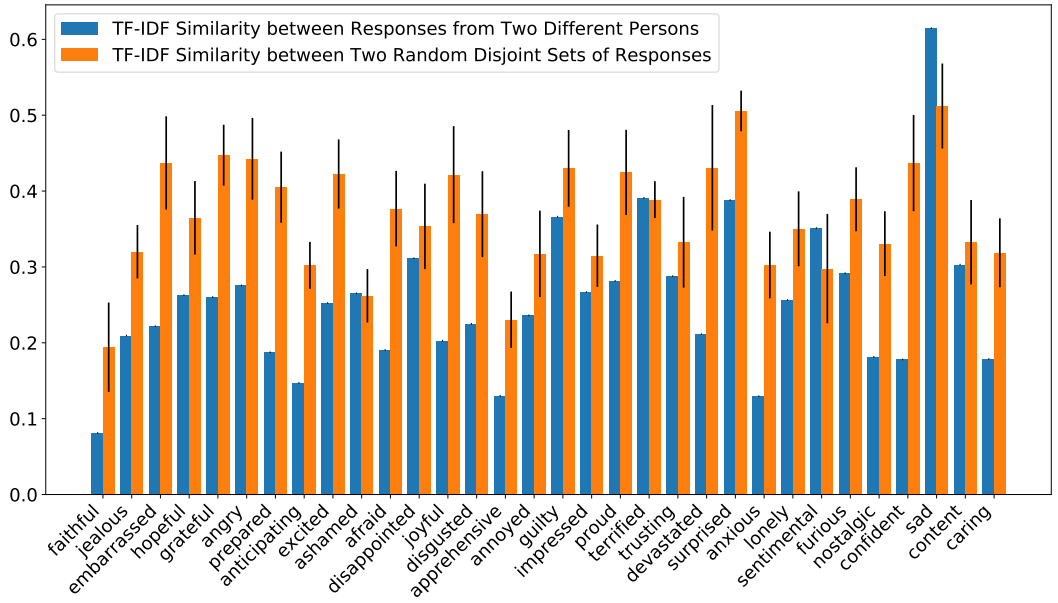


FIGURE 6.1: TF-IDF similarity between two sets of empathetic responses [2] for each emotion (best viewed in color). For most emotions (28 out of 32), the similarity between responses from two different speakers (blue) is substantially smaller than the similarity between two random disjoint sets of responses (orange, averaged over five runs).

controls. In recent years, neural network based conversational models [18, 19] are becoming dominant. Zhou et al. [47] designed XiaoIce, a popular AI companion with an emotional connection to satisfy the human need for communication, affection, and social belonging. Recently, Rashkin et al. [2] presented a new dataset and benchmark towards empathetic conversations and found that both Transformer-based generative models [49] and BERT-based retrieval models [76] relying on this dataset exhibit stronger empathy.

However, most existing studies, e.g., [2], do not consider persona when producing empathetic responses[2]. In Psychology, persona refers to the social face an individual presents to the world [37]. Persona is different from personality[3] but they have been shown to be highly correlated [38]. Personality has also been shown to influence empathy [242–244]. For example, the Big Five personality traits have been shown to be able to predict the empathy skills accurately across different cultures [244]. In addition, our empirical analysis of empathetic conversations in Figure

[2]One exception is XiaoIce [47]; however, her persona is not configurable and thus difficult to satisfy various human needs.

[3]Persona is considered more subjective because it is the appearance an individual wants to present to the world, whereas personality is considered more objective because it is the characteristics of an individual.

6.1 also shows that for most emotions, the empathetic responses from two different persons[4] have more differences than that between two disjoint sets of random responses, suggesting that different speakers have different "styles" for empathetic responding. Both the theories in Psychology and the evidence from our empirical analysis suggest that persona plays an important role in empathetic conversations, which, to the best of our knowledge, has not been investigated before[5].

To this end, we propose a new task towards persona-based empathetic conversations and present the first empirical study on the impact of persona on empathetic responding. However, one major challenge of this task is the lack of relevant datasets, i.e., existing datasets only focus on either persona or empathy but not both (see Table 6.4 for details).

In this chapter, we propose a novel large-scale multi-turn **P**ersona-based **E**mpathetic **C**onversation (**PEC**) dataset in two domains with contrasting sentiments, obtained from the social media Reddit, to facilitate our study.

We then propose CoBERT, an efficient BERT-based response selection model using multi-hop co-attention to learn higher-level interactive matching. CoBERT outperforms several competitive baselines on PEC, including Poly-encoder [245], the state-of-the-art BERT-based response selection model, by large margins. We conduct additional comparisons with several BERT-adapted models and extensive ablation studies to evaluate CoBERT more comprehensively.

Finally, based on PEC and CoBERT, we investigate the impact of persona and empathy on response quality and human ratings. In particular, we study the impact of persona on empathetic responding. In addition, we analyze how limited persona data improves model performance and how our model generalizes to new personas.

In summary, our main contributions in this chapter are as follows:

- We propose a new task and a novel large-scale multi-domain dataset, PEC, towards persona-based empathetic conversations.

[4]Each response in [2] has a speaker id but no persona.

[5]A very recent work [106] incorporates persona and empathy by fine-tuning on corresponding datasets; however, it does not investigate the impact of persona on empathetic responding.

- We propose CoBERT, a BERT-based response selection model that obtains the state-of-the-art performance on PEC. Extensive experimental evaluations show that CoBERT is both effective and efficient.
- We conduct experiments and show that both persona and empathy are beneficial to response quality and human ratings. In addition, we present the first empirical study on the impact of persona on empathetic responding. The results show that persona improves empathetic responding *more* when CoBERT is trained on empathetic conversations than non-empathetic ones, establishing an empirical link between persona and empathy in human conversations. This link may suggest that persona has a larger impact on empathetic responding than casual responding.

6.2 Related Work

Empathetic Conversational Models: Despite the growing number of studies in neural conversational models, less attention has been paid to make conversations empathetic until recently [2, 131–136, 246, 247], possibly due to the lack of empathetic conversation datasets. Rashkin et al. [2] proposed EMPATHETICDIALOGUES (**ED**), the first empathetic conversation dataset comprising 25K conversations in 32 emotions. Conversational models trained on the role of the listener in the dataset exhibited stronger empathy than models trained on non-empathetic datasets. The comparison between ED and PEC is presented in the last paragraph of Section 6.3. Recently, Lin et al. [135] proposed a mixture of empathetic listeners that first captures the user emotion distribution at each turn and then softly combines the output states of each emotional listener to generate empathetic responses.

Persona-Based Conversational Models: In recent years, personalized conversational models are emerging [7, 40, 127–129, 248]. Li et al. [40] proposed persona embeddings in a response generation model and achieved improved generation quality and persona consistency. Zhang et al. [7] proposed PERSONA-CHAT (**PC**), a crowd-sourced conversation dataset with persona information, to improve model engagingness and consistency. Mazare et al. [8] further presented a much larger persona-based conversation dataset collected from Reddit (**PCR**) and showed that

TABLE 6.1: Statistics of PEC. #Avg.PS and #Std.PS denote average and standard deviation of the number of persona sentences per speaker, respectively. #Avg.U denotes the average utterance length. #Avg.P denotes the average persona sentence length.

	happy			**offmychest**		
	train	valid	test	train	valid	test
#Conv.	157K	20K	23K	124K	16K	15K
#Utter.	367K	46K	54K	293K	38K	35K
#Speaker	93K	17K	19K	89K	16K	16K
#Avg.PS	66.0	70.8	70.0	59.6	66.8	67.1
#Std.PS	38.1	36.7	36.9	40.2	39.0	38.8
#Avg.U	21.5	21.9	21.3	30.4	31.5	30.0
#Avg.P	10.9	10.8	10.8	10.9	10.9	10.9

persona consistently improves model performance even when a large number of conversations is available. The comparison among PC, PCR, and PEC is presented in the last paragraph of Section 6.3. Recently, Gu et al. [249] proposed DIM, a personalized response selection model with interactive matching and hierarchical aggregation, and achieved state-of-the-art performance on PC.

Retrieval-Based Conversational Models: Recent neural retrieval-based conversational models generally have three modules: encoding, matching and aggregation [19, 92–94, 250–253] (see Section 2.2.2). The encoding module encodes text into vector representations using encoders such as LSTM, Transformer, or BERT. The matching module measures context-response associations using various attention mechanisms at different granularities. The aggregation module summarizes the matching information along each sequence to obtain the final representation. A recent work Humeau et al. [245] proposed Poly-encoder, an efficient BERT-based response selection model that obtained the state-of-the-art performance on multiple conversation datasets.

6.3 The PEC Dataset

In this section, we introduce the collection procedure and statistics of our proposed persona-based empathetic conversation (PEC) dataset.

TABLE 6.2: Sentiment and empathy of PEC and the control group based on human ratings. Sentiment ranges from -1 (negative) to 1 (positive). Empathy ranges from 0 (non-empathetic) to 1 (empathetic). Ratings are aggregated by majority voting (averaging shows similar results). The inter-annotator agreement, measured by Fleiss' kappa [6], for sentiment and empathy are 0.725 and 0.617, respectively. Both agreement statistics indicate "substantial agreement".

	happy	**offmychest**	**control group**
Sentiment	0.85	-0.39	0.03
Empathy	0.73	0.61	0.25

Data Source We collect empathetic conversations from two subreddits *happy*[6] and *offmychest*[7] on Reddit, a discussion forum where users can discuss any topics on their corresponding sub-forums/subreddits. The *happy* subreddit is where users share and support warm and happy stories and thoughts. The *offmychest* subreddit is where users share and support deeply emotional things that users cannot tell people they know. We choose these two subreddits as our data source because their posts have contrasting sentiments and their comments are significantly more empathetic than casual conversations, i.e., the control group, as shown in Table 6.2.

Conversation Collection Discussions on Reddit are organized in threads where each thread has one post and many direct and indirect comments. Each thread forms a tree where the post is the root node, and all comment nodes reply to their parent comment nodes or directly to the root node. Therefore, given a thread with n nodes, we can extract $n-1$ conversations where each conversation starts from the root node and ends at the $n-1$ non-root nodes. We randomly split conversations by threads according to the ratio of 8:1:1 for training, validation, and test sets, respectively.

Persona Collection Following [8], for each user in the conversations, we collect persona sentences from all posts and comments the user wrote on Reddit. The posts and comments are split into sentences, and each sentence must satisfy the following rules to be selected as a persona sentence: 1) between 4 and 20 words; 2) the first word is "i"; 3) at least one verb; 4) at least one noun or adjective; and 5) at least one content word. Our rules are stricter than that from [8], allowing us to extract less noisy persona sentences. For each user, we extract up to 100 persona sentences.

[6]https://www.reddit.com/r/happy/
[7]https://www.reddit.com/r/offmychest/

TABLE 6.3: Two example conversations with personas from PEC. The persona sentences correspond to the last speakers in the conversations.

	happy	**offmychest**
Conversation	Celebrating 43 years of marriage with the love of my life.	Worried. Am I becoming depressed again? Please don't leave me. Is everything okay? You don't seem yourself.
	She looks very young for someone who has been married 43 years. That must surely put her in the 63-73yr age range?!	I'm living these exact words.
	I just turned 61, thanks!	I hope everything works out for you. I'm trying not to fall apart.
	I hope I look that young when I'm 61! You guys are too cute, congratulations :)	Me too. If you ever want someone to talk to my messages are open to you.
Persona	I took an 800 mg Ibuprofen and it hasn't done anything to ease the pain.	I think I remember the last time I ever played barbies with my litter sister.
	I like actively healthy.	I have become so attached to my plants and I really don't want it to die.
	I want a fruit punch!	I'm just obsessed with animals.

TABLE 6.4: Comparisons between PEC and related datasets. ED denotes EMPATHETICDIALOGUES [2]. PC denotes PERSONA-CHAT [7]. PCR denotes the persona-based conversations from Reddit [8]. CS denotes crowd-sourced. The size denotes the number of expanded conversations.

Dataset	Source	Persona	Empathy	Size	Public
ED	CS	✗	✓	78K	✓
PC	CS	✓	✗	151K	✓
PCR	Reddit	✓	✗	700M	✗
PEC (ours)	Reddit	✓	✓	355K	✓

Note that we choose our approach to persona collection because 1) the well-established work [8] successfully trained personalized agents using this approach; 2) this approach is significantly more scalable and cost-effective than crowd-sourcing; and 3) we are concerned that using crowd-sourcing, i.e., assigning artificial personas to crowd-workers and asking them to chat empathetically based on the assigned personas, would introduce worker-related noises such that models may merely learn superficial empathetic responding patterns that crowd-workers deem suitable given the assigned personas.

Data Processing We keep a maximum of 6 most recent turns for each conversation. We filter conversations to ensure that 1) each post is between 2 and 90

words; 2) each comment is between 2 and 30 words[8]; 3) all speakers have at least one persona sentence; and 4) the last speaker is different from the first speaker in each conversation. The last requirement is to maximally ensure that the last utterance is the empathetic response instead of a reply of the poster. In addition, persona sentences appearing in the conversation responses are removed to avoid data leakage. Finally, we lower-case all data and remove special symbols, URLs, and image captions from each sentence. The statistics of PEC are presented in Table 6.1. Two examples of PEC are shown in Table 6.3.

Note that it may not be easy to see explicit links in Table 3, but that is exactly what we are studying for, i.e., to uncover the implicit (and possibly unexpected) links between persona and empathy using real user data. For example, the utterance "I hope I look that young" may implicitly link to the persona "I like actively healthy" in Table 6.3.

Data Annotations We manually annotate 100 randomly sampled conversations from each domain to estimate their sentiment and empathy. To avoid annotation bias, we add a control group comprising 100 randomly sampled casual conversations from the *CasualConversation*[9] subreddit, where users can casually chat about any topics. Finally, we mix and shuffle these 300 conversations and present them to three annotators. The annotation results are presented in Table 6.2. The posts in the happy and offmychest domains are mostly positive and negative, respectively. Both domains are significantly more empathetic than the control group ($p < 0.001$, one-tailed t-test).

