MANCHESTER
1824
The University of Manchester

REPRESENTATION LEARNING FOR EMOTION RECOGNITION AND MENTAL HEALTH ANALYSIS

A THESIS SUBMITTED TO THE UNIVERSITY OF MANCHESTER FOR THE DEGREE OF
MASTER OF PHILOSOPHY
IN THE FACULTY OF SCIENCE AND ENGINEERING

2022

Kailai Yang
Department of Computer Science

Contents

Word Count: 31659

List of Figures

List of Tables

List of Publications

Kailai Yang, Tianlin Zhang, Sophia Ananiadou. "A Mental State Knowledge-Aware and Contrastive Network for Early Stress and Depression Detection on Social Media". In *Information Processing and Management*, Volume 59, Issue 4, 2022.

Kailai Yang, Tianlin Zhang, Hassan Alhuzali, Sophia Ananiadou. "Cluster-Level Contrastive Learning for Emotion Recognition in Conversations". In *IEEE Transactions on Affective Computing* (In Press).

Terms and Abbreviations

- **NLP** Natural Language Processing
- **CV** Computer Vision
- **ERC** Emotion Recognition in Conversations
- **CL** Contrastive Learning
- **SCL** Supervised Contrastive Learning
- **UCL** Unsupervised Contrastive Learning
- **SCCL** Supervised Cluster-level Contrastive Learning
- **VAD** Valance-Arousal-Dominance
- **CNN** Convolution Neural Networks
- **RNN** Recurrent Neural Networks
- **PLMs** Pre-trained Language Models
- **LSTM** Long-Short Term Memory
- **GRU** Gated Recurrent Unit
- **FFN** Feed-Forward Network
- **BERT** Pre-training of Deep Bidirectional Transformers for Language Understanding
- **RoBERTa** Robustly Optimized BERT pre-training Approach
- **MLM** Masked Language Model
- **NSP** Next Sentence Prediction
- **GNN** Graph Neural Networks
- **GAT** Graph Attention Networks
- **DAG** Directed Acyclic Graph
- **DAGNN** Directed Acyclic Graph Neural Network
- **HRED** Hierarchical Recurrent Encoder-Decoder
- **SPIP** Sentiment Polarity Intensity Prediction

Abstract

A primary goal of artificial intelligence is to understand human mental states. One direction aims at emotionally coherent and empathetic machine systems. As emotion is often indicated in natural language, emotion recognition from text has become an important research topic in the Natural Language Processing (NLP) community. For example, Emotion Recognition in Conversations (ERC) aims to identify the emotion of each utterance within a dialogue, which has attracted growing research interest due to its wide applications in real-world scenarios. In another line of work, interdisciplinary researchers put much effort into automatic mental health analysis, which devises NLP techniques to detect and analyse mental health conditions (e.g. depression, stress and bipolar). Particularly, mental health analysis from social media posts develops fast with the growing availability of large-scale data from social networks.

This thesis aims to push the boundary of the above two tasks from the perspective of representation learning, which is the core of modern deep learning and NLP techniques. Firstly, we explore the application of contrastive learning. Though previous works mainly perform contrastive learning in an unsupervised manner, we focus on supervised contrastive learning as both tasks are modelled as text classification, and rich labelled data are available. For ERC, we propose a low-dimensional Supervised Cluster-level Contrastive Learning (SCCL). SCCL first reduces the high-dimensional contrastive learning space to a three-dimensional affect (emotion) representation space Valance-Arousal-Dominance (VAD), then performs cluster-level contrastive learning to incorporate measurable emotion prototypes from a human-labelled VAD sentiment lexicon. For stress and depression detection, we also introduce contrastive learning to fully leverage label information for capturing class-specific features. Secondly, we propose new knowledge infusion methods to enhance the representations. For ERC, we leverage the pre-trained knowledge adapters to infuse linguistic and factual knowledge in a plug-in manner. To explicitly model the speakers' mental states and enhance the mentalisation ability for stress and depression detection, we extract mental state knowledge from a commonsense knowledge base and infuse the knowledge explicitly to the representations. Then we propose a knowledge–aware mentalisation module to accordingly attend to the most relevant knowledge aspects.

Experiments show that our methods achieve new state-of-the-art results on three ERC and three stress and depression detection datasets. The analysis also proves that the VAD space is not only suitable for ERC but also interpretable, and VAD prototypes enhance the ERC performance and stabilise the training of SCCL. In addition, the pre-trained knowledge adapters benefit the performance of the utterance encoder and SCCL. Finally, factor-specific analysis and visualisation are performed to prove the effectiveness of all proposed modules.

Declaration of Originality

I hereby confirm that no portion of the work referred to in the thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright Statement

i The author of this thesis (including any appendices and/or schedules to this thesis) owns certain copyright or related rights in it (the "Copyright") and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.

ii Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made *only* in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.

iii The ownership of certain Copyright, patents, designs, trademarks and other intellectual property (the "Intellectual Property") and any reproductions of copyright works in the thesis, for example graphs and tables ("Reproductions"), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.

iv Further information on the conditions under which disclosure, publication and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see `http://documents.manchester.ac.uk/DocuInfo.aspx?DocID=24420`), in any relevant Thesis restriction declarations deposited in the University Library, The University Library's regulations and in The University's policy on Presentation of Theses.

Acknowledgements

I wish to thank my supervisor Prof. Sophia Ananiadou for providing me with kind support and encouragement throughout the research. I also wish to thank Tianlin and Hassan for their valuable suggestions and contributions to much of my work. I am always grateful to my family for unconditionally supporting me whenever I need it.

Chapter 1

Introduction

1.1 Motivation

1.1.1 Emotion Recognition

Emotion is defined as people's mental states related to their thoughts, feelings and behaviours, which is one of the most important aspects of human life. Charles Darwin hypothesizes that emotion evolves along with natural selection and develops universal characteristics across races and cultures, and the above unified definition of emotion is available and widely studied in psychology. There are two types of models for emotions in the literature: categorical emotion models and dimensional emotion models. Categorical models classify emotions into fixed discrete categories, and dimensional models define emotions into multi-dimensional continuous vectors, where each dimension defines a corresponding aspect of emotions. In categorical emotion definitions, Ekman et al. [1] defined the six most common emotions: happiness, surprise, sadness, anger, disgust and fear. In addition, Plutchik et al. [2] further defined eight primary emotion types and a wheel of emotions (see Figure 1.1), where each emotion is extended with fine-grained sub-types. In dimensional emotion definitions, a widely used model is Valance-Arousal-Dominance (VAD) [3], [4], where Valance reflects the pleasantness of a stimulus, Arousal reflects the intensity of emotion provoked by a stimulus, and Dominance reflects the degree of control exerted by a stimulus [5]. The dimensional model maps emotions into a continuous spectrum, which facilitates the comparison of emotions using vector computations such as similarity computation. The vectors also enable more fine-grained emotion classifications than categorical emotions, especially for semantically similar emotions such as happy and excited.

Emotion is often indicated in natural language. Therefore, NLP researchers have devoted much effort to emotion recognition from text [6]. However, early emotion recognition works mainly focus on detecting the emotions of a single sentence, which is inconvenient in real-world scenarios such as during conversations. Therefore, Emotion Recognition in Conversations (ERC) task is proposed as a sub-field of emotion recognition, which aims at identifying the emotion of each utterance within a dialogue from pre-defined emotion categories [7]. In recent years, ERC has attracted increasing research interest from the NLP community due to its wide applications. For example, ERC enables dialogue systems to generate emotionally coherent and empathetic responses [8], [9], which is often achieved by accurately recognis-

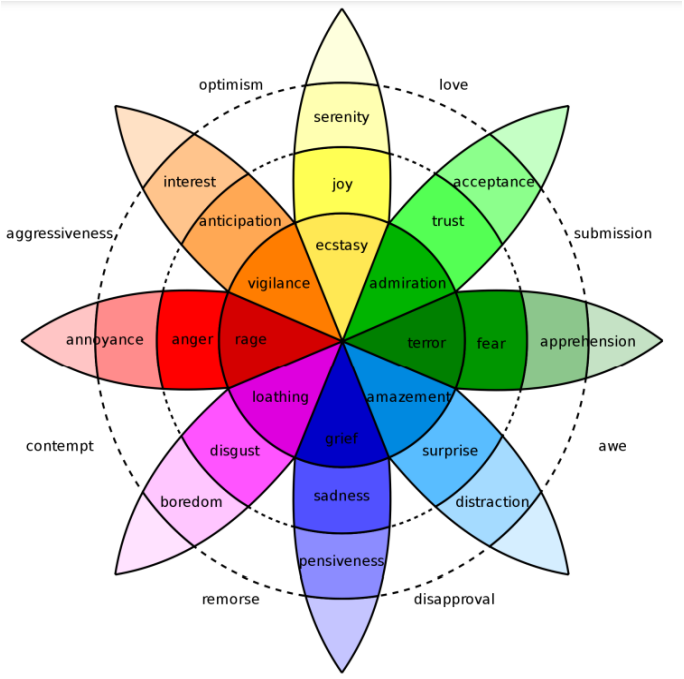


Figure 1.1. Illustration of the wheel of emotions. The figure is adapted from Plutchik et al. [2].

ing the emotion expressed by the dialogue participants and using it as the cue for response generation. It has also been utilised for opinion mining from customer reviews [10], [11]. A common application scenario is analysing the emotions expressed by the customers about certain products in a chat with customer service robots. In addition, ERC is also applied to emotion-related social media analysis [12], [13], where people's attitudes and feelings towards a target topic or public event are mined from their posts and responses on social media, such as Twitter.

Dataset	**Conv.(Train/Val/Test)**	**Utter.(Train/Val/Test)**	**Utter./Conv**
IEMOCAP [14]	100/20/31	4,778/980/1,622	49.2
MELD [15]	1,038/114/280	9,989/1,109/2,610	9.6
EmoryNLP [16]	713/99/85	9,934/1,344/1,328	14.1
DailyDialog [17]	11,118/1,000/1,000	87,170/8,069/7,740	7.9
SEMAINE [18]	63/32	4368/1430	69.3
EmotionLines [19]	720/80/200	10561/1178/2764	14.7
EmoContext [20]	30159/2754/5508	90477/8262/16524	3.0

Table 1.1. Statistics of the current mainstream ERC datasets.

The growing availability of public datasets with diverse characteristics also helps the development of ERC. We list the current mainstream ERC datasets in Table 1.1, where Conv. and Utter. denote the conversation and utterance number. Utter./Conv denotes the average utterance number per dialogue. According to the statistics, these datasets cover a wide range of average utterance numbers from 3.0 to 69.3 per dialogue, which facilitates the evaluation of many techniques, from context modelling to knowledge infusion. In addition, multimodal information (including acoustic, visual and textual information) is also provided in IEMOCAP, SEMAINE and MELD. The datasets also employ different emotion categorisation methods. For example, each utterance of SEMAINE is annotated with dimensional labels with four dimensions: Valance, Arousal, Expectancy (anticipation related factors) and Power

(Dominance), with each factor ranging from -1 to 1, while the rest of the datasets all utilise categorical emotion labels.

ERC introduces extra research challenges compared to vanilla emotion recognition, which are mainly derived from the complex nature of dialogues. We briefly summarise the main challenges of ERC as follows:

- **Context Modelling** A major challenge is the context modelling problem widely encountered in NLP. In a dialogue, the context usually refers to the dialogue history before the target utterance (sometimes also includes future conversations as future utterances can also bring cues to the emotion reasoning of the target utterance). The influence of the context on the target utterance's emotion often contains two aspects: intra- and inter-speaker dependencies [21]. Intra-speaker dependency models the emotional influence of the speaker's psychological activities during the conversation. Inter-speaker dependency deals with the emotional influence of other dialogue participants on the target utterance speaker. We provide an example dialogue in Figure 1.2 to further explain these dependencies, where solid lines show the influence of previous utterances on the emotion of the target utterance (marked blue). As shown, the dialogue history of the target utterance speaker reflects his happy mental state and the dialogue topic "go to the gym and jogging", which directly influences the emotions of the target utterance. The utterance of the other dialogue participant also provides key information such as "Sally is their friend" and raises the positive sentiment of the target utterance.