Conversation Analysis We conduct conversation analysis for PEC, similar to our analysis for ED [2] in Figure 6.1. Specifically, the TF-IDF similarities between responses from two different persons are 0.25 and 0.17 for happy and offmychest, respectively, whereas the TF-IDF similarities between two disjoint sets of random responses are 0.38 ($\pm$0.05) and 0.31 ($\pm$0.05) for happy and offmychest over 5 runs, respectively. The results show that empathetic responses between different persons are more different than that between random empathetic responses in PEC, suggesting that different speakers in PEC have different "styles" for empathetic responding.

[8]Posts are usually longer than comments. 87% posts and 82% comments on *happy* are less than 90 and 30 words, respectively. 24% posts and 59% comments on *offmychest* are less than 90 and 30 words, respectively.

[9]https://www.reddit.com/r/CasualConversation/

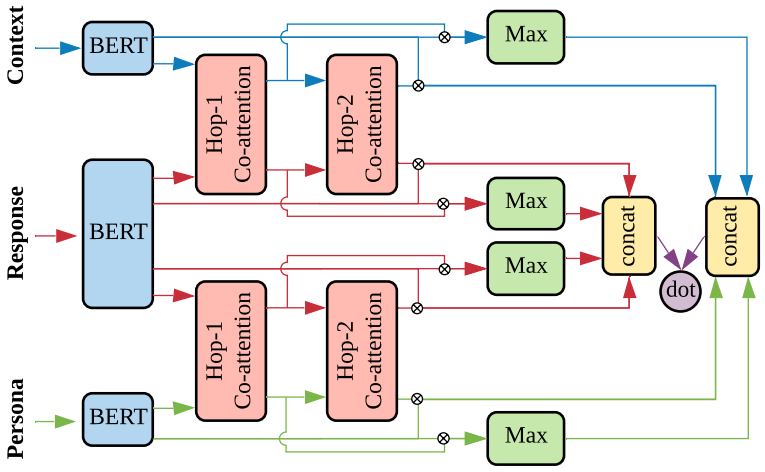


FIGURE 6.2: Our CoBERT architecture.

Comparisons with Related Datasets Table 6.4 presents the comparisons between PEC and related datasets. PEC has the unique advantage of being both persona-based and empathetic. In addition, PEC is collected from social media, resulting in a much more diverse set of speakers and language patterns than ED [2] and PC [7], which are collected from only hundreds of crowd-sourced workers. Finally, PEC is over 2x larger than the other two public datasets, allowing the exploration of larger neural models in future research.

6.4 BERT with Multi-Hop Co-Attention (CoBERT)

In this section, we briefly introduce the task of response selection and present our proposed CoBERT model, as shown in Figure 6.2.

6.4.1 Task Definition

We denote a training conversation dataset $\mathcal{D}$ as a list of N conversations in the format of (X, P, y), where $X = \{X_1, X_2, ..., X_{n_X}\}$ denotes the n_X context utterances, $P = \{P_1, P_2, ..., P_{n_P}\}$ denotes the n_P persona sentences of the respondent, and y denotes the response to X. The task of response selection can be formulated as learning a function $f(X, P, y)$ that assigns the highest score to the true candidate y and lower scores to negative candidates given X and P. During inference, the

trained model selects the response candidate with the highest score from a list of candidates.

6.4.2 BERT Representation

We use BERT [76] as our sentence encoders. Similar to the Bi-encoder [245], we concatenate context utterances as a single context sentence before passing it into BERT. Since there is no ordering among persona sentences, we concatenate randomly ordered persona sentences[10]. After passing the context, persona and response to BERT encoders, we obtain their vector representations $\mathbf{X} \in \mathbb{R}^{m \times d}$, $\mathbf{P} \in \mathbb{R}^{q \times d}$ and $\mathbf{Y} \in \mathbb{R}^{n \times d}$ from the last layer, respectively, where d denotes the embedding size of BERT, and m, q and n denote the sequence lengths of context, persona and response, respectively. Note that different segment ids are used to differentiate speaker and respondent utterances in the context.

6.4.3 Hop-1 Co-attention

Given $\mathbf{X}$ and $\mathbf{Y}$, we learn the first-order matching information using co-attention [254]. Specifically, we first compute the word-word affinity matrix $\mathbf{A_{XY}} \in \mathbb{R}^{m \times n}$:

$$\mathbf{A_{XY}} = \mathbf{X}\mathbf{Y}^T. \tag{6.1}$$

Then the context-to-response attention $\mathbf{A_{X2Y}} \in \mathbb{R}^{m \times n}$ and the response-to-context attention $\mathbf{A_{Y2X}} \in \mathbb{R}^{n \times m}$ can be computed as follows:

$$\mathbf{A_{X2Y}} = \mathit{softmax}(\mathbf{A_{XY}}), \tag{6.2}$$

$$\mathbf{A_{Y2X}} = \mathit{softmax}(\mathbf{A}^T_{\mathbf{XY}}), \tag{6.3}$$

where *softmax* denotes the softmax function along the second dimension. Finally, we obtain the attended context representation $\mathbf{X}' = \mathbf{A_{X2Y}}\mathbf{Y} \in \mathbb{R}^{m \times d}$ and response representation $\mathbf{Y}'_{\mathbf{X}} = \mathbf{A_{Y2X}}\mathbf{X} \in \mathbb{R}^{n \times d}$.

[10]Reusing the same positional information for all persona sentences [127] to model position invariance produces worse performance in our preliminary experiments.

To aggregate the first-order matching information and extract discriminative features, we apply max-pooling to $\mathbf{X}^{'}$ and $\mathbf{Y}^{'}_{\mathbf{X}}$ along the sequence dimension and obtain $\mathbf{X}^{'}_{max} \in \mathbb{R}^d$ and $\mathbf{Y}^{'}_{\mathbf{X},max} \in \mathbb{R}^d$.

6.4.4 Hop-2 Co-attention

We propose the hop-2 co-attention to learn second-order interactive matching. Different from the attention-over-attention for reading comprehension [255], our method learns bidirectional matching for response selection. Specifically, we apply attention over the attention matrices:

$$\mathbf{A_X}^{'} = mean(\mathbf{A_{X2Y}})\mathbf{A_{Y2X}}, \tag{6.4}$$

$$\mathbf{A_Y}^{'} = mean(\mathbf{A_{Y2X}})\mathbf{A_{X2Y}}, \tag{6.5}$$

where $\mathbf{A_X}^{'} \in \mathbb{R}^{1\times m}$ and $\mathbf{A_Y}^{'} \in \mathbb{R}^{1\times n}$ denote the second-order attention over $\mathbf{X}$ and $\mathbf{Y}$, respectively, and *mean* denotes mean pooling along the first dimension. Then we obtain the attended context representation $\mathbf{X}^{''} = \mathbf{A_X}^{'}\mathbf{X} \in \mathbb{R}^d$ and response representation $\mathbf{Y}^{''}_{\mathbf{X}} = \mathbf{A_Y}^{'}\mathbf{Y} \in \mathbb{R}^d$.

We apply the same procedure to match $\mathbf{P}$ and $\mathbf{Y}$, and obtain the first-order matching information $\mathbf{P}^{'}_{max} \in \mathbb{R}^d$ and $\mathbf{Y}^{'}_{\mathbf{P},max} \in \mathbb{R}^d$, and the second-order matching information $\mathbf{P}^{''} \in \mathbb{R}^d$ and $\mathbf{Y}^{''}_{\mathbf{P}} \in \mathbb{R}^d$.

Intuitively, our hop-1 co-attention learns attended representations for $\mathbf{X}$ and $\mathbf{Y}$, and our hop-2 co-attention learns "truly" attended representations for $\mathbf{X}$ and $\mathbf{Y}$ where the weights are computed from attentions over attentions.

6.4.5 Loss

We obtain the final context representation $\mathbf{X}_f = [\mathbf{X}^{'}_{max}; \mathbf{X}^{''}; \mathbf{P}^{'}_{max}; \mathbf{P}^{''}] \in \mathbb{R}^{4d}$ and the final response representation $\mathbf{Y}_f = [\mathbf{Y}^{'}_{\mathbf{X},max}; \mathbf{Y}^{''}_{\mathbf{X}}; \mathbf{Y}^{'}_{\mathbf{P},max}; \mathbf{Y}^{''}_{\mathbf{P}}] \in \mathbb{R}^{4d}$, where $[;]$ denotes concatenation. Then we use dot product to compute the final matching score:

$$f(X, P, y) = dot(\mathbf{X}_f, \mathbf{Y}_f). \tag{6.6}$$

We optimize our model by minimizing the cross-entropy loss for selecting the true candidate from a list of candidates. Formally, the loss Φ is computed as follows:

$$\Phi = \sum_{(X,P,y)\sim\mathcal{D}} -\frac{e^{f(X,P,y)}}{\sum_{\hat{y}\sim\mathcal{N}(X)\cup\{y\}} e^{f(X,P,\hat{y})}}, \tag{6.7}$$

where $\mathcal{N}(X)$ denotes a set of randomly sampled negative candidates for the context X.

6.5 Experiments

In this section, we present the datasets, evaluation metrics, baselines, model settings, experimental results and analysis.

6.5.1 Datasets and Evaluation Metrics

We evaluate models on PEC and its two sub-domains, i.e., happy and offmychest. The training, validation, and test splits of PEC are combined from the corresponding splits from happy and offmychest. The dataset statistics are shown in Table 5.3.

Following [93, 245, 249], we evaluate models using Recall@k where each test example has C possible candidates to select from, abbreviated to R@k, as well as mean reciprocal rank (MRR). In our experiments, we set $C = 100$ and $k = 1, 10, 50$. The candidate set for each test example includes the true response and other $C - 1$ randomly sampled responses from the test set.

6.5.2 Baselines and Model Settings

We compare CoBERT with several competitive baselines. Note that the BoW, HLSTM [19], and Bi-encoder [245] baselines share the same Tri-encoder architecture, where the final matching score is the dot product between the average of context and persona representations and the response representation.

BoW: The context, persona, and response encoders compute the averaged word embedding.

HLSTM [19]: The context encoder has an utterance-level BiLSTM and a context-level BiLSTM. All three encoders share the same utterance-level BiLSTM. We average separately encoded persona sentence representations to obtain the final persona representation.

DIM [249]: A state-of-the-art non-pretraiend model for persona-based response selection. DIM adopts finer-grained matching and hierarchical aggregation to learn rich matching representation.

Bi-encoder [245]: A state-of-the-art BERT-based model for empathetic response selection [2].

Poly-encoder [245]: A state-of-the-art BERT-based model for response selection. Poly-encoder learns latent attention codes for finer-grained matching. Note that we do not consider Cross-encoder [245] as an appropriate baseline because it performs two orders of magnitude slower than Poly-encoder in inference, rendering it intractable for real-time applications.

We use fastText [256] embeddings of size 300 to initialize BoW and HLSTM. We follow the released code[11] to implement DIM. For all BERT-based models, we use the base version of BERT and share parameters across all three encoders[12]. We use 128 context codes for Poly-encoder[13]. We optimize all BERT-based models using Adam [163] with a batch size of 64 and a learning rate of 0.00002. The positive to negative candidates ratio during training is set to 1:15. We use a maximum of $n_X = 6$ contextual utterances and a maximum of $n_P = 10$ persona sentences for each conversation. We conduct all experiments on NVIDIA V100 32GB GPUs in mixed precision.

6.5.3 Comparison with Baselines

Table 6.5 and Table 6.6 present the test results of CoBERT and all baselines on PEC and its two sub-domains. Among the non-pretrained models, DIM outperforms BoW and HLSTM by large margins on all datasets, demonstrating the importance

[11] https://github.com/JasonForJoy/DIM

[12] A shared BERT encoder obtained better performance than separate encoders in our preliminary experiments.

[13] More context codes result in memory error in our experiments. According to [245], more context codes only lead to marginally better results.

TABLE 6.5: Test performance (in %) of CoBERT and all baselines on happy and offmychest. Values in bold denote best results.

	happy				**offmychest**			
Models	R@1	R@10	R@50	MRR	R@1	R@10	R@50	MRR
BoW	10.2	45.6	85.2	21.8	13.9	51.6	87.1	26.2
HLSTM	15.7	53.6	91.6	28.1	17.6	55.7	91.8	30.2
DIM	31.3	67.0	95.5	43.0	40.6	72.6	96.4	51.2
Bi-encoder	32.4	71.3	96.5	45.1	42.4	78.4	97.6	54.5
Poly-encoder	33.7	72.1	96.7	46.4	43.4	79.3	97.7	55.3
CoBERT (ours)	**36.2**	**73.0**	**96.9**	**48.4**	**47.0**	**79.7**	**97.8**	**58.0**

TABLE 6.6: Test performance (in %) of CoBERT and all baselines on PEC. Values in bold denote best results.

	PEC (happy + offmychest)			
Models	R@1	R@10	R@50	MRR
BoW	15.4	52.9	86.7	27.4
HLSTM	22.2	63.0	94.8	35.2
DIM	39.3	74.6	97.3	50.5
Bi-encoder	42.3	79.2	98.1	54.4
Poly-encoder	43.0	79.8	98.2	55.2
CoBERT (ours)	**45.1**	**80.5**	**98.3**	**56.7**

TABLE 6.7: Transfer test of CoBERT in R@1 (in %).

Train \ Test	happy	offmychest	PEC
happy	36.2	41.2	40.5
offmychest	28.8	47.0	38.4
PEC	37.0	47.5	45.1

of finer-grained matching and hierarchical aggregation for response selection. The simple Bi-encoder performs noticeably better than DIM, suggesting that sentence representation is another critical factor in response selection and that BERT can provide much richer representation than the BiLSTM used in DIM. Poly-encoder performs best among all baselines because it leverages the strengths of both BERT and attention-based finer-grained matching.

Our CoBERT consistently outperforms all baselines on all datasets with large margins, including the state-of-the-art Poly-encoder. The performance gain is primarily attributed to our multi-hop co-attention, which learns higher-order bidirectional word-word matching between context and response, whereas Poly-encoder only learns the first-order unidirectional attention from response to context using latent attention codes. Efficiency-wise, CoBERT has slightly longer inference time

(1.50x) but requires much less memory usage (0.62x) than Poly-encoder, as shown in Table 6.8.

We further investigate how well our CoBERT can generalize across different domains, as reported in Table 6.7. In general, in-domain test results are better than out-of-domain test results. The transfer performance from happy to offmychest (41.2%) and vice versa (28.8%) are comparable to the in-domain performance of DIM (40.6% on offmychest and 31.3% on happy), suggesting that our CoBERT can generalize well across empathetic conversations in contrasting sentiments.

6.5.4 Comparison with BERT-adapted Models

To perform a more comprehensive evaluation of CoBERT, we further compare CoBERT with several competitive BERT-adapted models where the sentence encoders are replaced by BERT. We report the results in the middle section of Table 6.8.

BERT + MemNet [7]: MemNet incorporates persona into context using a Memory Network [257] with residual connections. The BERT+MemNet model performs slightly worse than Bi-encoder and much worse than our CoBERT, although it achieves slightly faster inference than Bi-encoder.

BERT+DAM [93]: DAM aggregates multi-granularity matching using convolutional layers. The BERT+DAM model performs significantly better than Bi-encoder in R@1, demonstrating the usefulness of learning n-gram matching over the word-word matching matrices. Nevertheless, CoBERT performs noticeably better and has faster inference (7.13x) than BERT+DAM.

BERT+DIM [249]: The BERT+DIM model combines the benefits from both the strong sentence representation of BERT and the rich finer-grained matching of DIM. However, BERT+DIM performs slightly worse than CoBERT, suggesting that the more complex matching and aggregation methods in DIM do not lead to performance improvement over our multi-hop co-attention. In addition, our CoBERT is substantially faster (9.18x) than BERT+DIM in inference, thus more practical in real-world applications.

TABLE 6.8: Validation performance (in %), inference time, and memory usage (RAM) for baselines, BERT-adapted models and ablation studies on PEC. Inference time and RAM are relative to the Bi-encoder.

Model	R@1	MRR	Inference Time	RAM
Baselines				
DIM	40.3	51.6	10.36x	**0.79x**
Bi-encoder	42.6	55.2	1.00x	1.00x
Poly-encoder	43.3	55.7	1.33x	1.84x
BERT-adapted Models				
BERT+MemNet	42.3	53.8	**0.87x**	0.89x
BERT+DAM	45.0	56.9	14.26x	1.57x
BERT+DIM	46.1	57.7	18.36x	1.78x
Ablations				
CoBERT (ours)	**46.2**	**57.9**	2.00x	1.14x
- hop-1	44.0	56.2	1.65x	1.11x
- hop-2	45.5	57.1	1.76x	1.11x
+ hop-3	46.0	57.6	2.70x	1.13x
- max + mean	44.1	56.3	2.12x	1.13x
+ mean	46.1	57.8	2.71x	1.15x

6.5.5 Ablation Study

We conduct ablation studies for CoBERT, as reported in the bottom section of Table 6.8. Removing either hop-1 or hop-2 co-attention results in noticeably worse performance, albeit slightly faster inference. Removing hop-1 leads to larger performance drop than removing hop-2, suggesting that the first-order matching information seems more important than the second-order matching information for response selection. An additional hop-3 co-attention results in slightly worse performance, suggesting that our two-hop co-attention is the sweet spot for model complexity.

Replacing the max pooling in the hop-1 co-attention by mean pooling leads to much worse performance. In addition, concatenating the results from both max and mean pooling slightly degrades performance, as well as inference speed, suggesting that max pooling may be essential for extracting discriminative matching information.

6.5.6 Human Evaluation

We conduct human evaluation to investigate the impact of persona and empathy on response quality. Various models are trained and tested using different training

TABLE 6.9: Human evaluation. P.Con. measures the persona consistency of the response. Empathy measures the empathy level of the response. Overall measures the overall quality of the response. All three metrics, i.e., P.Con., Empathy, and Overall are rated in the scale of [1, 5], higher is better.

Model	Train	Test	Persona	P.Con.	Empathy	Overall
CoBERT	CASUAL	CASUAL	✓	3.14	3.26	3.52
CoBERT	CASUAL	PEC	✓	3.02	3.54	3.75
CoBERT	PEC	PEC	✓	3.18	3.87	4.03
CoBERT	PEC	PEC	✗	2.95	3.61	3.81
Poly-encoder	PEC	PEC	✓	3.17	3.66	3.87

and testing datasets with or without persona information. Each setting is evaluated using 100 responses by three annotators. The results are reported in Table 6.9.

We have the following observations ($p < 0.01$, one-tailed t-test): 1) using PEC as test candidates improves empathy and overall ratings over CASUAL; 2) training on PEC improves empathy and overall ratings over CASUAL; 3) providing CoBERT with persona improves persona consistency, empathy, and overall ratings; and 4) CoBERT outperforms Poly-encoder in empathy and overall ratings. Observations 1) and 2) validate the effectiveness of our proposed PEC dataset and show the positive impact of empathy on human ratings. Observation 3) suggests that persona improves human ratings as well. Observation 4) shows that our proposed CoBERT outperforms Poly-encoder in human evaluations.

6.5.7 Impact of Persona on Empathetic Responding

Empathetic vs. Non-empathetic

We investigate whether persona improves empathetic responding more when CoBERT is trained on empathetic conversations than non-empathetic ones. First, we introduce a non-empathetic conversation dataset as the control group, denoted as CASUAL, which is the same as the control group in Section 6.3 but much larger in size. The CASUAL dataset is collected and processed in the same way as PEC but has significantly lower empathy than PEC (see Table 6.2). The sizes of training, validation, and test splits of CASUAL are 150K, 20K, and 20K, respectively. Then, we replace a random subset of training examples from CASUAL by the same number of random training examples from PEC. We then compare the persona improvement, i.e., R@1 ($n_P = 10$) − R@1 ($n_P = 0$), on the PEC validation set and

TABLE 6.10: Validation R@1 (in %), inference time, and memory usage (RAM) on PEC against different number of persona sentences n_P.

n_P	0	1	2	5	10	20
R@1	40.4	42.0	42.8	45.1	46.2	**47.1**
Inference Time	**1.00x**	1.34x	1.38x	1.55x	1.90x	2.96x
RAM	**1.00x**	1.05x	1.06x	1.19x	1.51x	2.29x

the CASUAL validation set for different replacement ratios, where n_P denotes the number of persona sentences used in each conversation.

The results are illustrated in Figure 6.3. It is unsurprising that for both cases, i.e., $n_P = 0$ and $n_P = 10$, the validation R@1 on PEC increases, and the validation R@1 on CASUAL decreases as the ratio of PEC in the training dataset increases. We also observe that persona consistently improves performance on both validation sets for all ratios.

By investigating the widths of the two shaded regions in Figure 6.3, we find that the persona improvement on casual responding remains almost constant as more CASUAL training examples are used (3.31% when trained on all 150K PEC conversations vs. 3.44% when trained on all 150K CASUAL conversations). However, the persona improvement on empathetic responding consistently increases as more PEC training examples are used (3.77% when trained on all 150K CASUAL conversations versus 6.32% when trained on all 150K PEC conversations), showing that persona improves empathetic responding significantly more when CoBERT is trained on empathetic conversations than non-empathetic ones ($p < 0.001$, one-tailed t-test).

This result reveals an empirical link between persona and empathy in human conversations and may suggest that persona has a greater impact on empathetic conversations than non-empathetic ones. The result also shows that CoBERT can learn this link during training and use it to perform better empathetic responding during testing. One possible psychological root of this link is that persona is highly correlated to personality [38], which in turn influences empathy and empathetic responding [243]. A more detailed analysis of this empirical link, e.g., which types of personas improve more on happy than offmychest, is left for future work.

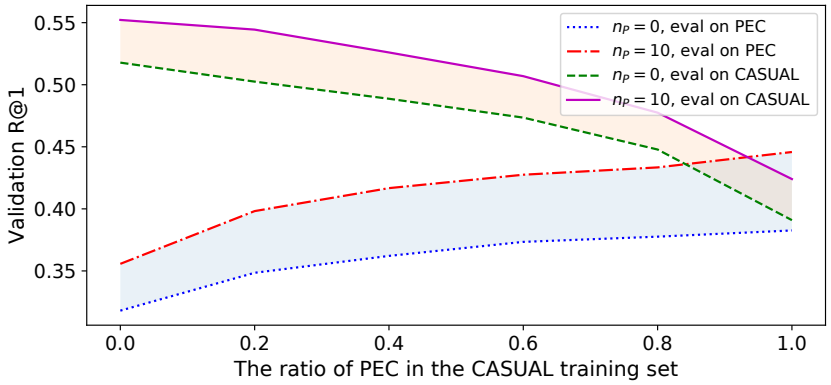


FIGURE 6.3: Validation R@1 (in %) against different ratios of PEC in the CASUAL training set.

TABLE 6.11: Test R@1 (in %) on PEC against examples with seen or unseen personas. n_P denotes the number of persona sentences.

n_P	seen (57.9%)	unseen (42.1%)	all (100%)
0	40.3	38.5	39.6
10	46.5	43.2	45.1

TABLE 6.12: Case study.

Context: I'm on a diet and lost ten pounds this month!	
Persona	**Model Response**
I am a college graduate.	Congrats! I'm trying to lose weight as well.
I work in a gym.	Good job! Doing some exercises will help you stay fit!
I am a doctor.	Congrats! Don't forget to take adequate nutrition though.

Number of Persona Sentences

We analyze the persona improvement with respect to different numbers of persona sentences n_P, as shown in Table 6.10[14]. It is clear that model performance, inference time, and memory usage all increase when more persona sentences are incorporated. Note that memory usage grows quadratically with n_P due to the self-attention operations in BERT. We chose $n_P = 10$ in our experiments because it achieves competitive performance at a reasonable cost of efficiency.

Performance on New Personas

We analyze the CoBERT performance on examples with new personas. In the PEC test set, 42.1% examples are from new speakers. The performance of CoBERT

[14]Using $n_P = 30$ results in memory error.

on test examples with seen and unseen (new) speakers is shown in Table 6.11. The results show that 1) CoBERT performs reasonably well on examples with unseen personas, suggesting that CoBERT can generalize well to unseen personas and retrieve the right response for new speakers accurately; 2) CoBERT performs worse on examples with unseen personas than seen personas; 3) leveraging personas during model training and testing improves CoBERT on examples with either seen or unseen personas; and 4) the persona improvement is more noticeable for examples with seen personas than unseen personas.

Case Study

We conduct a case study on how persona affects empathetic responding, as shown in Table 6.12. The model responses are selected by CoBERT from 1K candidates. It is clear that given the same context, different personas lead to different persona-based empathetic responses. For example, when the persona is "I am a doctor.", the model response expresses both praises and caring about the speaker's health.

6.6 Summary

In this chapter, we present a new task and a large-scale multi-domain dataset, PEC, towards persona-based empathetic conversations. We then propose CoBERT, an effective and efficient model that obtains substantially better performance than competitive baselines on PEC, including the state-of-the-art Poly-encoder and several BERT-adapted models. CoBERT is free from hyper-parameter tuning and universally applicable to the task of response selection in any domain. Finally, we investigate the impact of persona and empathy on response quality and human ratings, and present the first empirical study on the impact of persona on empathetic responding. The results show that both persona and empathy are beneficial to response quality and human ratings. In addition, the results reveal an empirical link between persona and empathy in human conversations and may suggest that persona has a greater impact on empathetic conversations than non-empathetic ones.

Chapter 7

Humanized Conversational Agents

7.1 Overview

In the previous chapters, we presented studies on endowing open-domain CAs with one or two human traits. In this chapter, we take a step further and propose a **H**umanized open-domain **CA** (**HCA**) that possesses all the following proposed human traits simultaneously: emotion, commonsense, persona, and empathy. Such a unified CA resembles humans more closely than CAs with only one or two human traits. In addition, the unified CA is more versatile because it can work in scenarios where multiple human traits are desired.

7.2 Related Work

Few studies investigated combining several human traits into a single CA until recently. Smith et al. [258] hypothesize that a good CA should seamlessly blend all human traits/skills into one cohesive conversation flow. They proposed a new conversation dataset called blended skill talk (BST), where speakers use various human skills, e.g., persona, knowledge, and empathy, to converse with each other. They further concluded that using MTL over several human skills results in better blended conversation performance than models trained on a single skill. Later,

Roller et al. [106] extended [258] to a larger scale and discussed recipes and challenges towards building a truly human-level CA.

Several related studies focus on improve open-domain CA without explicitly incorporating human traits. Adiwardana et al. [105] proposed Meena, a large Transformer-based generative CA trained on 867M Reddit conversations. They also proposed a new human evaluation metric SSA, which is found to correlate well with perplexity. They concluded that perplexity over a large evaluation dataset is a good automatic metric for generative CAs. Recently, Bao et al. [259] proposed PLATO-2, an open-domain CA trained on conversations from social media using curriculum learning. Specifically, in the first stage, a simple one-to-one response generation model is trained; in the second stage, a diverse response generation model with latent variables and a coherence estimation model are trained to produce diverse yet coherent responses. PLATO-2 has been shown to perform better than Meena and Blender in human evaluations.