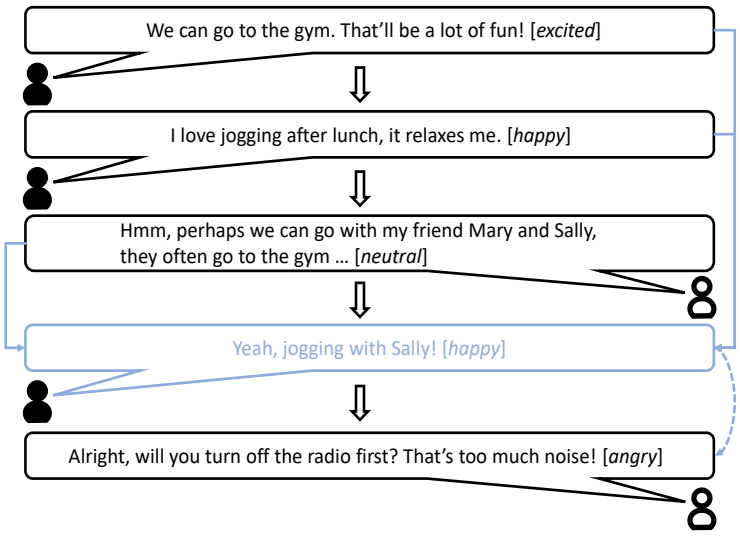


Figure 1.2. An example dialogue with inter- and intra-speaker dependencies.

The usefulness of the context is also influenced by conversation distance and the information richness of the target utterance. The local dialogue history usually plays a more important role in affecting the emotion, and distant dialogue history sometimes takes part in the emotion reasoning, such as when a distant utterance is referred to in the current utterance. In addition, context information plays a more important role in detecting emotions of less informative target utterances. The utterance length often reflects the informativeness as short utterances (e.g. "OK!", "Yes!" and "Why?") tend to be less informative.

- **Multi-Party Conversations** In multi-party conversations, more than two dialogue participants are involved. The intra- and inter-speaker dependencies become more complex, which requires the ERC model to attend to the speaker information and track the status of each individual and multiple co-references. Another challenge is the modelling of the speaker personas, as each speaker has unique and subtle ways of emotional expression. For example, some individuals tend to use sarcasm in their language expressions, where the meaning of certain words varies as the emotion and tone change. Since necessary backgrounds of dialogue participants are often missing from the dialogue, persona modelling is considered a useful technique for ERC.
- **Emotion Dynamics** While the emotions of a dialogue participant tend to stick to a particular status, external stimuli (usually from other dialogue participants) can disturb the consistency. A sudden change of the discussion topic can also lead to emotion dynamics. An example is presented in Figure 1.2, where dashed lines denote the sudden change of topic from "exercise at the gym" to "turn off the radio". While emotion dynamics across sentiment polarity (e.g. change from happy to sad) is relatively easy to model, emotion dynamics within certain sentiment polarity (e.g. change from fear to sad) remains challenging for current ERC models. It requires a deeper understanding of the utterance semantics and more clear distinction of similar emotions.

1.1.2 Mental Health Analysis

Mental health conditions are defined as the conditions that affect a person's thinking, feeling, behaviour or mood[1]. They pose serious public health problems worldwide. There are multiple types of mental health conditions, including depression, schizophrenia, bipolar, bulimia and other psychiatric impairments[2]. According to the latest mental health report, nearly one billion people are suffering from at least one type of mental health conditions, which can lead to self-harm, physical disability and even suicide [22]. However, many of these patients do not receive timely psychiatric treatment to avoid these serious consequences. One reason is that mental health conditions lead people to stigma, which prevents them from seeking clinical aids [23]. The COVID-19 pandemic also exacerbates this problem, with less availability of medical resources.

With social media becoming an integral part of our daily lives[3], people continuously turn to social media platforms such as Twitter and Reddit to share their feelings and express their stress. Similarly, people with mental health conditions often share their mental states and seek help for their mental health issues on these platforms by posting texts, photos and other links, which makes related cues from these social media texts a rich and useful resource for mental health analysis. On the other hand, the reliability of mental health analysis based on social media is rigorously studied. Early works in psychology prove the relations between people with mental health conditions and their textual expressions, which is referred to as "depressive

[1] https://www.nami.org/About-Mental-Illness/Mental-Health-Conditions
[2] https://www.nhs.uk/mental-health/conditions/
[3] https://wearesocial.com/uk/blog/2022/01/digital-2022/

language" [24], [25]. Other works study various mental health conditions and summarise the specific linguistic features of their expressions [25]–[27]. Gkotsis et al. [28] collect large-scale data from the social media Reddit and analyse the linguistic features associated with various kinds of mental disorders. Sentiment features are also proven relevant to mental health conditions. Based on the above observations, many works leverage NLP techniques for text-based mental health analysis on social media [29], [30]. Current methods mainly focus on mental conditions detection, which aims to detect mental conditions tendency from text posts and have achieved promising results.

In our work, we focus on early stress and depression detection on social media among various types of mental conditions for the following reasons:

- Stress is defined as the reaction to extant and future demands and pressures[4] expressed commonly in our daily lives. Many studies have shown too much stress as an indicative factor of mental health conditions [31], [32]. Therefore, stress detection provides valuable references for early diagnosis of mental health conditions.
- Depression remains a highly-untreated [33] but very threatening [34] mental health condition. Research has also shown that depression can manifest by the way people write [35], which facilitates text-based analysis. Depression is also widely researched in NLP-based mental health analysis. According to the statistics in Figure 1.3, 45% of mental health analysis focus on analysing depression, which denotes rich available data and many baseline works for comparison.

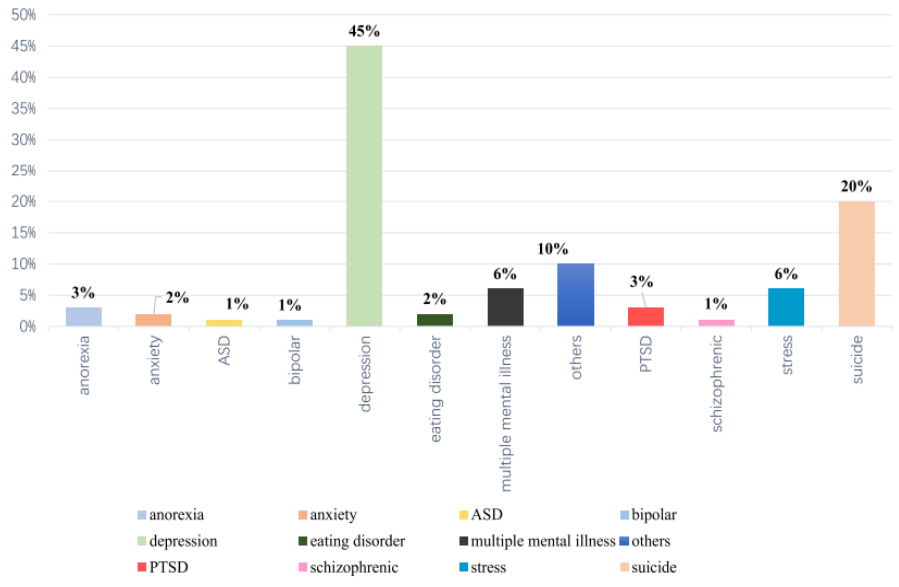


Figure 1.3. The percentages of different mental illnesses studied in mental health analysis. The figure is adapted from Zhang et al. [29].

- The applications of stress and depression detection are not limited to assisting early diagnosis. They can also be utilised in other scenarios, such as alleviating the ethical problems of chatbots [36]. For example, the medical chatbot based on GPT-3 is reported

[4] https://www.apa.org/topics/stress

to tell fictitious patients to commit suicide during the test[5], which can lead to severe consequences in practice. These medical chatbots must be able to detect stress or depression potential and carefully generate the proper response before actual deployment.

Stress and depression detection has many differences from other text classification tasks, and several key challenges remain:

- **Representation Learning** There are several challenges for representation learning:
  - The quantity and quality of the annotated data are not guaranteed. Most of the representation learning methods rely on supervised learning, which is attributed to large-scale training datasets. However, mental health analysis still lacks annotated public datasets. Diagnosis of mental conditions also requires expertise, which is usually time-consuming and expensive. Nevertheless, many datasets are not labelled by experts or only weakly labelled (e.g. labelling posts from different sub-regions of the online forum as the topic), which brings bias and noise to the annotations. In addition, most people do not share their mental states online due to the sensitivity of mental health conditions, which leads to label imbalance in the annotated datasets.
  - Short texts provide limited information. Some posts with depression or stress tendencies are short, which requires the context to provide more information or other commonsense knowledge for the correct detection. Therefore, appropriate techniques are needed for context modelling and knowledge incorporation.
  - The reasoning process for stress and depression detection can be complex as people have various writing styles and semantic heterogeneity. Model performance can be bad when transferred to another dataset. Therefore, more effort is required to develop robust representation learning techniques for different data sources.
- **Interpretability** Successful stress and depression detection methods not only achieve high-quality classification but also understand the cause or explanatory factors of the mental health conditions, which provides clues for the decision-making of the clinicians. However, current methods primarily leverage deep learning techniques to learn distributed text representations. Though achieving high accuracy performance in classification, they lack interpretability in key features utilised for reasoning underlying some predictions. Therefore, another research direction is to open the black box and enhance the explainability of the deep learning models.
- **Ethical Considerations** As stress and depression detection use large-scale mental health-related textual data, the relevant ethical concerns also grow increasingly. The concerns mainly involve the privacy and security of personal health data. Under the guidance of Bentan et al. [37], most previous works follow strict protocols to ensure that the data is appropriately applied in their experiments to protect the privacy and avoid further

[5] https://artificialintelligence-news.com/2020/10/28/medical-chatbot-openai-gpt3-patient-kill-themselves/

psychological distress. In addition, ethical approvals are required from human research ethics committees and institutional review boards for some sensitive data.

1.2 Research Questions

Emotion recognition in conversations and mental health analysis are practical techniques but still with many challenges. As deep learning algorithms become the state of the art in both tasks, enhancing the learnt textual representations grows to be the mainstream in the effort to improve model performance. Therefore, we also focus on developing helpful representation learning techniques for ERC and mental health analysis. In this thesis, we examine the effectiveness of two representation learning techniques: contrastive learning and knowledge infusion. Contrastive learning aims to enhance representation learning by using contrastive samples against each other to learn common features among data clusters and those that set apart each other. Knowledge infusion aims to incorporate task-related knowledge into the learnt representations explicitly or implicitly, which provides more information to the classification phase and facilitates reasoning on the representations. In the following sections, we detail each technique's research questions, hypotheses and objectives.

1.2.1 Contrastive Learning

As a new but prosperous sub-field of representation learning, contrastive learning originates in Computer Vision (CV) [38]–[40]. Its application in NLP includes both supervised and unsupervised manner for enhancing multiple-level representations [41]–[43]. Unsupervised contrastive learning usually constructs positive samples via data augmentation and randomly samples negative pairs from other instances. Supervised Contrastive Learning (SCL) is mainly devised to enhance the traditional supervised text classification, which regards samples with the same label as positive pairs. Therefore, we hope to examine the effectiveness of SCL on the two proposed tasks and raise the following research question:

Research Question #1 Can supervised contrastive learning enhance the representations for ERC and stress and depression detection task? For ERC, a natural method is to employ SCL based on emotion labels. In SCL, the samples labelled with the same emotion are clustered, and samples with different emotions are pushed apart. An expected benefit of SCL is that semantically similar emotions will be easier to distinguish. For example, previous works consistently report that a primary source of errors is the misclassifications between similar emotions (e.g. happy and excited) [44], as the expressions of these emotions in the utterances tend to be similar, which requires the awareness of more fine-grained features. With SCL pushing apart the representations, the model is forced to attend to the fine-grained difference between similar emotions, which benefits the final classification performance. A critical hypothesis of applying SCL is that the potential ambiguities between semantically similar emotions can be distinguished by the textual features learnt in the representations.