7.3 Humanized Conversational Agents (HCA)

Inspired by the recent success of pretrained language models, e.g., BERT [76], GPT-3 [260], Meena [105], and Blender [106], we follow the same pretrain-and-finetune paradigm for our HCA. Given that retrieval-based models are arguably more robust to out-of-distribution and adversarial inputs than generation-based models, we design HCA as a retrieval-based model (see Section 2.2.2).

7.3.1 Pretraining

We follow the pretraining strategies in [245] to pretrain the Transformer-based encoders in HCA on a large collection of conversations. Pretraining on conversation data has been shown to achieve better downstream performance on response retrieval than pretraining on conventional text corpus, e.g., Wikipedia and Toronto Books [245]. Specifically, the encoders in HCA follow the BERT-base architecture [76]. The pretraining involves a masked language model (MLM) task [76] and a next-utterance-prediction (NUP) task [245]. MLM is a fill-in-the-blank task, where the model uses surrounding context words to predict the masked word. NUP is an

utterance pair prediction task, where the model predicts whether the second utterance follows the first utterance. During pretraining, 50% of the time the second utterance in the NUP task is the correct next utterance and 50% of the time the second utterance is randomly sampled from the dataset. We alternate batches of MLM and NUP during pretraining.

7.3.2 Finetuning

Following [106, 258], we finetune our pretrained HCA using multi-task learning (MTL). Different from conventional MTL where multiple task objectives are optimized, HCA is optimized for the single task of response retrieval (see Equation 6.7) but on multiple finetuning datasets that exhibit different human traits. In other words, HCA is optimized to produce responses that can possess multiple human traits.

Our HCA is inspired by the BST model [258] and the blender [106]; however, HCA has two major differences from them: 1) HCA is finetuned using our proposed multi-hop co-attention (see Section 6.4.3 and 6.4.4) for finer-grained response matching; and 2) we use an improved version of the dynamic sampling strategy for MTL (DS-MTL) [261] for better sampling of training instances across multiple datasets in MTL. Empirical analysis shows that these differences lead to substantial performance improvement over baselines (see Section 7.4).

DS-MTL

We propose an improved version of the dynamic sampling strategy [261] for MTL. Conventional data sampling in MTL is often uniform or by size. **Uniform Sampling (US)** allows the model to train on a roughly equal number of training examples from each dataset per epoch. **Sampling by Size (SS)** allows the model to train on more examples from larger datasets per epoch. The former approach, i.e., US, often underfits large datasets, whereas the latter approach, i.e., SS, often leads to catastrophic forgetting on small datasets [261]. The **DS-MTL** proposed in [261] dynamically samples data from each dataset according to their corresponding normalized performance gap with single-task performance[1]. In other words,

[1]In our experiments, we use R@1 as the primary performance metric.

a larger performance gap on a dataset leads to more training examples from the dataset in future epochs. DS-MTL aims to achieve single-task performance for each task in a multi-task setting and has been empirically shown to perform better than conventional data sampling methods, i.e., US and SS, in multi-task reading comprehension [261].

However, in our preliminary experiments, we observed that DS-MTL leads to worse performance than SS on our datasets. The major reason is that DS-MTL is not suitable for tasks where the multi-task performance of a task tends to exceed its single-task performance, which is exactly our case. Specifically, in our preliminary experiments, DS-MTL stopped sampling the training data of a task early because the performance on its validation set exceeded its single-task performance quickly after the start of training. Consequently, the performance on the task could not improve further even if it has such potential. In our approach, to push the multi-task performance of HCA further, we heuristically redefine the performance gap as the gap with single-task performance or the gap with multi-task performance using SS, whichever is larger. Intuitively, a larger performance gap on a task indicates a larger potential for improvement on the task, and our model can sample more data from the task to realize the potential. We term the improved DS-MTL as **iDS-MTL** and investigate its impact in Section 7.4.4.

7.4 Experiments

In this section, we present the datasets, evaluation metrics, baselines, model settings, experimental results and analysis.

7.4.1 Datasets and Evaluation Metrics

We pretrain HCA on 174M English conversations from Reddit [245]. We use Byte Pair Encoding (BPE) [262] with a vocabulary size of 8K to tokenize all text data.

We finetune HCA on the following four English datasets that exhibit human traits:

ConvAI2 [7, 263]: ConvAI2 is a variant of the PERSONA-CHAT dataset [7] used in the NeurIPS 2018 ConvAI2 competition[2]. Each speaker in ConvAI2 is associated with a persona[3]. Speakers are encouraged to getting to know each other during conversations. ConvAI2 has 140K utterances with over 1K personas.

WoW [123]: Conversations in WoW involve in-depth discussions about given topics. Speakers can rely on the provided topic-related documents to chat with each other. WoW has 194K examples with over 1,250 topics.

ED [2]: Conversations in ED are empathetic. In each conversation, there is a speaker expressing certain emotions and a listener responding with empathy. HCA is trained in the role of the listener with the aim of learning empathetic responding. ED has 50k utterances.

BST [258]: BST aims to blend persona, knowledge, and empathy exhibited in the aforementioned three datasets. Each utterance in BST may be persona-based, knowledge-based, or empathetic, depending on the context. BST has 76k utterances.

To facilitate MTL, we create a unified input representation for all finetuning datasets, as illustrated in Figure 7.1. All contexts are provided with a topic and several persona sentences selected from the training sets of WoW and ConvAI2, respectively. Note that ConvAI2 does not have topic information, WoW does not have persona information, and ED has neither. Hence, we follow [258] to automatically fill in the missing information[4]. In our experiments, we found that randomly dropping topic or persona information during training improves the robustness of our model, which is similar to the knowledge drop (**KD**) in [123]. Hence, we adopt KD in our experiment and use KD=p to denote knowledge drop with probability p. We investigate the impact of KD in Section 7.4.4.

[2]http://convai.io/

[3]Each persona usually comprises several persona sentences. For example, "I like dog." is a persona sentence.

[4]Each conversation in ConvAI will be paired with a relevant topic. Each conversation in WoW will be paired with a relevant persona. Each conversation in ED will be paired with a relevant topic and a relevant persona

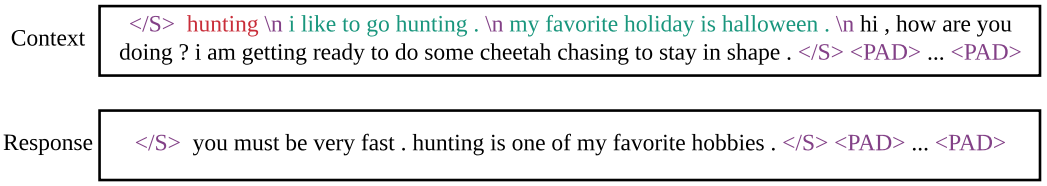


FIGURE 7.1: Context and response representations for finetuning HCA. Text in purple denotes special tokens. Text in red denotes a topic from WoW. Text in green denotes persona sentences from ConvAI2. Text in black denotes utterances. Each context is truncated and padded to 256 tokens. Each response is truncated and padded to 32 tokens.

For each finetuning dataset, we evaluate HCA on two versions of its validation dataset[5], namely the **original** version and the **complete** version with missing information filled, i.e., missing topic in ConvAi2 or missing persona in WoW or both in ED. We additionally evaluate HCA on three datasets: **mixed**, **mixed_complete** and **mixed_all**. The mixed dataset mixes candidates from all four original evaluation sets. The mixed_complete dataset mixes candidates from all four complete evaluation sets. The mixed_all dataset mixes candidates from all four original and four complete evaluation sets. We regard the mixed_all as the most important validation dataset because it tests model performance in scenarios where both partial information and complete information are available. Our evaluation datasets are comprehensive and cover a wide range of application scenarios with different availabilities of topic and persona information and different sets of candidates.

Following [106, 258], we evaluate models using Recall@1, where each test example has C possible candidates to select from, abbreviated to R@1. In our experiments, we set $C = 100$ for all datasets[6]. The candidate set for each test example includes the true response and other $C - 1$ randomly sampled responses from the test set.

7.4.2 Baselines and Model Settings

We compare HCA with two competitive baselines: Bi-encoder and Poly-encoder [245]. Both baselines use BERT as encoders. Bi-encoder matches context and response using dot product of their vector representations. Poly-encoder learns

[5]Since ConvAI2 only has a validation dataset and no public test datasets and we do not perform any hyper-parameter tuning, we report all results on the validation datasets for consistency.

[6]Note that in the literature of the ConvAI2 dataset [7, 106, 258, 263], C is set to 20. However, here we use $C = 100$ for a consistent evaluation with other datasets.

TABLE 7.1: Validation R@1 (in %) of HCA and all baselines on the **original** versions of the validation datasets. Note that the Mixed_all validation dataset includes both original and complete versions. All models use KD=0.5 and are trained with iDS-MTL in the Multi-Task setting. Values in bold denote best results.

Train \ Valid	Model	ConvAI2	WoW	ED	BST	Mixed	Mixed_all
ConvAI2	Bi-encoder	66.2	60.9	47.8	69.9	68.4	67.6
	Poly-encoder	67.0	58.0	46.4	69.8	68.9	68.2
	HCA	**73.2**	55.7	49.1	74.1	74.6	74.4
WoW	Bi-encoder	33.2	56.2	40.7	59.4	48.1	47.4
	Poly-encoder	32.7	57.4	40.9	57.6	47.5	46.6
	HCA	38.5	54.3	39.3	60.7	51.4	51.3
ED	Bi-encoder	29.5	51.6	57.6	56.4	48.2	48.0
	Poly-encoder	30.7	50.6	57.4	56.3	49.2	49.0
	HCA	30.8	43.3	58.3	55.0	49.8	49.9
BST	Bi-encoder	45.4	60.5	48.7	69.3	58.0	57.9
	Poly-encoder	45.6	60.2	48.1	70.0	58.6	58.2
	HCA	57.1	53.6	48.6	73.3	66.4	66.3
Multi-Task	Bi-encoder	65.8	60.9	58.4	74.4	73.6	73.2
	Poly-encoder	66.0	**63.1**	58.5	74.6	73.8	74.0
	HCA	71.7	61.8	**61.9**	**78.1**	**78.6**	**79.1**

latent attention codes for finer-grained matching and is the state-of-the-art model for response retrieval. Poly-encoder is the basis of the retrieval models in [106, 258]. Both Bi-encoder and Poly-encoder use the same pretrained weights as HCA.

During finetuning, we treat other in-batch responses as negative responses [245]. We finetune all models on all datasets using a batch size of 32 with gradient accumulation steps of 4[7]. Gradient accumulation allows the model to update its parameters using gradients from more examples than the GPU memory allows. The learning rate is set to 0.00005. All experiments are conducted on NVIDIA V100 32GB GPUs in mixed precision.

7.4.3 Performance Comparisons

We present the performance of all models on the original and the complete validation datasets in Table 7.1 and Table 7.2, respectively.

[7]Note that this is not equivalent to a batch size of 128 because the model is still trained to select the groundtruth response from 32 candidates.

TABLE 7.2: Validation R@1 (in %) of HCA and all baselines on the **complete** versions of the validation datasets. All models use KD=0.5 and are trained with iDS-MTL in the Multi-Task setting. Values in bold denote best results.

Train \ Valid	Model	ConvAI2	WoW	ED	BST	Mixed
ConvAI2	Bi-encoder	66.3	55.6	42.6	69.9	67.0
	Poly-encoder	66.6	54.5	44.2	69.6	67.5
	HCA	**73.1**	56.2	51.4	73.5	75.0
WoW	Bi-encoder	32.7	56.7	38.3	58.5	46.6
	Poly-encoder	32.6	58.2	38.3	57.7	46.1
	HCA	38.1	54.9	41.2	60.0	50.9
ED	Bi-encoder	29.7	49.6	58.4	55.8	48.2
	Poly-encoder	30.6	49.9	59.4	56.5	49.3
	HCA	30.9	42.6	67.4	54.6	50.9
BST	Bi-encoder	44.5	59.2	47.5	69.8	57.4
	Poly-encoder	45.0	60.7	47.8	69.8	58.2
	HCA	56.7	54.7	51.1	73.1	66.3
Multi-Task	Bi-encoder	65.3	61.5	59.4	74.3	73.5
	Poly-encoder	65.3	**61.8**	60.3	74.4	73.8
	HCA	71.5	60.2	**67.7**	**77.9**	**79.3**

Generally, for all models, the performance on the original validation datasets is similar to that on the complete validation datasets, except for ED. On ED, both bi-encoder and Poly-encoder often perform worse on the complete version of ED than the original version of ED, whereas our HCA is the opposite. In addition, the performance of all models on the three mixed datasets, i.e., mixed, mixed_complete, and mixed_all, are similar as well.

For single-task performance, all models generally perform better when trained on in-domain data than out-domain data. Comparing the single-task performance on the three mixed evaluation datasets across different training datasets, we observe that finetuning on ConvAI2 leads to the best performance, followed by BST. Possible reasons are that 1) ConvAI2 has the largest number of training examples; and 2) BST is a dataset with mixed traits, which is similar to the mixed validation datasets.

The multi-task performance of all models is generally better than their single-task performance, demonstrating the synergy of the four training datasets towards a unified CA.

Comparing different models in both single-task and multi-task settings, we observe that Poly-encoder slightly outperforms bi-encoder, and our HCA performs the best

TABLE 7.3: Multi-task R@1 (in %) of HCA and its variants on the **original** validation datasets. Note that the Mixed_all evaluation set includes both original and complete versions. KD=p denotes knowledge drop with probability p. SS denotes sampling by size. Values in bold denote best results.