Similar approaches and hypotheses for the stress and depression detection task are utilised for leveraging SCL. However, the effect is expected to be less significant, as stress and depression detection is modelled as a binary classification task. To increase the interpretability of the mental health analysis models, we also adapt our detection model to a stress factor detection task, which aims to analyse the causal factor of the stress and model it as a multi-class classification task. SCL is anticipated to perform well on the stress factor detection task.

1.2.2 Knowledge Infusion

Current representation learning methods for NLP leverage information from the input text and make inference based on the learnt embeddings. However, many NLP tasks rely on world knowledge to make correct reasoning. Therefore, many works incorporate appropriate knowledge sources and develop knowledge infusion methods to enrich the semantics of the representations and help the reasoning process. For example, large-scale knowledge graphs are constructed to store commonsense knowledge [45]–[47]. Graph neural networks [48] are devised to aggregate the knowledge for further infusion in tasks such as commonsense question answering, dialogue systems and text classifications. Implicit knowledge infusion is also widely explored. For example, Transformer-based pre-trained language models [49]–[51] infuse knowledge stored in texts via pre-training and perform well in various downstream tasks. With the success of knowledge infusion methods in many tasks, we also explore their applications in ERC and mental health analysis and raise the following research question:

Research Question #2: Can knowledge infusion enrich the representations and benefit the reasoning process for ERC and stress and depression detection task? For ERC, there are many scenarios where extra knowledge is required for correct classification. For example, some utterances are short and lack context. In these cases, some commonsense knowledge, such as the relations between certain entities and emotions, can boost the emotion reasoning process. Extra knowledge also helps the model to understand particular scenarios such as sarcasm. Based on the observations, we explore knowledge infusion to ERC from both explicit and implicit perspectives. First, factual and linguistic knowledge is infused explicitly by incorporating pre-trained knowledge adapters [52], which effortlessly infuses knowledge in a plug-in manner without re-training. Both knowledge types are anticipated to aid ERC as factual knowledge can enrich the semantics, and linguistic knowledge can help analyse the utterance structures. We also introduce human-labelled VAD supervision signals for each emotion from a sentiment lexicon NRC-VAD [53] to the SCL method, which not only lowers the dimension of the contrastive learning space but also aims to facilitate the convergence of the clustering process for each emotion. The VAD knowledge is expected to guide understanding each emotion category's sentiments.

For the stress and depression detection task, we consider the close relations between people's mental health conditions and mental states. We infuse mental state knowledge into the learnt representations. Specifically, we leverage COMET [54], a generative mental state knowledge source, to combine mental state knowledge at the sentence level. The utilised

COMET version is pre-trained on a large-scale mental state knowledge graph called ATOMIC [47]. Each sentence within the post is input to COMET and obtains knowledge from various pre-defined mental state aspects such as "intention of the speaker" and "effect on others". The knowledge-enriched representation for each aspect is further reasoned and combined for classification. The infused mental state knowledge is expected to provide clues on modelling the speakers' mental states, which facilitates the final diagnosis.

1.3 Contributions

1.3.1 Emotion Recognition in Conversations

A key challenge for ERC is distinguishing semantically similar emotions. Some works utilise SCL, which uses categorical emotion labels as supervision signals and contrasts in high-dimensional semantic space. However, categorical labels fail to provide quantitative information about emotions. ERC is also not equally dependent on all embedded features in the semantic space, which makes the high-dimensional SCL inefficient. To address these issues, we propose a novel low-dimensional Supervised Cluster-level Contrastive Learning (SCCL) method, which first reduces the high-dimensional SCL space to a three-dimensional affect representation space Valance-Arousal-Dominance, then performs cluster-level contrastive learning to incorporate measurable emotion prototypes from a sentiment lexicon. To help modelling the dialogue and enriching the context, we leverage the pre-trained knowledge adapters to infuse linguistic and factual knowledge. Experiments show that our method achieves new state-of-the-art results with 69.81% on IEMOCAP, 65.7% on MELD, and 62.51% on DailyDialog datasets. The analysis also proves that the VAD space is not only suitable for ERC but also interpretable, with VAD prototypes enhancing its performance and stabilising the training of SCCL. In addition, the pre-trained knowledge adapters benefit the performance of the utterance encoder and SCCL.

1.3.2 Stress and Depression Detection

Stress and depression detection on social media aim to analyse stress and identify depression tendencies from social media posts, which assist in the early detection of mental health conditions. Existing methods mainly model the mental states of the post-speaker implicitly. They also lack the ability to mentalise for complex mental state reasoning. Besides, they are not designed to capture class-specific features explicitly. To resolve the above issues, we propose a mental state Knowledge–aware and Contrastive Network (KC-Net). In detail, we first extract mental state knowledge from a commonsense knowledge base COMET, and infuse the knowledge using Gated Recurrent Units (GRUs) to model the speaker's mental states explicitly. Then we propose a knowledge–aware mentalisation module based on dot-product attention to accordingly attend to the most relevant knowledge aspects. A supervised contrastive learning module is also utilised to fully leverage label information for capturing

class-specific features. We test the proposed methods on a depression detection dataset Depression_Mixed with 3165 Reddit and blog posts, a stress detection dataset Dreaddit with 3553 Reddit posts, and a stress factors recognition dataset SAD with 6850 SMS-like messages. The experimental results show that our method achieves new state-of-the-art results on all datasets: 95.4% of F1 scores on Depression_Mixed, 83.5% on Dreaddit and 77.8% on SAD, with 2.07% average improvement. Factor-specific analysis and ablation study prove the effectiveness of all proposed modules, while UMAP analysis and case study visualise their mechanisms. We believe our work facilitates the detection and analysis of depression and stress on social media data, and shows potential for applications to other mental health conditions.

1.4 Thesis Structure

The thesis consists of five chapters. In Chapter #1 (Introduction, the current chapter), we introduce the importance of ERC and stress and depression detection and our motivation for conducting research on these tasks. Then we briefly describe our research questions, which mainly focus on adapting contrastive learning and knowledge infusion methods to enhance the representation learning process on these tasks. In addition, our contributions to the literature of both tasks in this thesis are summarised.

In Chapter #2 (Background), we introduce the necessary background information for this thesis. We first focus on the development of representation learning techniques. It starts from introducing state-of-the-art neural network architectures such as CNN, RNN and the Transformer, which are the basis of most modern representation learning methodologies. Then we describe the literature on three cutting-edge representation learning techniques that this thesis focuses on: Transformer-based pre-trained language models, contrastive learning and knowledge-enhanced methods. We also introduce current efforts to improve the representation learning of ERC and mental health analysis.

In Chapter #3 (Cluster-Level Contrastive Learning), we detail our proposed cluster-level contrastive learning method and its application in ERC. Firstly, we briefly summarise the new techniques developed: cluster-level contrastive learning and pre-trained knowledge adapters. Secondly, we explain the process of leveraging factual and linguistic knowledge with pre-trained knowledge adapters in a plug-in manner. Thirdly, we introduce the features and building process of the NRC-VAD emotion lexicon and the detailed methodology of lowering the high dimension of vanilla SCL, combining NRC-VAD, and performing contrastive learning at the cluster level. Finally, we conduct various experiments to examine the effectiveness of the proposed methods, including performance comparison on ERC datasets, ablation study, comparison of different contrastive learning methods, visualisation, etc.

In Chapter #4 (Mental State Knowledge Infusion), we detail our proposed mental state knowledge infusion method and its application in stress and depression detection. Firstly, we briefly introduce the data pre-processing process. Secondly, the mental state knowledge

infusion process is introduced in detail, including the knowledge extraction process from the knowledge source, the knowledge infusion methods, and the automatic mentalisation process. Thirdly, we describe the application of SCL to enhance representation learning. Finally, we conduct experiments on a stress detection, depression detection and stress factor detection task to examine the effectiveness of the proposed model on different mental health analysis tasks. Other analyses such as error analysis, case study and visualisation are exerted to prove the effectiveness of each module further.

In Chapter #5 (Conclusion), we summarise our contributions of this thesis, analyse the limitations of current methods, and propose future research directions.

Chapter 2

Background

2.1 Representation Learning

Representation learning is a subset of machine learning approaches that aims to discover the representations required for feature detection from the data. Machine learning starts by designing features manually, but feature engineering was later replaced by representation learning techniques, where data is sent to the machine to learn representations on its own. Early works explicitly represent the features in each dimension of the representation, such as the bag-of-words representations in NLP. With the development of deep learning techniques, distributed representations have become the mainstream in artificial intelligence, which reduces the high-dimensional representations to low-dimensional dense vectors. Various types of neural network architectures are devised to compute the representations. During training, the representations are usually optimised in a supervised or unsupervised manner via back-propagation [55] using designed loss functions, such as the cross-entropy loss for text classification. For example, the cross-entropy loss for multi-class classification is as follows:

$$\mathcal{L}_{CE} = -\sum_{j=1}^{|C|} Y^j log\hat{Y}^j \tag{2.1}$$

where Y^j and $\hat{Y}^j$ are the j-th element of the prediction probability distribution $Y \in \mathbb{R}^{|C|}$ and the one-hot label $\hat{Y}_i \in \mathbb{R}^{|C|}$ respectively, C denotes the set of classes. This loss is widely utilised in text classification and generation tasks. Based on neural networks, many other techniques are proposed to enhance the representations under different circumstances, such as the recent contrastive learning methods [38], [41]. Progress in knowledge engineering also helps develop various knowledge-enhancing methods for learning representations. This section briefly introduces these techniques and mainly focuses on state-of-the-art representation learning methodologies in NLP.

2.1.1 Neural Networks

Neural networks are inspired by the operations of the human brain and mimic how biological neurons signal to one another, which are a subset of machine learning methods and the core of deep learning. With the growing availability of computational resources, neural networks are

leveraged for representation learning in most application scenarios of artificial intelligence (e.g. NLP, CV, recommendation systems) and achieve state-of-the-art performance.

Feed-Forward Neural Networks

The basic component of neural networks is called the node (also known as neuron). The computation of a single node is called the Perceptron, which is formalised as:

$$y = f(w^\top x + b) \tag{2.2}$$

where $x \in \mathbb{R}^n$ denotes the input signal usually from the target source such as text, image and music. $w \in \mathbb{R}^n$ is a vector and b is a scalar, which are learnable parameters. The fundamental type of neural network is the Feed-Forward Network (FFN), which is composed of multiple Perceptrons. It is formulated as:

$$y = f(\mathbf{W}x + \mathbf{b}) \tag{2.3}$$

where $\mathbf{W} \in \mathbb{R}^{m \times n}$ is the weight matrix, and $b \in \mathbb{R}^m$ is the bias vector. The FFN can naturally be extended to several layers to build deeper neural networks, which solves more complex problems. An example two-layer FFN is formalised as:

$$y = f^{'}(\mathbf{W}^{'} f(\mathbf{W}x + \mathbf{b}) + \mathbf{b}^{'}) \tag{2.4}$$

where $y \in \mathbb{R}^d$ is the learnt representation, $\mathbf{W} \in \mathbb{R}^{m \times n}$, $\mathbf{W}^{'} \in \mathbb{R}^{d \times m}$ are the weight matrix of the two layers, and $b \in \mathbb{R}^m$, $b^{'} \in \mathbb{R}^d$ are the corresponding bias vectors. The multi-layer FFN is able to solve classification problems with non-linear decision boundaries, such as the solution of the famous logic XOR operator. We provide an intuitive view of multi-layer FFN in Figure 2.1, where the input is connected to the output via three intermediate hidden layers. The input of each hidden layer is the output of the last layer, and the output of the final layer is used for classification.

In multi-layer FFNs, all input parts are equally transformed within each layer, and each node within one layer is connected to all nodes in the previous layer. These priors limit its applications in many real-world signals, such as sequence-based texts and two-dimensional images. Therefore, other architectures are developed to process more complex inputs, such as the three mainstream architectures widely used in NLP tasks: CNN, RNN and the Transformer.

In addition, initial NLP works represent words and sentences in real-valued vectors, where each binary dimension denotes the appearance of a word in the vocabulary. This method is known as one-hot representations. Considering the sparsity and high dimension problems of one-hot representations, another branch of work uses multi-layer FFNs to compute a dense representation vector for each word and trains the neural networks on a large-scale dataset. Representative works include "word2vec" [56], which trains the model to predict the context

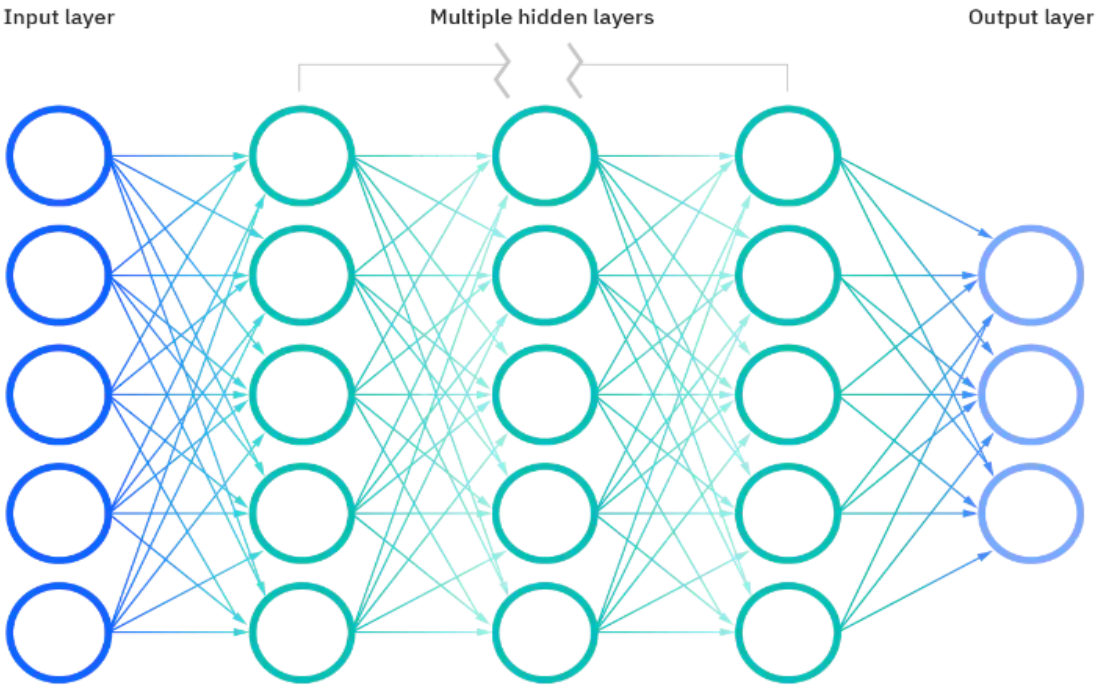


Figure 2.1. Illustration of a three-layer FFN.

words within a context window. Another work "Glove" [57] aggregates and maps the word co-occurrence into a meaningful space where the representations of frequently co-occurred words are distantly similar. These learnt word representations are used as the foundation for sentence-level processing and perform well in numerous downstream NLP tasks. However, these word representations are static as the context of the word changes, which brings limitations in many context-dependent scenarios.

Convolutional Neural Networks

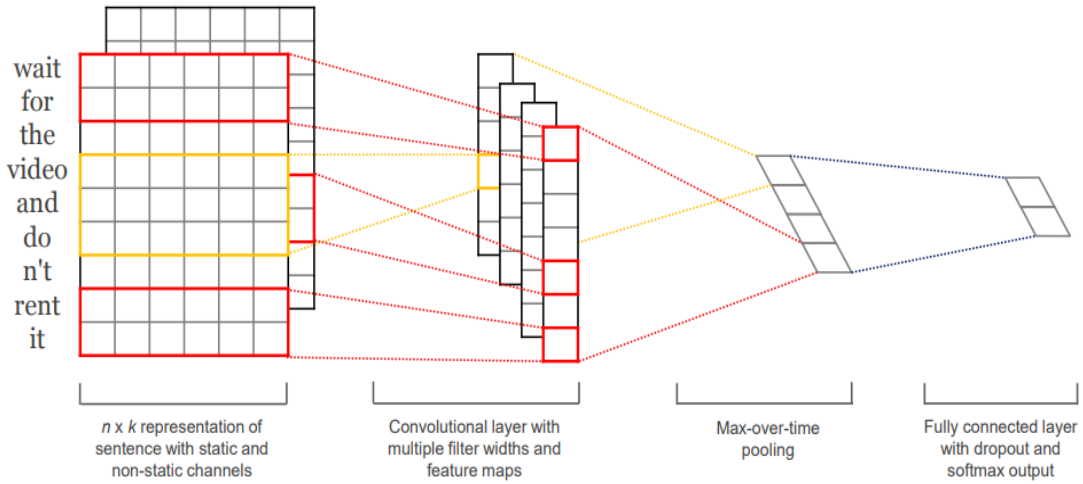


Figure 2.2. Illustration of the text-based CNN structure. The figure is adapted from Kim et al [58].

Convolutional Neural Networks (CNN) were originally designed to extract two-dimensional image features [59], and later applied to the text modality [58] for sentence-level or document-level representation learning. An example of CNN is presented in Figure 2.2. As illustrated, the text sequence is transferred to a word representation matrix $\mathbf{w} \in \mathbb{R}^{s \times d}$ via an embed-

ding look-up process, where s is the sequence length, and d is the dimension of the word representations. The look-up table usually comes from the pre-trained word embeddings previously introduced. Then the representation matrix walks through a set of filters, and each filter learns a phrase-level feature. Specifically, the i-th filter $f_i \in \mathbb{R}^{d \times h_i}$ is a parameterised convolution kernel, where h_i is the corresponding window size. The computation of filter f_i on k-th window is as follows:

$$r_i^k = g(f_i \mathbf{w}_{k:k+h_i-1} + b) \tag{2.5}$$

where g is the non-linear activation function, b is a scalar, and r_i^k is the feature. Then the window shifts in sequence and obtains a series of features. The features are concatenated to and pooled (such as max-pooling or mean-pooling) to get the representation:

$$\begin{aligned} r_i &= [r_i^1; r_i^2; ...; r_i^{s-h_i+1}] \\ \hat{r}_i &= Pool(r_i) \end{aligned} \tag{2.6}$$

where ; denotes the concatenation operation, the stride of the shift can also be adjusted. The output features are usually concatenated when multiple filters are introduced to get the final representations.

Recurrent Neural Networks

As sequence-based signals, natural language expression is influenced by previous history. Most inference and reasoning tasks over texts also depend closely on the context. However, this dependence is not modelled well by either FFN or CNN, which leads to the development of Recurrent Neural Networks (RNN). The idea behind RNN is to consider the history information during the representation learning process of the current word/sentence. The basic architecture of RNN is presented in Figure 2.3.

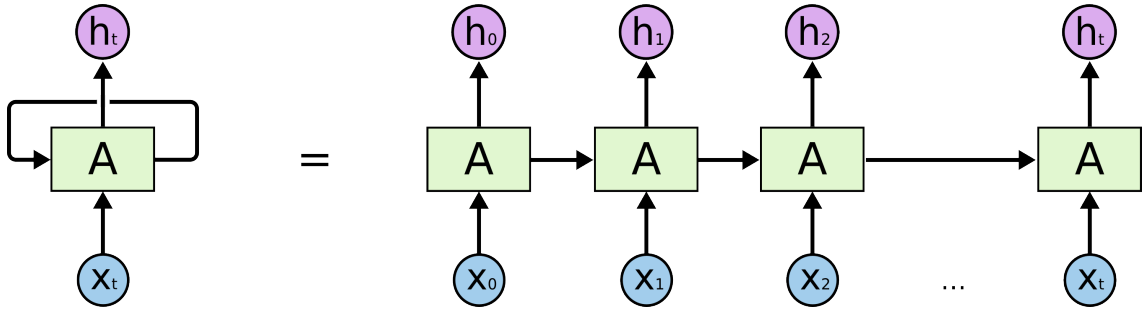


Figure 2.3. Illustration of the basic RNN structure. The figure is adapted from the blog *Understanding LSTM Networks*.

As illustrated, the RNN performs computation sequentially on the text and accepts a memory of history from the output of the last time step. This structure corresponds to the nature of the text and allows the context information to pass through the sequence. Current most popular RNN models are two of its variants: Long-Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) [60], where LSTM is designed to solve long-term dependencies and

gradient vanishing problems, and GRU consists of less training parameters. We introduce LSTM in detail as it is widely used in many NLP tasks.

Though the vanilla RNN structure performs well on many context-dependent tasks, it has two limitations: (a) The current task is not equally relevant to all history, while previous RNN structures handle different parts of the context information in a unified manner; (b) During the training process of RNN, the long distance of back-propagation through the time sequence easily leads to gradient vanishing problems. Considering the above problems, Hochreiter et al. [61] propose LSTM, which is depicted in Figure 2.4, where each green block denotes the LSTM at a time step. Each yellow block represents a neural network part of LSTM, and each pink block denotes an operation.

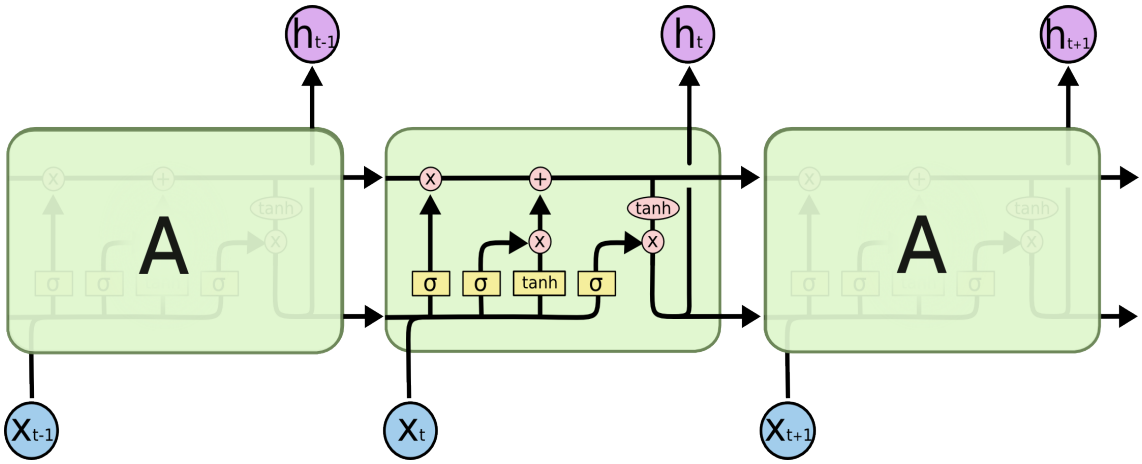


Figure 2.4. Illustration of the LSTM structure. The figure is adapted from the blog *Understanding LSTM Networks*.