Model	ConvAI2	WoW	ED	BST	Mixed	Mixed_all
HCA (KD=0, iDS-MTL)	71.5	60.0	57.3	77.8	77.8	78.5
HCA (KD=0.25, iDS-MTL)	71.9	**61.9**	60.1	77.9	78.5	78.9
HCA (KD=0.5, iDS-MTL)	71.7	61.8	61.9	**78.1**	78.6	**79.2**
HCA (KD=0.75, iDS-MTL)	71.5	61.5	**62.8**	77.8	78.2	78.4
HCA (KD=0.5, SS)	**73.7**	57.4	58.8	77.3	**78.7**	78.9
HCA (KD=0.5, DS-MTL)	72.7	61.6	60.1	76.5	77.9	78.4

TABLE 7.4: Multi-task R@1 (in %) of HCA and its variants on the **complete** validation datasets. KD=p denotes knowledge drop with probability p. SS denotes sampling by size. Values in bold denote best results.

Model	ConvAI2	WoW	ED	BST	Mixed
HCA (KD=0, iDS-MTL)	71.8	**61.2**	**70.5**	77.4	**79.7**
HCA (KD=0.25, iDS-MTL)	71.4	60.8	69.3	77.4	79.5
HCA (KD=0.5, iDS-MTL)	71.5	60.2	67.7	**77.9**	79.3
HCA (KD=0.75, iDS-MTL)	71.2	60.5	63.5	77.8	78.8
HCA (KD=0.5, SS)	**73.6**	55.3	60.3	77.8	79.1
HCA (KD=0.5, DS-MTL)	72.2	60.0	65.0	75.9	78.6

on all validation datasets except WoW. In particular, our HCA outperforms the state-of-the-art Poly-encoder by large margins, e.g., 74.0% of R@1 for Poly-encoder versus 79.1% of R@1 for HCA on mixed_all in the multi-task setting. Our results are consistent with the results in Table 6.6, validating that our multi-hop co-attention captures more accurate matching information than the latent attention codes in Poly-encoder.

7.4.4 Model Analysis

We analyze the impact of knowledge drop (KD) and data sampling strategies on model performance, as presented in Table 7.3 and Table 7.4.

Comparing different values of KD in the top four rows of both tables, we observe that KD=0.5 performs best on the original validation datasets, and KD=0 performs best on the complete validation datasets. One important finding is that KD improves model performance when only a subset of persona/topic information is available during testing. We choose KD=0.5 as the best hyper-parameter value

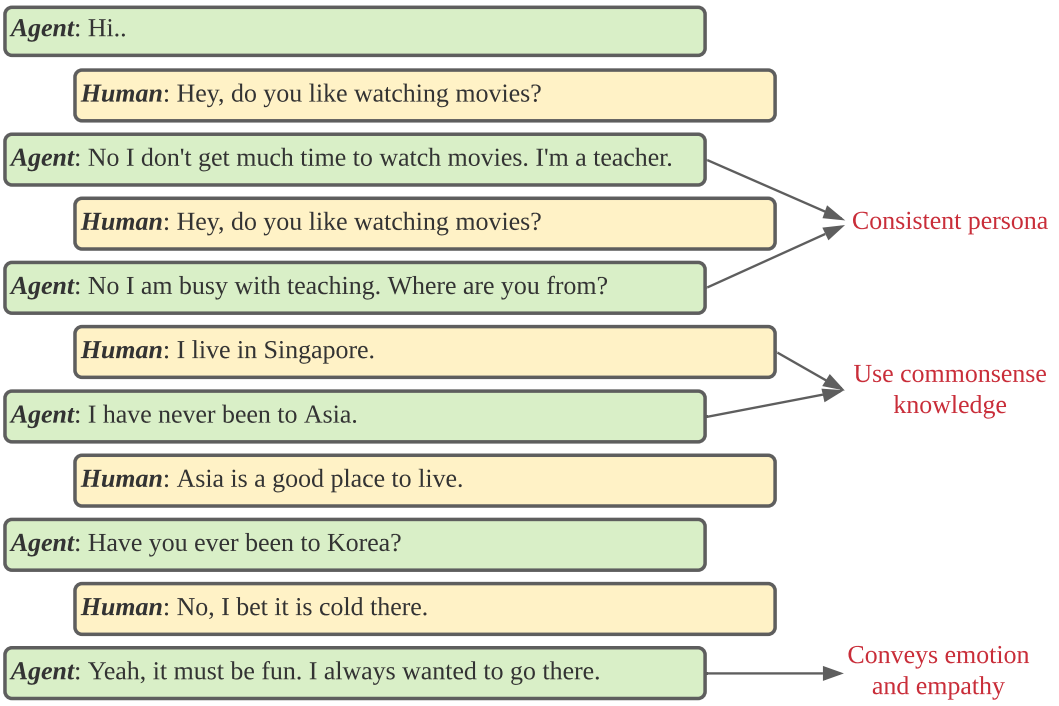


FIGURE 7.2: An example conversation between a human and our HCA.

because it performs best on the mixed_all validation dataset, which is regarded as the most comprehensive and important validation dataset.

Comparing SS and DS-MTL in the bottom two rows of both tables, we observe that SS performs noticeably worse than DS-MTL on WoW and ED and DS-MTL achieves more balanced performance across all validation datasets. However, DS-MTL fails to outperform SS on mixed_all in our experiments, suggesting the limit of DS-MTL for our tasks. In contrast, our proposed iDS-MTL performs best on mixed_all, showing that our redefined performance gap in iDS-MTL is more suitable than that in DS-MTL for our tasks, where the multi-task performance of a task tends to exceed its single-task performance.

7.4.5 Case Study

Figure 7.2 shows an example conversation between a human and our HCA. First, HCA shows consistent persona of being a teacher in its second and third responses. Second, HCA is able to leverage commonsense knowledge to infer that Singapore is part of Asia in its forth response. Third, HCA can produce emotional responses and show empathy by acknowledging the user in its last response. Overall, HCA

demonstrates multiple human traits and produces consistent, informative and empathetic responses.

7.5 Summary

In this chapter, we propose HCA, a humanized conversational agent that possesses several human traits simultaneously: emotion, commonsense, persona, and empathy. HCA is a retrieval-based model trained by the pretrain-and-finetune paradigm. Specifically, the encoders in HCA follow the BERT-base architecture and are pretrained on a large collection of generic conversations. HCA is further finetuned on multiple conversation datasets with human traits in an MTL setting. Different from prior work, our HCA is finetuned using our proposed multi-hop co-attention mechanism and the improved dynamic sampling strategy for MTL (iDS-MTL).

We evaluate HCA and baselines on various validation datasets with different availabilities of persona and topic information and different sets of candidates. The experimental results show that the multi-task performance of HCA is often better than its single-task performance, indicating that these human traits can make complementary positive impact on model performance. In addition, the model analysis shows that our proposed multi-hop co-attention mechanism and iDS-MTL lead to substantial performance improvement over baselines. Furthermore, HCA is shown to outperform the state-of-the-art models for response retrieval on multiple evaluation datasets. Finally, our case study shows that HCA can demonstrate multiple human traits and produce consistent, informative, and empathetic responses.

Chapter 8

Conclusion

In this thesis, we studied the problem of building humanized open-domain CAs by endowing them with desirable human traits: emotion, commonsense, persona, and empathy. Specifically, we first investigated endowing CAs with emotional intelligence to generate emotional responses and recognize emotions in conversations. After observing the importance of commonsense in conversational emotion recognition, we then proposed to combine emotion and commonsense into a single CA to generate commonsense-aware emotional responses. To incorporate more desirable human traits into CAs, we further proposed to augment persona and empathy into CAs because we observed implicit links between persona and empathy in our preliminary analysis of empathetic conversations. Finally, we proposed HCA to endow the CAs with all aforementioned human traits using multi-task learning.

The models proposed in this thesis advanced the state-of-the-art for several benchmarks in open-domain conversation generation and conversational emotion recognition in terms of both automatic metrics, e.g., perplexity, diversity and F1, and human evaluations, e.g., fluency, content quality, emotion quality, persona consistency, and empathy. In addition, extensive experimental results show that emotion, commonsense, persona, and empathy all have positive impact on the response quality and human ratings of open-domain CAs, indicating that these human traits are beneficial to more engaging human-agent interaction and desirable for truly human-level open-domain CAs. Finally, endowing our proposed HCA with all human traits has been shown to have complementary positive impact on model performance, and

case studies show that HCA can demonstrate multiple human traits and produce consistent, informative, and empathetic responses.

In the following sections, we start by summarizing the contributions of each chapter, followed by a discussion of future work.

8.1 Contributions

Emotional Response Generation: In Chapter 3, we proposed an emotional response generation model (AR-S2S) that can generate natural and emotional responses without language fluency degradation. Specifically, we incorporate psycholinguistic knowledge of words and considers negators and intensifiers in emotional response generation. We further propose a novel affective loss function to encourage the generation of emotional words. Experimental results show that our proposed methods are effective in improving response quality, and AR-S2S outperforms the state-of-the-art models in terms of both content and emotion qualities in human evaluations. In addition, our study suggests that incorporating emotion into open-domain CAs has a positive impact on response quality and human ratings.

Conversational Emotion Recognition: In Chapter 4, we proposed a knowledge-enriched transformer (KET) to recognize emotions in human conversations. Specifically, we model the hierarchical structure of conversation using a hierarchical self-attention mechanism and incorporate commonsense knowledge to enrich the learned representation. Experimental results show that our proposed methods improve emotion recognition substantially, and KET achieves competitive performance on multiple benchmark datasets of varying sizes and domains. In addition, our study shows that both context and commonsense knowledge are helpful for conversational emotion recognition.

Commonsense-Aware Emotional Response Generation: In Chapter 5, we combined emotion and commonsense into a single response generation model (CARE). Specifically, we first proposed a novel knowledge graph based approach to extract commonsense-aware emotional latent concepts. We then proposed three methods to collaboratively incorporate the latent concepts during model training and

inference. Experimental results show that our proposed methods outperform alternative approaches, and CARE can produce more accurate and commonsense-aware emotional responses and achieve better human ratings than state-of-the-art commonsense-aware models and emotional models. Our results also validate the positive impact of combining commonsense and emotion on open-domain CAs.

Persona-Based Empathetic Conversational Model: In Chapter 6, we proposed to endow open-domain CAs with persona and empathy. Specifically, we proposed a new task and a new dataset (PEC) towards persona-based empathetic conversations. We further proposed a multi-hop co-attention method for effective and efficient response retrieval (CoBERT). Experimental results show that our proposed PEC dataset leads to better response quality and human ratings than casual conversations, and CoBERT outperforms state-of-the-art baselines on response retrieval. In addition, our study suggests that endowing open-domain CAs with persona and empathy improves response quality, and persona has a larger impact on empathetic conversations than non-empathetic ones.

Humanized Conversational Agents: In Chapter 7, we proposed to endow open-domain conversational agents with multiple human traits simultaneously: emotion, commonsense, persona, and empathy. We follow a pretrain-and-finetune paradigm to build a retrieval-based humanized conversational agent (HCA). Specifically, we pretrain the encoders of HCA on a large collection of diverse conversations and then finetune HCA on multiple small datasets with human traits using multi-task learning (MTL). The response matching module during finetuning is based on the multi-hop co-attention method from CoBERT in Chapter 6. We further propose an improved dynamic sampling strategy for MTL. Experimental results show that the multi-task performance of HCA is often better than its single-task performance, indicating that these human traits can make complementary positive impact on model performance. In addition, the results show that our proposed methods improve model performance over baseline methods, and HCA outperforms the state-of-the-art open-domain CAs on response retrieval. Finally, our case study shows that HCA can demonstrate multiple human traits and produce consistent, informative, and empathetic responses.

8.2 Future Work

While we have made much progress towards humanizing open-domain CAs, much work remains to be done because we are still far from human-level open-domain CAs [24, 25]. In the following paragraphs, we discuss potential research directions towards this goal:

More Human Traits: Our thesis only covers a subset of human traits, i.e., emotion, commonsense, persona, and empathy. There are many other human traits that are not well studied in open-domain CAs, e.g., curiosity, persuasion, and humor, etc. Endowing more human traits into open-domain CAs is a promising direction towards human-level performance.

Better Multi-Task Learning: Our HCA relies on multi-task learning (MTL) to combine multiple human traits. Better MTL methods, e.g., better data sampling strategies, would help HCA avoid the problems of underfitting and catastrophic forgetting, and thus balance different human traits better.

More Controllable Response Generation: Current open-domain CAs have limited capabilities in controlling the syntax and semantics of responses. Developing more controllable response generation methods would allow the CAs to selectively generate responses of specific human traits that are appropriate in the application domains and conversational contexts. For example, empathetic responses may be preferred to other types of responses in the counseling domain.

Few-Shot Learning for Open-Domain CAs: Existing CAs often require a large number of conversations to train or finetune; however, conversation datasets exhibiting human traits are generally small. Developing efficient few-shot learning methods would allow CAs to learn these human traits with fewer examples.

More Reliable Automatic Evaluation Methods for Open-Domain CAs: A major and long-lasting challenge for the research in open-domain CAs is the lack of a reliable automatic evaluation method. Existing open-domain CAs are primarily evaluated by humans. However, human evaluation is costly, and different human evaluation results are difficult to compare due to different annotators. A more reliable automatic evaluation method for open-domain CAs would lead to a faster research cycle and a more consistent comparison between models.

Bibliography

[1] Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. Dailydialog: A manually labelled multi-turn dialogue dataset. In *IJCNLP*, pages 986–995, 2017. xx, 31, 42, 52

[2] Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *ACL*, pages 5370–5381, 2019. xx, xxii, 3, 21, 62, 80, 81, 82, 85, 86, 87, 91, 103

[3] Zhenqiao Song, Xiaoqing Zheng, Lu Liu, Mu Xu, and Xuan-Jing Huang. Generating responses with a specific emotion in dialog. In *ACL*, pages 3685–3695, 2019. xxi, 20, 59, 60, 62, 70, 71, 77

[4] Joseph L Fleiss and Jacob Cohen. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and Psychological Measurement*, 33(3):613–619, 1973. xxii, 33, 73

[5] Saif M. Mohammad and Peter D. Turney. Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3):436–465, 2013. xxii, 75

[6] Joseph L Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378, 1971. xxii, 84

[7] 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? In *ACL*, pages 2204–2213, 2018. xxii, 3, 4, 20, 82, 85, 87, 93, 103, 104

[8] Pierre-Emmanuel Mazare, Samuel Humeau, Martin Raison, and Antoine Bordes. Training millions of personalized dialogue agents. In *EMNLP*, pages 2775–2779, 2018. xxii, 82, 84, 85

[9] Dan Jurafsky. *Speech & language processing*. Pearson Education India, 2019. 1, 18

[10] Robin IM Dunbar, Anna Marriott, and Neil DC Duncan. Human conversational behavior. *Human Nature*, 8(3):231–246, 1997. 2

[11] Rainer Winkler and Matthias Soellner. Unleashing the potential of chatbots in education: A state-of-the-art analysis. In *Academy of Management Proceedings*, page 15903, 2018. 2

[12] Pavel Smutny and Petra Schreiberova. Chatbots for learning: A review of educational chatbots for the facebook messenger. *Computers & Education*, page 103862, 2020. 2

[13] Joao Luis Zeni Montenegro, Cristiano André da Costa, and Rodrigo da Rosa Righi. Survey of conversational agents in health. *Expert Systems with Applications*, 129:56–67, 2019. 2

[14] Aditya Nrusimha Vaidyam, Hannah Wisniewski, John David Halamka, Matcheri S Kashavan, and John Blake Torous. Chatbots and conversational agents in mental health: a review of the psychiatric landscape. *The Canadian Journal of Psychiatry*, 64(7):456–464, 2019. 2

[15] INTELLIGENCE BY AM TURING. Computing machinery and intelligence-am turing. *Mind*, 59(236):433, 1950. 2

[16] 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. 2, 16

[17] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. 2

[18] Oriol Vinyals and Quoc Le. A neural conversational model. *arXiv preprint arXiv:1506.05869*, 2015. 3, 18, 25, 71, 80

[19] Ryan Lowe, Nissan Pow, Iulian Vlad Serban, and Joelle Pineau. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *SIGDIAL*, pages 285–294, 2015. 17, 18, 80, 83, 90, 91

[20] Jianfeng Gao, Michel Galley, and Lihong Li. Neural approaches to conversational ai. In *SIGIR*, pages 1371–1374, 2018. 3, 17, 18

[21] Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *AAAI*, pages 730–739, 2018. 3, 20, 25, 41, 59, 62, 69, 70, 71, 77

[22] Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. Commonsense knowledge aware conversation generation with graph attention. In *IJCAI*, pages 4623–4629, 2018. 4, 20, 42, 44, 59, 60, 61, 66, 68, 69, 71

[23] Heung-Yeung Shum, Xiao-dong He, and Di Li. From eliza to xiaoice: challenges and opportunities with social chatbots. *Frontiers of Information Technology & Electronic Engineering*, 19(1):10–26, 2018. 4

[24] Minlie Huang, Xiaoyan Zhu, and Jianfeng Gao. Challenges in building intelligent open-domain dialog systems. *ACM Transactions on Information Systems (TOIS)*, 38(3):1–32, 2020. 114

[25] Stephen Roller, Y-Lan Boureau, Jason Weston, Antoine Bordes, Emily Dinan, Angela Fan, David Gunning, Da Ju, Margaret Li, Spencer Poff, et al. Open-domain conversational agents: Current progress, open problems, and future directions. *arXiv preprint arXiv:2006.12442*, 2020. 3, 4, 114

[26] Scott Brave and Clifford Nass. Emotion in human-computer interaction. *Human-Computer Interaction Fundamentals*, 20094635:53–68, 2009. 4

[27] John-Jules Ch Meyer and Frank Veltman. Intelligent agents and common sense reasoning. *Handbook of Modal Logic*, 3:991–1029, 2007.

[28] Gene Ball. Emotion and personality in a conversational agent. *Embodied Conversational Agents*, pages 189–219, 2000.

[29] Adjamir M Galvao, Flavia A Barros, Andre MM Neves, and Geber L Ramalho. Persona-AIML: An architecture developing chatterbots with personality. In *AAMAS*, pages 1266–1267, 2004.

[30] Iolanda Leite, André Pereira, Samuel Mascarenhas, Carlos Martinho, Rui Prada, and Ana Paiva. The influence of empathy in human–robot relations. *International Journal of Human-Computer Studies*, 71(3):250–260, 2013. 4

[31] Reinhard Fiehler. How to do emotions with words: Emotionality in conversations. *The Verbal Communication of Emotions*, pages 79–106, 2002. 4

[32] Kyo-Joong Oh, Dongkun Lee, Byungsoo Ko, and Ho-Jin Choi. A chatbot for psychiatric counseling in mental healthcare service based on emotional dialogue analysis and sentence generation. In *2017 18th IEEE International Conference on Mobile Data Management (MDM)*, pages 371–375. IEEE, 2017. 4

[33] Asma Ghandeharioun, Daniel McDuff, Mary Czerwinski, and Kael Rowan. Emma: An emotion-aware wellbeing chatbot. In *ACII*, pages 1–7. IEEE, 2019. 4

[34] Anbang Xu, Zhe Liu, Yufan Guo, Vibha Sinha, and Rama Akkiraju. A new chatbot for customer service on social media. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 3506–3510, 2017. 4

[35] FL van Holthoon and David R Olson. *Common sense: the foundations for social science*, volume 6. University Press of America, 1987. 4, 20

[36] 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. 4, 20, 44, 61

[37] Carl Jung. *Psychological types*. Taylor & Francis, 2016. 4, 20, 80

[38] Mark R Leary and Ashley Batts Allen. Personality and persona: Personality processes in self-presentation. *Journal of Personality*, 79(6):1191–1218, 2011. 4, 80, 96

[39] François Mairesse, Marilyn A Walker, Matthias R Mehl, and Roger K Moore. Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of Artificial Intelligence Research*, 30:457–500, 2007. 4

[40] Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. A persona-based neural conversation model. In *ACL*, pages 994–1003, 2016. 4, 20, 21, 66, 82

[41] Fatima Ali Amer Jid Almahri, David Bell, and Mahir Arzoky. Personas design for conversational systems in education. In *Informatics*, page 46. Multidisciplinary Digital Publishing Institute, 2019. 4

[42] Asbjørn Følstad, Cecilie Bertinussen Nordheim, and Cato Alexander Bjørkli. What makes users trust a chatbot for customer service? an exploratory interview study. In *International Conference on Internet Science*, pages 194–208. Springer, 2018. 4

[43] Kimberley Rogers, Isabel Dziobek, Jason Hassenstab, Oliver T Wolf, and Antonio Convit. Who cares? revisiting empathy in asperger syndrome. *Journal of Autism and Developmental Disorders*, 37(4):709–715, 2007. 5, 21, 79

[44] Joana Fernandes Coutinho, Patrícia Oliveira Silva, and Jean Decety. Neurosciences, empathy, and healthy interpersonal relationships: Recent findings and implications for counseling psychology. *Journal of Counseling Psychology*, 61(4):541, 2014. 5, 21, 79

[45] Jonathan Tarter Klein. *Computer response to user frustration*. PhD thesis, Massachusetts Institute of Technology, 1998. 5, 21, 79

[46] K Liu and Rosalind W Picard. Embedded empathy in continuous, interactive health assessment. In *CHI Workshop on HCI Challenges in Health Assessment*, volume 1, page 3, 2005. 5, 21

[47] Li Zhou, Jianfeng Gao, Di Li, and Heung-Yeung Shum. The design and implementation of xiaoice, an empathetic social chatbot. *Computational Linguistics*, 0:1–62, 2018. 5, 59, 60, 79, 80

[48] Peixiang Zhong, Di Wang, and Chunyan Miao. An affect-rich neural conversational model with biased attention and weighted cross-entropy loss. In *AAAI*, pages 7492–7500, 2019. 5, 23, 44, 62

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017. 5, 12, 13, 14, 15, 42, 47, 50, 66, 68, 71, 80

[50] Peixiang Zhong, Di Wang, and Chunyan Miao. Knowledge-enriched transformer for emotion detection in textual conversations. In *EMNLP-IJCNLP*, pages 165–176, 2019. 5, 41

[51] Peixiang Zhong, Di Wang, Pengfei Li, Chen Zhang, Hao Wang, and Chunyan Miao. CARE: Commonsense-aware emotional response generation with latent concepts. In *AAAI*, 2021. 6, 59

[52] Peixiang Zhong, Chen Zhang, Hao Wang, Yong Liu, and Chunyan Miao. Towards persona-based empathetic conversational models. In *EMNLP*, pages 6556–6566, 2020. 6, 79

[53] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of Machine Learning Research*, 3(Feb):1137–1155, 2003. 9

[54] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *NeurIPS*, pages 3111–3119, 2013. 9

[55] Chigozie Nwankpa, Winifred Ijomah, Anthony Gachagan, and Stephen Marshall. Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*, 2018. 10

[56] Yoshua Bengio, Patrice Simard, Paolo Frasconi, et al. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994. 10, 54

[57] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. 10, 44

[58] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *EMNLP*, pages 1724–1734, 2014. 11, 18, 23, 44

[59] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014. 11

[60] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. A review of recurrent neural networks: Lstm cells and network architectures. *Neural Computation*, 31(7):1235–1270, 2019. 11

[61] SHI Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *NeurIPS*, pages 802–810, 2015. 11

[62] Kai Sheng Tai, Richard Socher, and Christopher D Manning. Improved semantic representations from tree-structured long short-term memory networks. In *ACL*, pages 1556–1566, 2015. 11

[63] James Bradbury, Stephen Merity, Caiming Xiong, and Richard Socher. Quasi-recurrent neural networks. *arXiv preprint arXiv:1611.01576*, 2016. 11

[64] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *NeurIPS*, pages 3104–3112, 2014. 11, 18, 23

[65] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014. 12, 43, 50

[66] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *EMNLP*, pages 1412–1421, 2015. doi: 10.18653/v1/D15-1166. 12, 24

[67] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 14

[68] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *ACL*, pages 2978–2988, 2019. 15, 45

[69] Gonçalo M Correia, Vlad Niculae, and André FT Martins. Adaptively sparse transformers. In *EMNLP-IJCNLP*, pages 2174–2184, 2019.

[70] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*, 2020. 15

[71] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020. 15

[72] Sinong Wang, Belinda Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

[73] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Jared Davis, Tamas Sarlos, David Belanger, Lucy Colwell, and Adrian Weller. Masked language modeling for proteins via linearly scalable long-context transformers. *arXiv preprint arXiv:2006.03555*, 2020.

[74] Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *arXiv preprint arXiv:2009.06732*, 2020. 15

[75] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018. 15

[76] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186, 2019. 15, 45, 70, 80, 88, 100

[77] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. 15

[78] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *ICCV*, pages 19–27, 2015. 15

[79] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*, 2019. 15, 70

[80] 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. 15

[81] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019. 15

[82] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. In *ICLR*, 2020. 15

[83] Weijie Liu, Peng Zhou, Zhiruo Wang, Zhe Zhao, Haotang Deng, and QI JU. FastBERT: a self-distilling BERT with adaptive inference time. In *ACL*, pages 6035–6044, 2020. 15

[84] Ji Xin, Raphael Tang, Jaejun Lee, Yaoliang Yu, and Jimmy Lin. DeeBERT: Dynamic early exiting for accelerating BERT inference. In *ACL*, pages 2246–2251, 2020. 15

[85] Alexis Conneau and Guillaume Lample. Cross-lingual language model pretraining. In *NeurIPS*, pages 7059–7069, 2019. 15

[86] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *ACL*, pages 8440–8451, 2020. 15

[87] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mass: Masked sequence to sequence pre-training for language generation. In *ICML*, pages 5926–5936, 2019. 15

[88] Kenneth Mark Colby, Sylvia Weber, and Franklin Dennis Hilf. Artificial paranoia. *Artificial Intelligence*, 2(1):1–25, 1971. 16

[89] Sina Jafarpour, Christopher JC Burges, and Alan Ritter. Filter, rank, and transfer the knowledge: Learning to chat. *Advances in Ranking*, 10:2329–9290, 2010. 17

[90] Anton Leuski and David Traum. Npceditor: Creating virtual human dialogue using information retrieval techniques. *AI Magazine*, 32(2):42–56, 2011. 17

[91] Alan Ritter, Colin Cherry, and William B Dolan. Data-driven response generation in social media. In *EMNLP*, pages 583–593, 2011. 17, 18

[92] Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In *ACL*, pages 496–505, 2017. 18, 83

[93] Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. Multi-turn response selection for chatbots with deep attention matching network. In *ACL*, volume 1, pages 1118–1127, 2018. 45, 90, 93

[94] Chunyuan Yuan, Wei Zhou, Mingming Li, Shangwen Lv, Fuqing Zhu, Jizhong Han, and Songlin Hu. Multi-hop selector network for multi-turn response selection in retrieval-based chatbots. In *EMNLP-IJCNLP*, pages 111–120, 2019. 18, 83

[95] Lifeng Shang, Zhengdong Lu, and Hang Li. Neural responding machine for short-text conversation. In *ACL*, pages 1577–1586, 2015. 18, 25

[96] Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. A neural network approach to context-sensitive generation of conversational responses. In *NAACL*, pages 196–205, 2015. 18

[97] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. In *NAACL*, pages 110–119, 2016. 18, 33, 70

[98] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C Courville, and Yoshua Bengio. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*, pages 3295–3301, 2017. 18, 23

[99] Yuanlong Shao, Stephan Gouws, Denny Britz, Anna Goldie, Brian Strope, and Ray Kurzweil. Generating high-quality and informative conversation responses with sequence-to-sequence models. In *EMNLP*, pages 2210–2219, 2017. 18

[100] Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiujun Li, Chris Brockett, and Bill Dolan. Generating informative and diverse conversational responses via adversarial information maximization. In *NeurIPS*, pages 1810–1820, 2018. 18

[101] Abigail See, Stephen Roller, Douwe Kiela, and Jason Weston. What makes a good conversation? how controllable attributes affect human judgments. In *NAACL*, pages 1702–1723, 2019. 18

[102] Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *ACL*, pages 654–664, 2017. 18, 25

[103] Yookoon Park, Jaemin Cho, and Gunhee Kim. A hierarchical latent structure for variational conversation modeling. In *NAACL*, pages 1792–1801, 2018.