LSTM introduces three gates to enable the model to memorise critical long-term contexts and remove redundant information, where each gate is responsible for an objective. First, a forget gate is proposed to determine the excluded information from memory. Specifically, the forget gate takes the hidden state of last time step h_{t-1} and the current input x_t as input and computes the forget co-efficient as follows:

$$o_f = \sigma(W_f[x_t; h_{t-1}] + b_f) \tag{2.7}$$

where W_f and b_f are learnable parameters, σ denotes the sigmoid activation function. Another input gate is leveraged to determine the new information to be included in the memory. The input co-efficient o_i is obtained via a similar computation process to Eqn. 2.7, and we filter the candidate vectors from the memory and inputs:

$$c_t = tanh(W_c[x_t; h_{t-1}] + b_c) \tag{2.8}$$

where W_c and b_c are learnable parameters and $tanh$ denotes the Tanh activation function. Then we update the memory by considering both the forget and input gate, where the forget gate filters the hidden states of the previous time step and the input gate filters the candidate states:

$$\hat{c}_t = o_f * \hat{c}_{t-1} + o_i * c_t \tag{2.9}$$

where $*$ denotes the element-wise multiplication operation. Finally, an output gate is devised to determine the aspects to be output at the current time step. The output co-efficient o_t is computed via a similar procedure to Eqn. 2.7, and the final output is computed as follows:

$$h_t = o_t * tanh(\hat{c}_t) \tag{2.10}$$

In addition, Schuster et al. [62] notice that the current task can also benefit from future contexts. To introduce both past and future contextual information, they design two LSTMs to walk through the text sequence from left-to-right and right-to-left and concatenate both outputs at the corresponding time step, which is denoted as bi-directional LSTM:

$$\hat{h}_t = [\overleftarrow{h}_t; \overrightarrow{h}_t] \tag{2.11}$$

where $\overleftarrow{h}_t$ and $\overrightarrow{h}_t$ denote the output of left-to-right and right-to-left LSTM at time step t. LSTM dynamically manages the short- and long-term memory and prevents the gradient vanishing problem, which makes it perform well compared to other RNN structures. It is widely utilised in many NLP tasks, such as text classification [63], [64], text generation [65], [66], machine translation [67], [68] and recommendation systems [69], [70].

The Transformer

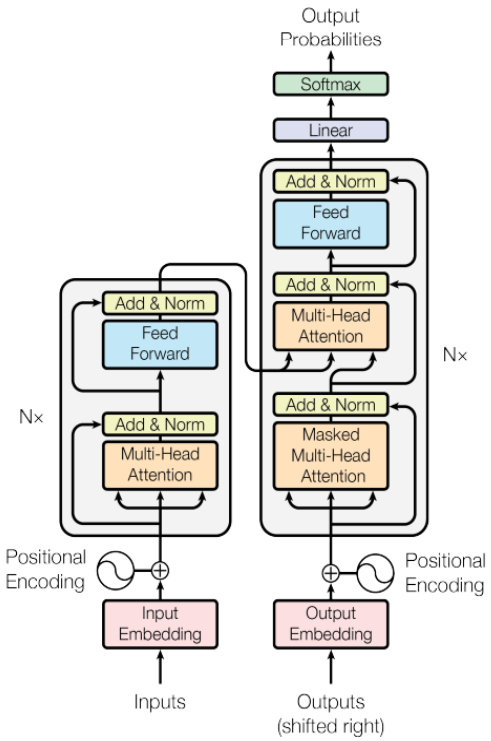


Figure 2.5. Illustration of the Transformer structure. The figure is adapted from Vaswani et al. [71].

Pre-trained word representations are widely leveraged in most previous introduced neural networks to initialise the embedding look-up table. A significant limitation of these static embeddings is that their real-valued vectors remain the same in different contexts. However, the semantics of a word can alter as the context changes. For example, the word "apple"

represents a fruit in the sentence "An apple a day, keep the doctors away", while it denotes a technology company in the sentence "Steve Jobs is the founding father of Apple". Another limitation of CNN and RNN is that they only allow direct interaction of neighbouring signals. In CNN, direct aggregation is performed within the context window of the filter, while the reasoning between long-range signals requires multi-layer convolution. In RNN, modelling long-range dependency relies on the message passing of hidden states, while crucial information can be lost during this process.

Considering the above limitations, Vaswani et al. [71] propose a novel neural network architecture called the Transformer, where the model overview is presented in Figure 2.5. As the Transformer was first applied to machine translation, it consists of an encoder and decoder parts. The encoder and decoder share a randomly initialised embedding look-up table, and a position embedding is directly summed with the word embeddings. Precisely, the position embeddings are calculated as follows:

$$\begin{aligned} PE_{(pos,2i)} &= sin(pos/10000^{2i/d_{model}}) \\ PE_{(pos,2i+1)} &= cos(pos/10000^{2i/d_{model}}) \end{aligned} \tag{2.12}$$

where pos denotes the absolute position of the word, i denotes the dimension, and d_{model} denotes the dimension of the word representations.

The encoder consists of N identical layers, where the critical component is the multi-head attention, which is based on the scaled dot-product attention. Their structures are illustrated in Figure 2.6.

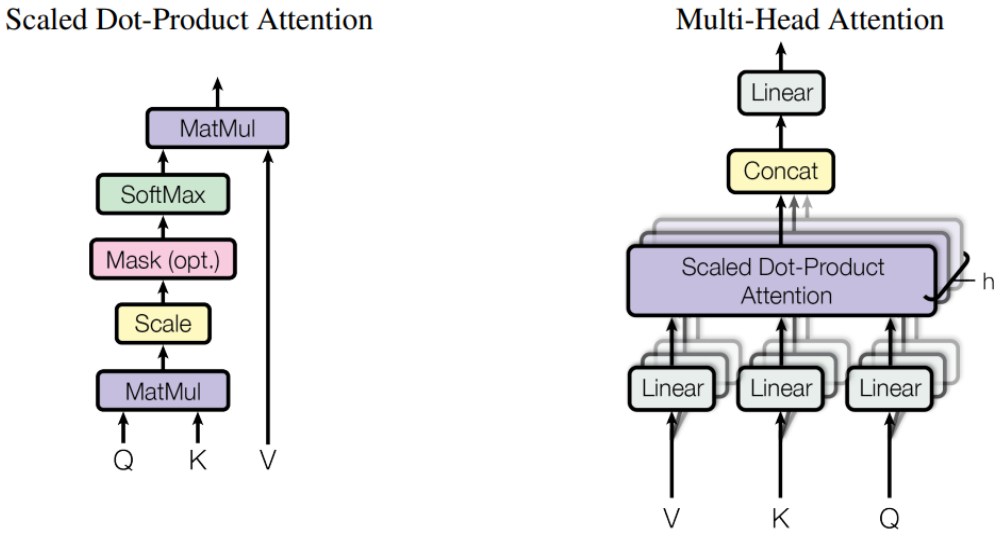


Figure 2.6. Illustration of the scaled dot-product attention and multi-head attention. The figure is adapted from Vaswani et al. [71].

As shown in the left part of Figure 2.6, the scaled dot-product attention takes the query (Q), keys (K) and values (V) as input. First, a matrix multiplication operation is performed between Q and K, and the results are divided by $\sqrt{d_q}$ (the scaling operation), where d_q denotes the dimension of query and key vectors. For large values of d_q, the dot-product result also

becomes large in magnitude, which leads the softmax into regions with minimal gradients. The authors introduce the scaling operation to alleviate this problem. Then a softmax is computed to obtain the normalised weights on the values. Finally, the weights are summed on V to get the output. In practice, the operations are packed into matrices, where $\mathbf{Q} \in \mathbb{R}^{B \times d_q}$, $\mathbf{K} \in \mathbb{R}^{B \times n \times d_q}$ and $V \in \mathbb{R}^{B \times n \times d_v}$. B denotes the batch size, n denotes the number of keys and values, and d_v denotes the dimension of the value vectors. This process is formalised as:

$$att(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = Softmax(\frac{\mathbf{Q}^\top \mathbf{K}}{\sqrt{d_q}})\mathbf{V} \tag{2.13}$$

CNN walks the filters sequentially on the input signals, which aims to perceive features from different parts. Inspired by the success of CNN, the Transformer also employs a multi-head attention mechanism, which enables the model to jointly attend to features from multiple representation subspaces at different positions. As shown in the right part of Figure 2.6, instead of computing a single dot-product attention with a d_{model}-dimensional keys (d_{model} denotes the dimension of the original input representations), h heads are computed, where for each head, a set of FFN networks are used to linearly transform the Q,K,V pairs into a d_q, d_q and d_v dimensions. Then h attention operations are performed in parallel, which results in h different outcomes. The outcomes are concatenated and linear projected as the final output. To facilitate the linear projection and reduce the computational cost, the dimensions are normally set to $d_v = d_{model}/h$. This process is formulated as follows:

$$\begin{aligned} MultiHead(\mathbf{Q}, \mathbf{K}, \mathbf{V}) &= Concat(head_1, ..., head_h)W^O \\ where\ head_i &= att(\mathbf{Q}W_i^Q, \mathbf{K}W_i^K, \mathbf{V}W_i^V) \end{aligned} \tag{2.14}$$

where $Concat$ denotes the concatenation operation, $W^O \in \mathbb{R}^{hd_v \times d_{model}}$, $W_i^Q \in \mathbb{R}^{d_{model} \times d_q}$, $W_i^K \in \mathbb{R}^{d_{model} \times d_q}$, $W_i^V \in \mathbb{R}^{d_{model} \times d_v}$ are learnable parameters.

In Transformer encoders, we employ the multi-head attention in a self-attention manner, where $\mathbf{Q} = \mathbf{K} = \mathbf{V}$, and $d_q = d_v$. The input of a layer comes from the output of the last layer, and each position in the current layer can attend to all positions in the last layer. In addition, residual connections (also known as skip connections) are widely used in CV and achieve outstanding performance [72]. It is proved especially effective in deep neural networks since it mitigates the degeneration problem as the layers increase and avoid the problem of gradient vanishing, which facilitates the training of models with deep layers. Therefore, the Transformer also introduces residual connections to the structure. For the multi-head attention, the residual connection is added as follows:

$$x_i = LayerNorm(\hat{x}_{i-1} + MultiHead(\hat{x}_{i-1}, \hat{x}_{i-1}, \hat{x}_{i-1})) \tag{2.15}$$

where $\hat{x}_{i-1}$ denotes the output of layer $i-1$, and $LayerNorm$ denotes the layer normalisation operation. Then the outputs pass through a point-wise FFN and another residual connection

to form the output of the i-th layer:

$$\begin{aligned} x_i^{'} &= max(0, x_i W_1 + b_1) W_2 + b_2 \\ \hat{x}_i &= LayerNorm(x_i + x_i^{'}) \end{aligned} \tag{2.16}$$

where W_1, W_2, b_1, b_2 are learnable parameters, and the output of N-th layer is used for decoding.

In the decoder part of the Transformer, the basic blocks are similar to their counterparts in the encoder, except that an interaction module is inserted between the multi-head attention and FFN modules, which works as the function of the typical encoder-decoder attention mechanisms in previous sequence-to-sequence models. Specifically, the module performs standard multi-head attention with the encoder output $\hat{x}_N$ as the K, V, and the output of the last multi-head attention module y_i in the decoder as Q, which is depicted as follows:

$$y_i^{'} = LayerNorm(y_i + MultiHead(\hat{x}_N, \hat{x}_N, y_i)) \tag{2.17}$$

In the decoder, the leftward information flow is prohibited as the expected auto-regressive property. Therefore, in the scaled dot-product attention, all values in the right of the current query are masked with $-\infty$ at the input of the softmax (see Figure 2.6).

Compared to CNN and RNN, a vital feature of the pure attention-based Transformer is that the representation of each position is allowed to interact with all positions directly, facilitating long-range reasoning and message passing. Multi-head attention also enables the model to focus on different input subspaces. These architectures equip the Transformer with more vital context modelling ability. Unlike RNN, which requires a sequential encoding process, the Transformers takes in and encodes the input at once, which makes it easy to deploy and optimise on the hardware (such as a GPU) to perform computation in parallel. The above advantages set the foundation for its application in broader scenarios and the later success of Transformer-based pre-trained language models.

Transformers-Based Pre-trained Language Models

Pre-trained Language Models (PLMs) [73] utilise appropriate neural networks and is trained on large-scale datasets in a supervised or unsupervised manner, which aims to learn valuable patterns and knowledge. Then the pre-trained weights are transferred to downstream tasks. One branch of works applies pre-trained representations as features [56], [57], [74], and design task-specific architectures for each downstream task. Another line of work introduces minimal task-specific parameters and fine-tunes all pre-trained parameters on the downstream tasks [49], [51], [75]. The model architecture is also crucial for performance. Early works utilise the FFN to pre-train word representations [56], [57]. Other works construct the model with the stack of RNN and obtain superior performance than the FFN [74], [76]. In recent years, the Transformer-based PLMs significantly outperform other neural architectures in most NLP tasks [49], [51], [75], and become the mainstream in NLP research.

Significantly, the work "Pre-training of Deep Bidirectional Transformers for Language Understanding", also known as BERT [49], is pioneering and most influential in this line of work, which we introduce in detail.

Most finetuning-based PLMs are pre-trained in an auto-regressive manner, where each token can only attend to previous tokens [75]. However, the negligence of leftward information flows limits many downstream tasks that rely on future contexts. Therefore, Devlin et al. [49] propose BERT, which builds the PLM with the Transformer encoder and attends to all contexts for each token. The main components of BERT are presented in Figure 2.7, where the same architecture is used for both stages, and the same pre-trained weights are leveraged to initialise for different downstream tasks. BERT is composed of N stacked layers of standard Transformer encoder structures, and the input of each layer is the output of the last layer's Transformer.

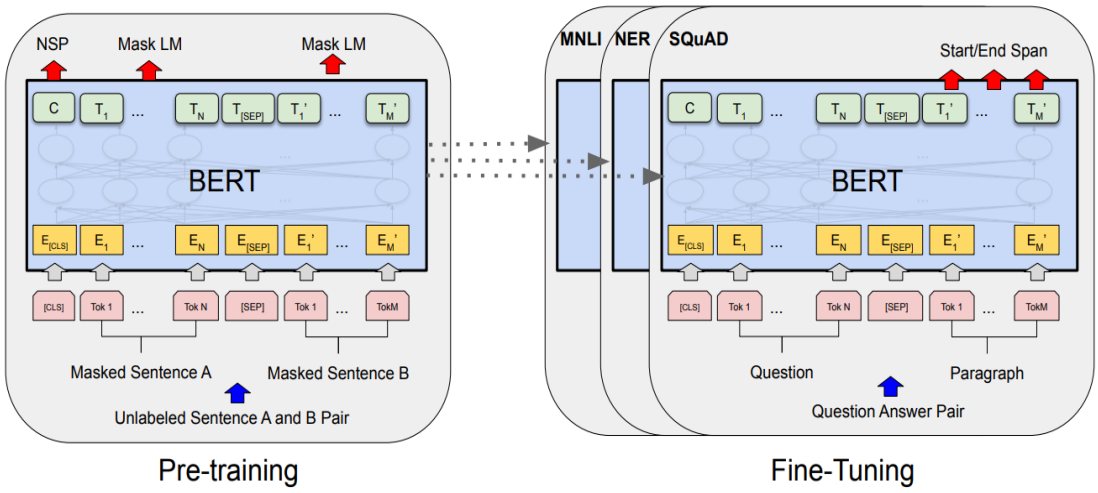


Figure 2.7. Illustration of the BERT pre-training and fine-tuning stage. The figure is adapted from Devlin et al. [49].

In the pre-training stage, each input sentence is tokenised and projected into word embeddings, where the look-up table is randomly initialised. Another set of embeddings is also randomly initialised to embed the sentence segment. The input representation is constructed by summing the corresponding word, segment and position embeddings. Then the input embeddings pass through the encoder to obtain the word-level representations. BERT is pre-trained with two tasks: Masked Language Model (MLM) and Next Sentence Prediction (NSP). MLM is inspired by the Cloze task, where 15% of all tokens within the dataset are masked at random (the token is replaced by "[MASK]"). The model is trained to reconstruct the masked tokens and optimised with cross-entropy loss. However, there is a mismatch between pre-training and fine-tuning as the "[MASK]" token does not appear during fine-tuning. Therefore, the task replaces the chosen token with: (a) The "[MASK]" token 80% of the time; (b) A random token 10% of the time; (c) The unchanged token 10% of the time. The NSP task randomly selects a series of adjacent sentences A and B from the corpus. 50% of the sentences B is replaced with a random sentence in the corpus, and 50% of B is unchanged. The model is trained to predict whether the sentence B is the following sentence of A and optimised with binary cross entropy loss.

In the fine-tuning stage, a task-specific output layer (usually an FFN with a small number of parameters) is put on top of the BERT output representations. All weights are fine-tuned on the target dataset. For example, in text classification tasks, a common approach is to build the output classification layer on top of the representations of the start-of-sentence token "[CLS]". These fine-tuning strategies lead BERT to achieve state-of-the-art performance on 11 NLP tasks [49]. The outstanding performance of BERT also started a revolution in NLP research paradigms.

After BERT, there are many works to improve the PLMs. For example, RoBERTa [50] was pre-trained on more data and discards the NSP pre-training task. XLNet [51] introduces segment recurrence to enable lengthy text processing and re-introduces auto-regressive training to BERT while allowing future contexts. These approaches advance the performance of Transformer-based PLMs. PLMs have become the foundation of most NLP research, and our works depend on the strong representations learnt by these PLMs.

2.1.2 Contrastive Learning

In Sec. 2.1.1, we have introduced the mainstream neural networks utilised for representation learning. Most of these methods use task-specific loss functions (e.g. the cross entropy loss) to directly train on the target datasets or leverage the learnt representations from pre-trained word embeddings or PLMs to introduce extra information. The former requires large-scale data to learn decent representations while performing poorly in low-data resource scenarios. The pre-training process of modern large-scale Transformer-based PLMs requires computational resources that are not affordable for most institutes. For example, the powerful GPT-3 model has 175 billion parameters and requires 800 GB of storage. A single training process costs 4.6 million dollars and 355 GPU years[1]. Though the weights of many PLMs have been released for free by organisations such as Huggingface[2], the PLMs can still perform poorly in low-resource tasks without proper fine-tuning. Therefore, methodologies other than target-oriented training are developed to enhance representation learning further.

One of the most successful methods is contrastive learning (CL). It is also inspired by human learning paradigms and shows promising results in many artificial intelligence areas (e.g. NLP, CV) under the deep learning framework. CL aims to enhance representation learning by using contrastive samples against each other to learn common features among data clusters and those that set apart each other. This basic idea makes CL a part of deep metric learning [77]. For implementation, CL designed a contrastive loss and trained a model to learn representations of input signals. Similar samples lie closer in the representation space while different samples fall apart. Specifically, for two positive pairs and their representations z_i and z_j within the batch N where the rest are negative pairs, the contrastive loss is implemented as follows:

$$\mathcal{L}_{i,j} = -log \frac{exp(sim(z_i, z_j)/\tau)}{\Sigma_{k=1}^{N} exp(sim(z_i, z_k)/\tau)} \ (k \neq i) \tag{2.18}$$

[1] https://en.wikipedia.org/wiki/GPT-3
[2] https://huggingface.co/

where $sim()$ is the similarity metric function which is usually the dot product operation, τ is a pre-defined temperature co-efficient. This loss is a revised version of the cross entropy loss and is minimised to encode positive samples in similar representations and negative samples in different representations. The critical process of CL is mining positive example pairs, which can be implemented in a supervised or unsupervised manner. Therefore, CL is divided into Unsupervised Contrastive Learning (UCL) and supervised contrastive learning, where UCL mainly constructs positive pairs via various data augmentation methods and SCL mines positive pairs according to the existing labels of the data. We introduce both methods in detail in the following two sections.

Unsupervised Contrastive Learning

Unsupervised Contrastive Learning (UCL) aims to construct training samples in an unsupervised manner. Each sample has only one positive pair obtained by data augmentation, and negative pairs are randomly sampled from the dataset. CL is trained with the loss function in Eqn. 2.18. UCL was first applied to CV. A representative work is SimCLR [38], which sequentially applies three stochastic data augmentation methods to obtain the positive pair: (a) Random cropping followed by resizing back to the original size; (b) Random colour distortion; (c) Random Gaussian blur. The thorough experiments on image classification prove the effectiveness of UCL. Further analysis shows several key properties that benefit the following works: (a) Composition of data augmentation operation is crucial for learning good representations; (b) UCL needs stronger data augmentation than normal supervised learning; (c) UCL benefits more from larger models; (d) CL benefits more from larger batch sizes and more training. Based on SimCLR, more works improve UCL on CV from different perspectives. For example, Li et al. [39] propose contrastive clustering to produce clustering-favourite representations, which regard each classification class as a cluster and obtain positive pairs from two different data augmentation methods. Then contrastive learning is performed on both instance and cluster levels. There are also many attempts to reduce the high-dimensional UCL space to incorporate prior knowledge [78], [79], boost semi-supervised learning [80] and visualise the results [81].

With a similar training framework in NLP, UCL is mainly devised to enforce the sentence representations of PLMs to distinguish similar semantics. A representative work is SimCSE [41], where the main structure is presented in Figure 2.8. The left part denotes the structure for the unsupervised setting, and the right part denotes the structure for the supervised setting. As shown in the left part, the unsupervised SimCSE simply passes the same sentence to the encoder twice with the standard dropout operation. These two different embeddings are used as positive pairs. It is viewed as a minimal form of data augmentation as the positive pairs only differ in dropout masks. Other samples within the same mini-batch are regarded as negative pairs. CL is trained with the loss function in Eqn. 2.18. The authors compare the dropout operation with other data augmentation techniques such as crop, word deletion and replacement on the semantic textual similarity task, and the result proves the effectiveness of the dropout operation. Further analysis explains that SimCSE can keep a

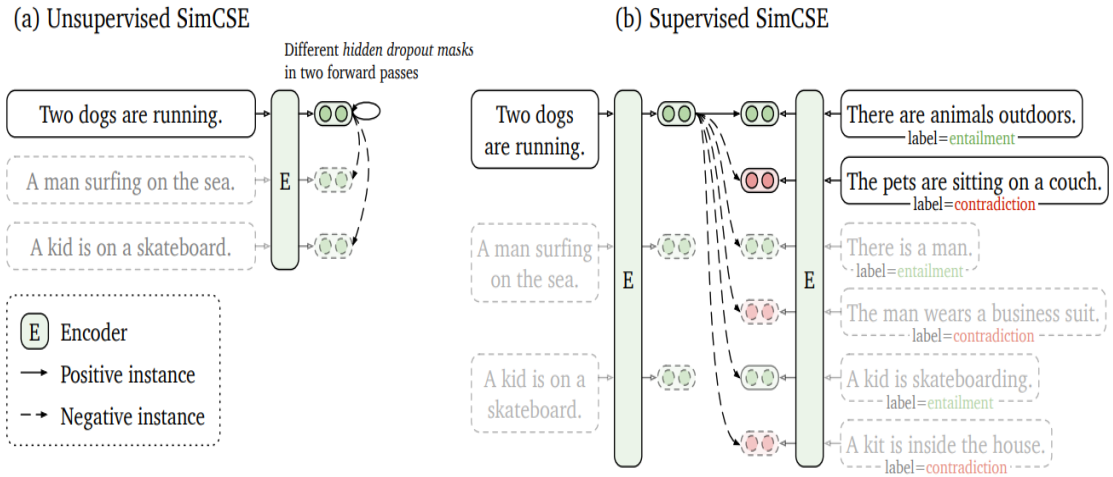


Figure 2.8. Illustration of the SimCSE for contrastive learning. The figure is adapted from Gao et al. [41].

steady alignment thanks to the use of dropout noise, which does not change the semantics of the sentence.