[104] Siqi Bao, Huang He, Fan Wang, Hua Wu, and Haifeng Wang. PLATO: Pre-trained dialogue generation model with discrete latent variable. In *ACL*, pages 85–96, Online, 2020. 18

[105] 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. 19, 100

[106] 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. 19, 60, 81, 100, 101, 104, 105

[107] Magalie Ochs, Catherine Pelachaud, and David Sadek. An empathic virtual dialog agent to improve human-machine interaction. In *AAMAS*, pages 89–96, 2008. 20, 25, 62

[108] Thomas S Polzin and Alexander Waibel. Emotion-sensitive human-computer interfaces. In *ISCA Tutorial and Research Workshop (ITRW) on Speech and Emotion*, 2000. 62

[109] Klaus R Scherer, Angela Schorr, and Tom Johnstone. *Appraisal processes in emotion: Theory, methods, research*. Oxford University Press, 2001. 20, 62

[110] Miles Hewstone and W Stroebe. *Introduction to Social Psychology: A European Perspective*. Oxford Blackwell Publishers, 01 2001. 20, 25

[111] Nabiha Asghar, Pascal Poupart, Jesse Hoey, Xin Jiang, and Lili Mou. Affective neural response generation. In *ECIR*, pages 154–166, 2018. 20, 44, 62

[112] Xianda Zhou and William Yang Wang. Mojitalk: Generating emotional responses at scale. In *ACL*, pages 1128–1137, 2018. 20, 26, 62

[113] Chul Min Lee and Shrikanth S Narayanan. Toward detecting emotions in spoken dialogs. *IEEE Transactions on Speech and Audio Processing*, 13(2): 293–303, 2005. 20, 43

[114] Laurence Devillers and Laurence Vidrascu. Real-life emotions detection with lexical and paralinguistic cues on human-human call center dialogs. In *ICSLP*, 2006. 43, 44

[115] Laurence Devillers, Ioana Vasilescu, and Lori Lamel. Annotation and detection of emotion in a task-oriented human-human dialog corpus. In *Proceedings of ISLE Workshop*, 2002. 20, 43

[116] Devamanyu Hazarika, Soujanya Poria, Amir Zadeh, Erik Cambria, Louis-Philippe Morency, and Roger Zimmermann. Conversational memory network for emotion recognition in dyadic dialogue videos. In *NAACL*, volume 1, pages 2122–2132, 2018. 20, 42, 44

[117] Devamanyu Hazarika, Soujanya Poria, Rada Mihalcea, Erik Cambria, and Roger Zimmermann. Icon: Interactive conversational memory network for multimodal emotion detection. In *EMNLP*, pages 2594–2604, 2018. 42, 44

[118] Sayyed M Zahiri and Jinho D Choi. Emotion detection on tv show transcripts with sequence-based convolutional neural networks. In *Workshops at AAAI*, 2018. 52

[119] Ankush Chatterjee, Umang Gupta, Manoj Kumar Chinnakotla, Radhakrishnan Srikanth, Michel Galley, and Puneet Agrawal. Understanding emotions in text using deep learning and big data. *Computers in Human Behavior*, 93:309 – 317, 2019. ISSN 0747-5632. 41, 52

[120] Soujanya Poria, Navonil Majumder, Rada Mihalcea, and Eduard Hovy. Emotion recognition in conversation: Research challenges, datasets, and recent advances. *arXiv preprint arXiv:1905.02947*, 2019. 20, 44, 54

[121] Soujanya Poria, Erik Cambria, Devamanyu Hazarika, Navonil Majumder, Amir Zadeh, and Louis-Philippe Morency. Context-dependent sentiment analysis in user-generated videos. In *ACL*, volume 1, pages 873–883, 2017. 20, 42, 44, 46, 53, 54

[122] Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen-tau Yih, and Michel Galley. A knowledge-grounded neural conversation model. In *AAAI*, pages 5110–5117, 2018. 20, 44, 61

[123] Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. Wizard of wikipedia: Knowledge-powered conversational agents. In *ICLR*, 2019. 59, 103

[124] Yi-Lin Tuan, Yun-Nung Chen, and Hung-yi Lee. DyKgChat: Benchmarking dialogue generation grounding on dynamic knowledge graphs. In *EMNLP-IJCNLP*, pages 1855–1865, 2019.

[125] Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. OpenDialKG: Explainable conversational reasoning with attention-based walks over knowledge graphs. In *ACL*, pages 845–854, 2019. 61

[126] Houyu Zhang, Zhenghao Liu, Chenyan Xiong, and Zhiyuan Liu. Grounded conversation generation as guided traverses in commonsense knowledge graphs. In *ACL*, 2020. 20, 59, 62, 66, 71

[127] Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149*, 2019. 20, 82, 88

[128] Andrea Madotto, Zhaojiang Lin, Chien-Sheng Wu, and Pascale Fung. Personalizing dialogue agents via meta-learning. In *ACL*, pages 5454–5459, 2019.

[129] Yinhe Zheng, Rongsheng Zhang, Xiaoxi Mao, and Minlie Huang. A pre-training based personalized dialogue generation model with persona-sparse data. *arXiv preprint arXiv:1911.04700*, 2019. 20, 82

[130] Peter Wright and John McCarthy. Empathy and experience in hci. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 637–646, 2008. 21, 79

[131] Farhad Bin Siddique, Onno Kampman, Yang Yang, Anik Dey, and Pascale Fung. Zara returns: Improved personality induction and adaptation by an empathetic virtual agent. In *ACL*, pages 121–126, 2017. 21, 82

[132] Robert R Morris, Kareem Kouddous, Rohan Kshirsagar, and Stephen M Schueller. Towards an artificially empathic conversational agent for mental health applications: system design and user perceptions. *Journal of Medical Internet Research*, 20(6):e10148, 2018.

[133] Weiyan Shi and Zhou Yu. Sentiment adaptive end-to-end dialog systems. In *ACL*, pages 1509–1519, 2018.

[134] Jamin Shin, Peng Xu, Andrea Madotto, and Pascale Fung. Happybot: Generating empathetic dialogue responses by improving user experience look-ahead. *arXiv preprint arXiv:1906.08487*, 2019.

[135] Zhaojiang Lin, Andrea Madotto, Jamin Shin, Peng Xu, and Pascale Fung. MoEL: Mixture of empathetic listeners. In *EMNLP-IJCNLP*, pages 121–132, 2019. 21, 62, 82

[136] Rohola Zandie and Mohammad H Mahoor. Emptransfo: A multi-head transformer architecture for creating empathetic dialog systems. *arXiv preprint arXiv:2003.02958*, 2020. 21, 82

[137] Nalin Chhibber and Edith Law. Using conversational agents to support learning by teaching. *arXiv preprint arXiv:1909.13443*, 2019. 21

[138] Edith Law, Parastoo Baghaei Ravari, Nalin Chhibber, Dana Kulic, Stephanie Lin, Kevin D. Pantasdo, Jessy Ceha, Sangho Suh, and Nicole Dillen. *Curiosity Notebook: A Platform for Learning by Teaching Conversational Agents*, page 1–9. Association for Computing Machinery, New York, NY, USA, 2020. 21

[139] Tatsuya Narita and Yasuhiko Kitamura. Persuasive conversational agent with persuasion tactics. In *International conference on persuasive technology*, pages 15–26. Springer, 2010. 21

[140] Xuewei Wang, Weiyan Shi, Richard Kim, Yoojung Oh, Sijia Yang, Jingwen Zhang, and Zhou Yu. Persuasion for good: Towards a personalized persuasive dialogue system for social good. In *ACL*, pages 5635–5649, 2019. 21

[141] Pawel Dybala, Michal Ptaszynski, Rafal Rzepka, and Kenji Araki. Humoroids: conversational agents that induce positive emotions with humor. In *AAMAS'09 Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems*, volume 2, pages 1171–1172. ACM, 2009. 21

[142] Andreea Niculescu, Betsy van Dijk, Anton Nijholt, Haizhou Li, and Swee Lan See. Making social robots more attractive: the effects of voice pitch, humor and empathy. *International journal of social robotics*, 5(2):171–191, 2013. 21

[143] Rosalind W Picard. *Affective computing*, volume 252. MIT Press Cambridge, 1997. 23

[144] Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. Topic aware neural response generation. In *AAAI*, pages 3351–3357, 2017. 23, 63

[145] Zoraida Callejas, David Griol, and Ramón López-Cózar. Predicting user mental states in spoken dialogue systems. *EURASIP Journal on Advances in Signal Processing*, 2011(1):6, 2011. 23

[146] Bilyana Martinovski and David Traum. Breakdown in human-machine interaction: the error is the clue. In *Proceedings of the ISCA Tutorial and Research Workshop on Error Handling in Dialogue Systems*, pages 11–16, 2003. 23

[147] Jennifer Robison, Scott McQuiggan, and James Lester. Evaluating the consequences of affective feedback in intelligent tutoring systems. In *ACII*, pages 1–6, 2009. 23

[148] Kathleen Kara Fitzpatrick, Alison Darcy, and Molly Vierhile. Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (woebot): a randomized controlled trial. *JMIR Mental Health*, 4(2):e19, 2017. 23, 79

[149] Sayan Ghosh, Mathieu Chollet, Eugene Laksana, Louis-Philippe Morency, and Stefan Scherer. Affect-LM: A neural language model for customizable affective text generation. In *ACL*, pages 634–642, 2017. 24, 63

[150] Amy Beth Warriner, Victor Kuperman, and Marc Brysbaert. Norms of valence, arousal, and dominance for 13,915 english lemmas. *Behavior Research Methods*, 45(4):1191–1207, 2013. 24, 27

[151] Albert Mehrabian. Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament. *Current Psychology*, 14(4):261–292, 1996. 24, 27, 47, 58

[152] Marcin Skowron. Affect listeners: Acquisition of affective states by means of conversational systems. In *Development of Multimodal Interfaces: Active Listening and Synchrony*, pages 169–181. Springer, 2010. 25

[153] Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen tau Yih, and Michel Galley. A knowledge-grounded neural conversation model. In *AAAI*, 2018. 25

[154] Nabiha Asghar, Pascal Poupart, Jesse Hoey, Xin Jiang, and Lili Mou. Affective neural response generation. In *ECIR*, pages 154–166, 2018. 26, 36, 37

[155] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. A simple but tough-to-beat baseline for sentence embeddings. In *ICLR*, 2016. 28, 29

[156] Svetlana Kiritchenko and Saif Mohammad. The effect of negators, modals, and degree adverbs on sentiment composition. In *WASSA*, pages 43–52, 2016. doi: 10.18653/v1/W16-0410. 29, 35

[157] Jorge Carrillo-de Albornoz and Laura Plaza. An emotion-based model of negation, intensifiers, and modality for polarity and intensity classification. *JASIST*, 64(8):1618–1633, 2013. 29

[158] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543, 2014. 29, 54, 71, 72

[159] Jörg Tiedemann. News from opus-a collection of multilingual parallel corpora with tools and interfaces. In *Recent Advances in Natural Language Processing*, volume 5, pages 237–248, 2009. 31

[160] Cristian Danescu-Niculescu-Mizil and Lillian Lee. Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*, 2011. 31

[161] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002. 32

[162] Chia-Wei Liu, Ryan Lowe, Iulian Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *EMNLP*, pages 2122–2132, 2016. doi: 10.18653/v1/D16-1230. 32

[163] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 33, 54, 72, 91

[164] Navonil Majumder, Soujanya Poria, Devamanyu Hazarika, Rada Mihalcea, Alexander Gelbukh, and Erik Cambria. Dialoguernn: An attentive rnn for emotion detection in conversations. In *AAAI*, 2019. 42, 44, 52, 53, 54

[165] 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. 42, 43, 53, 54

[166] Jianpeng Cheng, Li Dong, and Mirella Lapata. Long short-term memory-networks for machine reading. In *EMNLP*, pages 551–561, 2016. 42

[167] Peixiang Zhong and Chunyan Miao. ntuer at SemEval-2019 task 3: Emotion classification with word and sentence representations in RCNN. In *SemEval*, pages 282–286, 2019. 44

[168] Sangdo Han, Jeesoo Bang, Seonghan Ryu, and Gary Geunbae Lee. Exploiting knowledge base to generate responses for natural language dialog listening agents. In *SIGDIAL*, pages 129–133, 2015. 44, 61

[169] Prasanna Parthasarathi and Joelle Pineau. Extending neural generative conversational model using external knowledge sources. In *EMNLP*, pages 690–695, 2018.

[170] Shuman Liu, Hongshen Chen, Zhaochun Ren, Yang Feng, Qun Liu, and Dawei Yin. Knowledge diffusion for neural dialogue generation. In *ACL*, pages 1489–1498, 2018. 61

[171] Nikita Moghe, Siddhartha Arora, Suman Banerjee, and Mitesh M Khapra. Towards exploiting background knowledge for building conversation systems. In *EMNLP*, pages 2322–2332, 2018. 61

[172] Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. Wizard of wikipedia: Knowledge-powered conversational agents. In *ICLR*, 2019. 44

[173] Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. Mem2seq: Effectively incorporating knowledge bases into end-to-end task-oriented dialog systems. In *ACL*, pages 1468–1478, 2018. 44, 61

[174] Chien-Sheng Wu, Richard Socher, and Caiming Xiong. Global-to-local memory pointer networks for task-oriented dialogue. In *ICLR*, 2019.