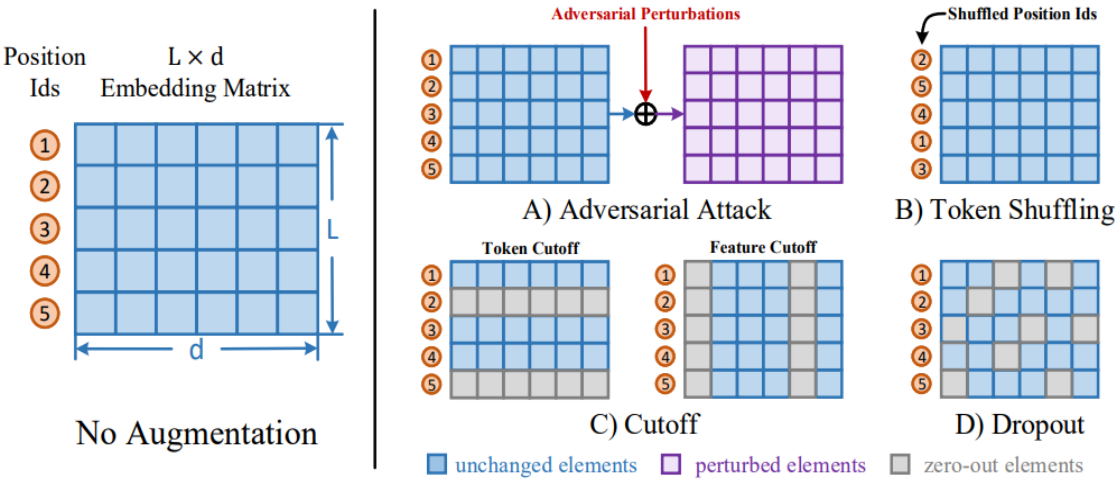


Figure 2.9. Illustration of the four data augmentation methods for contrastive learning. The figure is adapted from Yan et al. [42].

Later UCL works mainly focus on developing better techniques to obtain positive pairs. Apart from dropout, Yan et al. [42] develop several new data augmentation methods, as presented in Figure 2.9: (a) adversarial attack: generate adversarial samples by adding a worst-case perturbation to the input sample; (b) token shuffling: randomly shuffle the token orders of the input sentence; (c) cut-off: randomly discard some tokens, feature dimensions or token spans in the feature matrix. Giorgi et al. [82] did not use the data augmentation methods but regarded textual segments sampled from nearby in the same document as positive pairs. Kim et al. [83] trained a Siamese model to construct positive pairs. However, the copy of PLMs such as BERT leads to more costs in model storage and computation.

Supervised Contrastive Learning

Supervised contrastive learning introduces supervised learning to self-supervised contrastive methods by leveraging label information. A key difference from UCL is that several samples instead of one are considered positive pairs in SCL. An example in CV is shown in Figure 2.10, where the UCL contrasts a single positive pair while SCL can have multiple positive pairs from the same class for each sample. Self-supervised contrastive learning constructs a single positive pair from each target sample (also known as "anchor" in CV) and randomly samples negative pairs, while SCL contrasts the sets of all samples with the same label to the target as positive pairs and those with different labels as negative pairs.

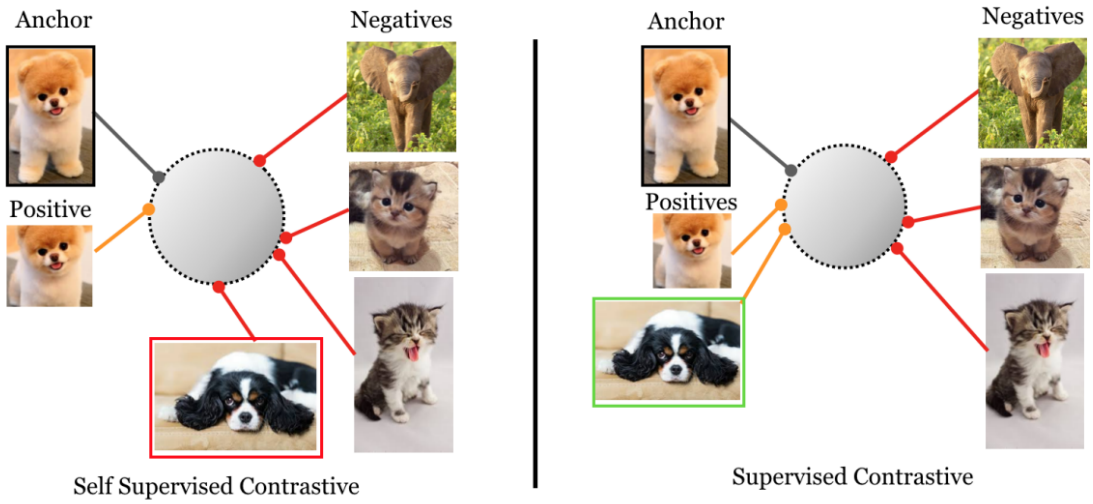


Figure 2.10. The training of self-supervised (unsupervised) and supervised contrastive learning. The figure is adapted from Khosla et al. [40].

With this expansion, the loss of SCL needs to be generalised to arbitrary numbers of positive pairs. One of the widely used modifications is as follows:

$$\mathcal{L}_i^{SCL} = -\frac{1}{|P(i)|}\sum_{p\in P(i)} log\frac{sim(z_i, z_j)/\tau}{\Sigma_{a\in A(i)} exp(sim(z_i, z_a)/\tau)} \tag{2.19}$$

where $A(i)$ is the sampled mini-batch, and $P(i) = \{p|p \in A(i); y_p = y_i\}$ where y_i denotes the label of i-th sample. Some other works also put the log function outside:

$$\mathcal{L}_i^{SCL} = -log\left(\frac{1}{|P(i)|}\sum_{p\in P(i)} \frac{sim(z_i, z_j)/\tau}{\Sigma_{a\in A(i)} exp(sim(z_i, z_a)/\tau)}\right) \tag{2.20}$$

In CV, experiments show that SCL performs better than traditional cross entropy loss in applications such as image classification [40]. In NLP, an early SCL work is still SimCSE, where the structure is presented in the right part of Figure 2.8. As illustrated, SimCSE incorporates supervision signals from natural language inference datasets, which predicts whether the relationship between two sentences is entailment, contradiction or neutral. During training, the entailment sentences are used as positive pairs, and the contradiction and neutral pairs are used as negative pairs. The experiments show the advantage of adding these supervision

signals over the unsupervised SimCSE model.

SCL is also successfully applied to fine-tuning PLMs. Gunel et al. [43] combine an SCL loss to the cross entropy loss during the fine-tuning of BERT on the single sentence and sentence-pair classification tasks. For single-sentence classification tasks, the sentences with the same label are considered positive pairs (such as the samples with the sentiment "positive" in sentiment analysis). In sentence-pair classification tasks, two sentences s_1 and s_2 are concatenated: $s = [CLS; s_1; SEP; s_2]$, where CLS is the start-of-sentence token and SEP is the separation token in BERT. s is input to BERT, and sentences with correct relationships are considered positive pairs. During training, the SCL loss is combined with the cross entropy loss in a multi-task learning manner:

$$\mathcal{L} = \alpha \mathcal{L}_{CE} + (1 - \alpha)\mathcal{L}_{SCL} \tag{2.21}$$

where α is the co-efficient that controls the weights of the two losses, $\mathcal{L}_{CE}$ is the cross entropy loss and $\mathcal{L}_{SCL}$ is the SCL loss. This multi-task learning paradigm enables SCL to enforce the traditional cross entropy classification process and is extended to many other tasks. In ERC, Li et al. [84] utilise SCL to distinguish sentences and emotions with similar semantics, and also combine the SCL loss with an utterance reconstruction loss in a multi-task learning setting. Alhuzali and Ananiadou [85] also introduce a centre loss apart from the triplet loss (a variation of SCL loss), which pushes close samples from the same class towards the corresponding centre and combines both intra- and inter-class variations into the emotion classification loss function.

2.1.3 Knowledge-Enhanced Methods

Most of the inferences and decisions humans make rely on previous experiences and knowledge. The knowledge is usually expected to be possessed by most people in their daily communications [86], and helps them make sense of everyday situations. For example, in the scenarios of daily dialogue, one participant could ask: "Where is the headquarter of Apple?". It requires the critical commonsense knowledge that "Apple is a technology company" and "Apple Park is the headquarters of Apple Inc., located in Cupertino, California, United States"[3] to answer the question correctly. Similarly, lack of knowledge often presents a challenge for many NLP tasks, especially when not all necessary information is available in the processed text. Therefore, enhancing representation learning with external knowledge has been a primary focus of the NLP research community. Two crucial aspects of related research are knowledge sources and knowledge infusion methods. Research on knowledge sources mainly focuses on constructing knowledge structures suitable to combine in representation learning, whereas the representative works include knowledge graphs, generative knowledge sources, sentiment lexicons, etc. Research on knowledge infusion tries to develop appropriate methods to infuse knowledge, where the NLP models can quickly leverage them for inference. Current works either infuse knowledge explicitly or implicitly. In the following sections, we

[3] https://en.wikipedia.org/wiki/Apple_Park

introduce state-of-the-art methods in these two directions separately.

Knowledge Sources

One of the most important types of knowledge is commonsense knowledge. Obvious examples of their applications include commonsense question answering, dialogue systems, natural language inference and text generation [87]. Many text classification tasks, such as sentiment analysis, require external knowledge to understand the context and special semantics, such as irony. A good commonsense knowledge source for NLP requires representing commonsense knowledge in a machine-readable form. Considering the features of commonsense knowledge, a natural choice is to store it in the triplet form, where a pre-defined relation connects two entities. For example, the knowledge that "David Bowie is an English singer" is represented in a triplet <David Bowie, occupation, singer> and <David Bowie, nationality, English>. A knowledge graph is constructed with the entities as nodes and the pre-defined relations as edges. Representative works in this line include ConceptNet [45], ATOMIC [47] and WebChild [88]. We list several knowledge graphs widely used in various tasks in Figure 2.11, where "Relations" denotes the pre-defined relation types in the knowledge source. Except on the phrase level, there are also knowledge graphs built on other granularity, such as the sentence-level knowledge graph CICERO [89], which provides commonsense knowledge for dialogue-level reasoning and inference.

Category	Source	Relations	Example 1	Example 2
Commonsense KGs	ConceptNet*	34	*food - capable of - go rotten*	*eating - is used for - nourishment*
	ATOMIC	9	*Person X bakes bread - xEffect - eat food*	*PersonX is eating dinner - xEffect - satisfies hunger*
	GLUCOSE	10	*Someone$_A$ makes Something$_A$ (that is food)*	*Causes/Enables Someone$_A$ eats Something$_A$*
	WebChild	4 (groups)	*restaurant food - quality#n#1 - expensive*	*eating - type of - consumption*
	Quasimodo	78,636	*pressure cooker - cook faster - food*	*herbivore - eat - plants*
	SenticNet	1	*cold_food - polarity - negative*	*eating breakfast - polarity - positive*
	HasPartKB	1	*dairy food - has part - vitamin*	*n/a*
	Probase	1	*apple - is a - food*	*n/a*
	Isacore	1	*snack food - is a - food*	*n/a*
Common KGs	Wikidata	6.7k	*food - has quality - mouthfeel*	*eating - subclass of - ingestion*
	YAGO4	116	*banana chip - rdf:type - food*	*eating - rdfs:label - feeding*
	DOLCE*	1	*n/a*	*n/a*
	SUMO*	1,614	*food - hyponym - food_product*	*process - subsumes - eating*
Lexical resources	WordNet	10	*food - hyponym - comfort food*	*eating - part-meronym - chewing*
	Roget	2	*dish - synonym - food*	*eating - synonym - feeding*
	FrameNet	8 (f2f)	*Cooking_creation - has frame element - Produced_food*	*eating - evoke - Ingestion*
	MetaNet	14 (f2f)	*Food - has role - food_consumer*	*consuming_resources - is - eating*
	VerbNet	36 (roles)	*feed.v.01 - Arg1-PPT - food*	*eating - hasPatient - comestible*

Figure 2.11. A brief summary of commonsense knowledge graphs. The figure is adapted from Ilievski et al. [90].