[175] Junqing He, Bing Wang, Mingming Fu, Tianqi Yang, and Xuemin Zhao. Hierarchical attention and knowledge matching networks with information enhancement for end-to-end task-oriented dialog systems. *IEEE Access*, 7: 18871–18883, 2019. 44

[176] Chloé Kiddon, Luke Zettlemoyer, and Yejin Choi. Globally coherent text generation with neural checklist models. In *EMNLP*, pages 329–339, 2016. 44

[177] Yanchao Hao, Yuanzhe Zhang, Kang Liu, Shizhu He, Zhanyi Liu, Hua Wu, and Jun Zhao. An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge. In *ACL*, pages 221–231, 2017. 61

[178] Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William Cohen. Open domain question answering using early fusion of knowledge bases and text. In *EMNLP*, pages 4231–4242, 2018. 61

[179] Todor Mihaylov and Anette Frank. Knowledgeable reader: Enhancing cloze-style reading comprehension with external commonsense knowledge. In *ACL*, pages 821–832, 2018. 44

[180] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *EMNLP*, pages 79–86. Association for Computational Linguistics, 2002. 44

[181] Sida Wang and Christopher D Manning. Baselines and bigrams: Simple, good sentiment and topic classification. In *ACL*, pages 90–94. Association for Computational Linguistics, 2012.

[182] Armin Seyeditabari, Narges Tabari, and Wlodek Zadrozny. Emotion detection in text: a review. *arXiv preprint arXiv:1806.00674*, 2018. 44

[183] Muhammad Abdul-Mageed and Lyle Ungar. Emonet: Fine-grained emotion detection with gated recurrent neural networks. In *ACL*, volume 1, pages 718–728, 2017. 44

[184] Yuxiang Zhang, Jiamei Fu, Dongyu She, Ying Zhang, Senzhang Wang, and Jufeng Yang. Text emotion distribution learning via multi-task convolutional neural network. In *IJCAI*, pages 4595–4601, 2018. 44

[185] Hamed Khanpour and Cornelia Caragea. Fine-grained emotion detection in health-related online posts. In *EMNLP*, pages 1160–1166, 2018. 44

[186] Yajie Hu, Xiaoou Chen, and Deshun Yang. Lyric-based song emotion detection with affective lexicon and fuzzy clustering method. In *ISMIR*, pages 123–128, 2009. 44

[187] Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307, 2011.

[188] Anil Bandhakavi, Nirmalie Wiratunga, Stewart Massie, and Deepak Padmanabhan. Lexicon generation for emotion detection from text. *IEEE Intelligent Systems*, 32(1):102–108, 2017. 44

[189] Carlo Strapparava, Alessandro Valitutti, et al. Wordnet affect: an affective extension of wordnet. In *LREC*, volume 4, pages 1083–1086. Citeseer, 2004. 44

[190] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In *LREC*, pages 2200–2204, 2010. 44

[191] Saif Mohammad. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 english words. In *ACL*, pages 174–184, 2018. 44, 46, 67

[192] Abhishek Kumar, Daisuke Kawahara, and Sadao Kurohashi. Knowledge-enriched two-layered attention network for sentiment analysis. In *NAACL*, volume 2, pages 253–258, 2018. 45

[193] Christiane Fellbaum. Wordnet. *The Encyclopedia of Applied Linguistics*, 2012. 45

[194] Jiacheng Zhang, Huanbo Luan, Maosong Sun, Feifei Zhai, Jingfang Xu, Min Zhang, and Yang Liu. Improving the transformer translation model with document-level context. In *EMNLP*, pages 533–542, 2018. 45

[195] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*, 2019. 45

[196] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. Ernie: Enhanced language representation with informative entities. In *ACL*, pages 1441–1451, 2019. 45

[197] Matthew E Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A Smith. Knowledge enhanced contextual word representations. In *EMNLP-IJCNLP*, pages 43–54, 2019.

[198] Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. K-bert: Enabling language representation with knowledge graph. In *AAAI*, pages 2901–2908, 2020. 45

[199] Shang-Yu Su, Pei-Chieh Yuan, and Yun-Nung Chen. How time matters: Learning time-decay attention for contextual spoken language understanding in dialogues. In *NAACL*, volume 1, pages 2133–2142, 2018. 46, 56

[200] Robyn Speer, Joshua Chin, and Catherine Havasi. ConceptNet 5.5: an open multilingual graph of general knowledge. In *AAAI*, pages 4444–4451, 2017. 46, 64

[201] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *ICLR*, 2018. 48, 53

[202] Dinghan Shen, Guoyin Wang, Wenlin Wang, Martin Renqiang Min, Qinliang Su, Yizhe Zhang, Chunyuan Li, Ricardo Henao, and Lawrence Carin. Baseline needs more love: On simple word-embedding-based models and associated pooling mechanisms. In *ACL*, pages 440–450, 2018. 48, 49

[203] Saif M. Mohammad. Word affect intensities. In *LREC*, 2018. 49

[204] Paul Ekman. An argument for basic emotions. *Cognition & Emotion*, 6(3-4): 169–200, 1992. 52, 70

[205] 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. 52

[206] 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. 52

[207] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014. 52, 54

[208] Dacher Keltner and Jonathan Haidt. Social functions of emotions at four levels of analysis. *Cognition & Emotion*, 13(5):505–521, 1999. 59

[209] Andrew M Colman. Cooperation, psychological game theory, and limitations of rationality in social interaction. *Behavioral and Brain Sciences*, 26(2):139–153, 2003. 59

[210] Helmut Prendinger and Mitsuru Ishizuka. The empathic companion: A character-based interface that addresses users'affective states. *Applied Artificial Intelligence*, 19(3-4):267–285, 2005. 59

[211] Ronald De Sousa. *The rationality of emotion*. MIT Press, 1990. 59

[212] Michel Tuan Pham. Emotion and rationality: A critical review and interpretation of empirical evidence. *Review of General Psychology*, 11(2):155–178, 2007. 59

[213] Chien-Sheng Wu, Richard Socher, and Caiming Xiong. Global-to-local memory pointer networks for task-oriented dialogue. In *ICLR*, 2019. 61

[214] Igor Shalyminov, Sungjin Lee, Arash Eshghi, and Oliver Lemon. Data-efficient goal-oriented conversation with dialogue knowledge transfer networks. In *EMNLP-IJCNLP*, pages 1741–1751, 2019. 61

[215] Pratyay Banerjee, Kuntal Kumar Pal, Arindam Mitra, and Chitta Baral. Careful selection of knowledge to solve open book question answering. In *ACL*, pages 6120–6129, 2019. 61

[216] Yubo Xie, Ekaterina Svikhnushina, and Pearl Pu. A multi-turn emotionally engaging dialog model. *arXiv preprint arXiv:1908.07816*, 2019. 62

[217] Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. Toward controlled generation of text. In *ICML*, pages 1587–1596, 2017. 63

[218] Ke Wang and Xiaojun Wan. Sentigan: Generating sentimental texts via mixture adversarial networks. In *IJCAI*, pages 4446–4452, 2018.

[219] Pan Li and Alexander Tuzhilin. Towards controllable and personalized review generation. In *EMNLP-IJCNLP*, pages 3228–3236, 2019. 63

[220] Can Xu, Wei Wu, Chongyang Tao, Huang Hu, Matt Schuerman, and Ying Wang. Neural response generation with meta-words. In *ACL*, pages 5416–5426, 2019. 63

[221] Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. CTRL: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*, 2019. 63, 71

[222] Jingyuan Li and Xiao Sun. A syntactically constrained bidirectional-asynchronous approach for emotional conversation generation. *arXiv preprint arXiv:1806.07000*, 2018. 63

[223] Yehong Peng, Yizhen Fang, Zhiwen Xie, and Guangyou Zhou. Topic-enhanced emotional conversation generation with attention mechanism. *Knowledge-Based Systems*, 163:429–437, 2019. 63

[224] Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. RotatE: Knowledge graph embedding by relational rotation in complex space. In *ICLR*, 2019. 63, 65

[225] Pengfei Li, Kezhi Mao, Xuefeng Yang, and Qi Li. Improving relation extraction with knowledge-attention. In *EMNLP-IJCNLP*, pages 229–239, 2019. 63

[226] Xiang Li, Aynaz Taheri, Lifu Tu, and Kevin Gimpel. Commonsense knowledge base completion. In *ACL*, pages 1445–1455, 2016. 63, 65

[227] Itsumi Saito, Kyosuke Nishida, Hisako Asano, and Junji Tomita. Commonsense knowledge base completion and generation. In *CoNLL*, pages 141–150, 2018. 63

[228] Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. COMET: Commonsense transformers for automatic knowledge graph construction. In *ACL*, pages 4762–4779, 2019. 63, 65, 71

[229] Kenneth Ward Church and Patrick Hanks. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29, 1990. 64

[230] Omer Levy, Yoav Goldberg, and Ido Dagan. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics (TACL)*, 3:211–225, 2015. 64

[231] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *NeurIPS*, pages 2787–2795, 2013. 65

[232] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951. 68

[233] Matthew Henderson, Paweł Budzianowski, Iñigo Casanueva, Sam Coope, Daniela Gerz, Girish Kumar, Nikola Mrkšić, Georgios Spithourakis, Pei-Hao Su, Ivan Vulic, and Tsung-Hsien Wen. A repository of conversational datasets. In *Proceedings of the Workshop on NLP for Conversational AI*, 2019. 69

[234] Saif Mohammad. #emotional tweets. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 246–255, 2012. 70

[235] Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. SemEval-2018 task 1: Affect in tweets. In *SemEval*, pages 1–17, 2018. 70

[236] Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *EMNLP*, pages 1615–1625, 2017. 70

[237] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *EMNLP-IJCNLP*, pages 2463–2473, 2019. 71

[238] Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical neural story generation. In *ACL*, pages 889–898, 2018. 71

[239] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8), 2019. 71

[240] Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. OpenNMT: Open-source toolkit for neural machine translation. In *ACL*, 2017. 71

[241] Scott Brave, Clifford Nass, and Kevin Hutchinson. Computers that care: investigating the effects of orientation of emotion exhibited by an embodied computer agent. *International Journal of Human-Computer Studies*, 62(2): 161–178, 2005. 79

[242] Nadine R Richendoller and James B Weaver III. Exploring the links between personality and empathic response style. *Personality and Individual Differences*, 17(3):303–311, 1994. 80

[243] Patricio Costa, Raquel Alves, Isabel Neto, Pedro Marvao, Miguel Portela, and Manuel Joao Costa. Associations between medical student empathy and personality: a multi-institutional study. *PloS one*, 9(3), 2014. 96

[244] Martin C Melchers, Mei Li, Brian W Haas, Martin Reuter, Lena Bischoff, and Christian Montag. Similar personality patterns are associated with empathy in four different countries. *Frontiers in Psychology*, 7:290, 2016. 80

[245] Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *ICLR*, 2020. 81, 83, 88, 90, 91, 100, 102, 104, 105

[246] Zhaojiang Lin, Peng Xu, Genta Indra Winata, Zihan Liu, and Pascale Fung. Caire: An end-to-end empathetic chatbot. *arXiv preprint arXiv:1907.12108*, 2019. 82

[247] Qintong Li, Hongshen Chen, Zhaochun Ren, Zhumin Chen, Zhaopeng Tu, and Jun Ma. EmpGAN: Multi-resolution interactive empathetic dialogue generation. *arXiv preprint arXiv:1911.08698*, 2019. 82

[248] Zhangming Chan, Juntao Li, Xiaopeng Yang, Xiuying Chen, Wenpeng Hu, Dongyan Zhao, and Rui Yan. Modeling personalization in continuous space for response generation via augmented wasserstein autoencoders. In *EMNLP-IJCNLP*, pages 1931–1940, 2019. 82

[249] Jia-Chen Gu, Zhen-Hua Ling, Xiaodan Zhu, and Quan Liu. Dually interactive matching network for personalized response selection in retrieval-based chatbots. In *EMNLP-IJCNLP*, pages 1845–1854, 2019. 83, 90, 91, 93

[250] Xiangyang Zhou, Daxiang Dong, Hua Wu, Shiqi Zhao, Dianhai Yu, Hao Tian, Xuan Liu, and Rui Yan. Multi-view response selection for human-computer conversation. In *EMNLP*, pages 372–381, 2016. 83

[251] Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. Modeling multi-turn conversation with deep utterance aggregation. In *COLING*, pages 3740–3752, 2018.

[252] Qian Chen and Wen Wang. Sequential attention-based network for noetic end-to-end response selection. *arXiv preprint arXiv:1901.02609*, 2019.

[253] Jiazhan Feng, Chongyang Tao, Wei Wu, Yansong Feng, Dongyan Zhao, and Rui Yan. Learning a matching model with co-teaching for multi-turn response selection in retrieval-based dialogue systems. In *ACL*, pages 3805–3815, 2019. 83

[254] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *NeurIPS*, pages 289–297, 2016. 88

[255] Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. Attention-over-attention neural networks for reading comprehension. In *ACL*, pages 593–602, 2017. 89

[256] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, pages 8024–8035, 2019. 91

[257] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *NeurIPS*, pages 2440–2448, 2015. 93

[258] 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 *ACL*, pages 2021–2030, 2020. 99, 100, 101, 103, 104, 105

[259] Siqi Bao, Huang He, Fan Wang, Hua Wu, Haifeng Wang, Wenquan Wu, Zhen Guo, Zhibin Liu, and Xinchao Xu. Plato-2: Towards building an open-domain chatbot via curriculum learning. *arXiv preprint arXiv:2006.16779*, 2020. 100

[260] 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. 100

[261] Ananth Gottumukkala, Dheeru Dua, Sameer Singh, and Matt Gardner. Dynamic sampling strategies for multi-task reading comprehension. In *ACL*, pages 920–924, 2020. 101, 102

[262] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *ACL*, pages 1715–1725, 2016. 102

[263] 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. 103, 104