As an example, we explain the building process of ConceptNet [45] in detail. ConceptNet is a knowledge graph that connects concepts (natural language words and phrases) with weighted edges (assertions), which is expected to include world knowledge from many different sources in multiple languages. It represent relations between concepts such as "A *net* is used for *catching fish*" and "*leaves* is a form of the word *leaf*". Specifically, ConceptNet 5.5 is built from the following sources: Facts from Open Mind Common Sense (OMCS) [46] and its sister projects in other languages, Wiktionary in multiple languages, the multilingual WordNet [91], the Japanese dictionary JMDict [92], the knowledge base OpenCyc [93], and

a subset of the factual knowledge base DBPedia [94].

ConceptNet 5.5 aligns the knowledge sources on 36 kinds of relation, which are listed as follows: *Antonym*, *DistinctFrom*, *EtymologicallyRelatedTo*, *LocatedNear*, *RelatedTo*, *SimilarTo*, *Synonym*, *AtLocation*, *CapableOf*, *Causes*, *CausesDesire*, *CreatedBy*, *DefinedAs*, *DerivedFrom*, *Desires*, *Entails*, *ExternalURL*, *FormOf*, *HasA*, *HasContext*, *HasFirstSubevent*, *HasLastSubevent*, *HasPrerequisite*, *HasProperty*, *InstanceOf*, *IsA*, *MadeOf*, *MannerOf*, *MotivatedByGoal*, *ObstructedBy*, *PartOf*, *ReceivesAction*, *SenseOf*, *SymbolOf*, and *UsedFor*.

All the above methods of constructing knowledge graphs are extractive and store knowledge with canonical templates, which lack flexibility and do not include a large amount of world knowledge in natural language. Therefore, Bosselut et al. [54] explore the development of generative commonsense knowledge models. The proposed method COMET leverages existing commonsense knowledge graphs as the seeds and trains a Transformer-based PLM on them, which aims to enable the PLM to adapt its representations to knowledge generation and produce novel knowledge triplets in natural language.

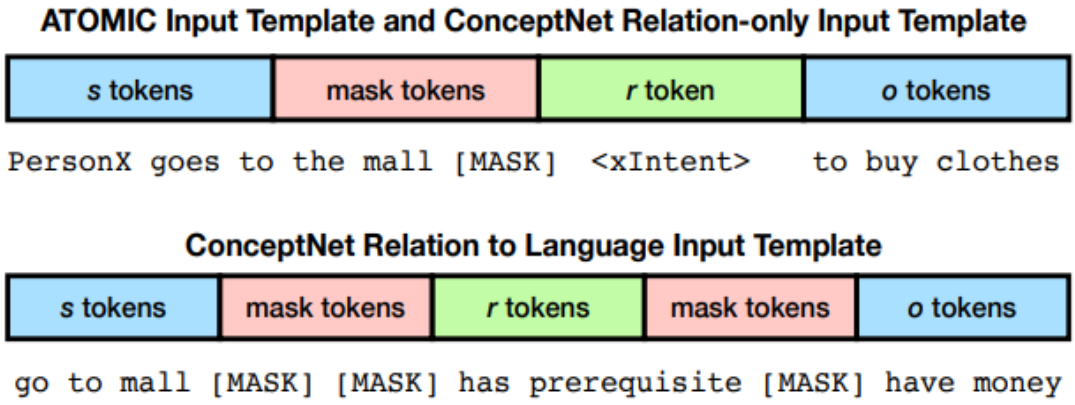


Figure 2.12. The input token setup for the two knowledge sources. The figure is adapted from Bosselut et al. [54].

Specifically, COMET leverages the GPT-2 [75] language model as the generative model and trains the model on two knowledge graphs: ConceptNet and ATOMIC [47]. As the input to the model, each triplet is concatenated to a sequence of words, as shown in Figure 2.12. For ATOMIC and relation-only input of ConceptNet, the first phrase is followed by the "[MASK]" token, then followed by the relation token (such as xIntent) and the second phrase. Since the relation of some items in ConceptNet has more than one token, another input format is used where two "[MASK]" tokens are used between the first phrase and the relation, and another "[MASK]" is asserted between the relation and the second phrase.

During training, the COMET learns to generate the second phrase e_2 of the triplet given the first phrase e_1 and the relation token r. Specifically, with the concatenated tokens of e_1 and r as in Figure 2.12, the model is trained to generate all tokens of e_2. COMET is optimised

to maximise the conditional log-likelihood of predicting e_2:

$$\mathcal{L} = -\sum_{|e_1|+|r|}^{|e_1|+|r|+|e_2|} log\ P(x_t|x_{\leq t}) \tag{2.22}$$

where $|e_1|$, $|r|$ and $|e_2|$ are the number of tokens in e_1, r and e_2. Empirical studies on the quality, novelty and diversity of the newly produced triplets show that COMET can generate high-quality commonsense knowledge since human evaluation proves that 77.5% of ATOMIC's generated tuples and 91.7% of ConceptNet's generated tuples are correct. The success of generative commonsense knowledge sources facilitates the discovery of new knowledge and the construction of new knowledge graphs. It also inspires more ways of incorporating knowledge into the representations.

The above-introduced knowledge sources mainly consist of factual knowledge for common use in NLP tasks. However, other knowledge sources also provide common sense for certain aspects. As our works mainly focus on emotion-related tasks, we introduce knowledge sources specially designed to facilitate sentiment analysis, known as sentiment lexicons. The most widely used sentiment lexicons are SenticNet [95] and SenticWordNet [96]. SenticWordNet is a lexical resource where each WordNet [91] synset is assigned three scores ranging from 0 to 1. These scores show how objective, positive and negative the terms in synset are. The three scores are derived from the results of a committee of ternary classifiers.

SenticWordNet is widely utilised in sentiment-related tasks but only provides sentiment polarity at the syntactical level and contains much noise. Considering these limitations, Cambria et al. [95] develop SenticNet, which aims to construct a collection of sentiment polarity for phrase-level concepts such as "look attractive" and "good deal". They discard concepts without strong emotions and only associate each concept with one value p_c ranging from -1 to 1, quantifying its sentiment polarity from very negative to very positive. The computing of the scores is mainly based on the Hourglass of Emotions [97], where four affective dimensions are considered: Pleasantness (*Plsn*), Attention (*Attn*), Sensitivity (*Snst*) and Aptitude (*Apti*). Each dimension is further defined by six activation levels (also known as sentic levels), which reflect the intensity of the expressed/perceived emotion. SenticNet computes p_c as follows:

$$p_c = \frac{|Plsn(c)| + |Attn(c)| - |Snst(c)| + |Apti(c)|}{9} \tag{2.23}$$

Another widely leveraged knowledge type is linguistic knowledge. It mainly refers to the knowledge of the grammatical structure of sentences, which establishes the relationships between headwords and words that modify the headwords. An example is shown in Figure 2.13. According to the example, the terms below the sentence denote the part-of-speech tag for each word. The arrow from the word "control" to the word "good" indicates that "good" modifies "control", and the label "amod" describes the exact nature of the dependency (e.g. "amod" denotes "good" works as the adjective of "control").

Linguistic knowledge helps the model understand the sentence structures and is helpful in

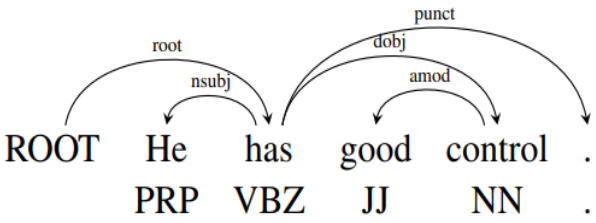


Figure 2.13. The example of a dependency parsing tree of a short sentence.

representation learning for many downstream NLP tasks such as entity typing and question answering [52]. The primary resources of linguistic knowledge are the human-labelled treebanks, such as the famous Penn Treebank[4]. With the fast development of the basic NLP technique dependency parsing, Chen et al. [98] first utilise neural networks to train on the large-scale treebanks and obtain high accuracy in labelling linguistic knowledge. Their research is developed as the widely used dependency parsing tool called Stanford Parser[5]. Nowadays, researchers do not access linguistic knowledge directly from the treebanks but run the dependency parser on the target text to obtain the parsing tree, which is easy to implement.

Knowledge Infusion Methods

Developing appropriate knowledge infusion methods is a crucial challenge of leveraging external knowledge to aid representation learning. Wrong knowledge infusion methods introduce noise to the model and affect the learnt representations. Therefore, there have been many efforts in knowledge infusion algorithms, roughly divided into explicit knowledge infusion, knowledge adapters, pretraining-based knowledge infusion and finetuning-based knowledge infusion.

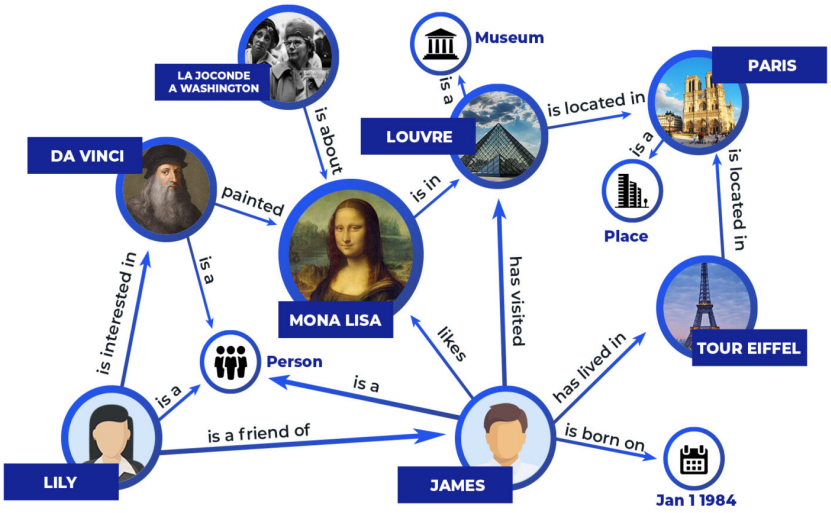


Figure 2.14. An example of knowledge graphs.

Explicit knowledge infusion designs extra modules for learning representations for the knowledge and combines them with the text representations. A representative line of work is the knowledge graph embedding methods [99], which aims to embed components of a knowl-

[4] https://catalog.ldc.upenn.edu/docs/LDC95T7/cl93.html
[5] https://nlp.stanford.edu/software/lex-parser.html

edge graph, including entities and relations, into continuous vector spaces, as the example of a knowledge graph shown in Figure 2.14. Considering its graph structure, a popular approach is utilising Graph Neural Networks (GNNs) to encode the graph. The intuition is regarding each entity as a node and each relation as an edge, then learning the representation for each node. Many GNN structures are effective for knowledge graph embedding, such as graph convolution networks [100] and Graph Attention Networks (GAT) [101]. We take GAT as an example to introduce the process. Firstly, each node i is initialised a representation $h_i^{(0)}$ either randomly or from the pre-trained word embeddings. Then a L-layer GAT is used to encode for each node. For the l-th layer, the pair-wise un-normalised attention score for each neighbour node j is computed as follows:

$$\begin{aligned} z_i^{(l)} &= W^{(l)} h_i^{(l)} \\ e_{ij}^{(l)} &= LeakyReLU\left(a_{(l)}^{\top}(z_i^{(l)}; z_j^{(l)})\right) \ (j \in \mathcal{N}(i)) \end{aligned} \tag{2.24}$$

where $W^{(l)}$ and $a_{(l)}$ are learnable parameters, $\mathcal{N}(i)$ denotes the set of node i's neighbors (node i is also regarded as its own neighbor), ; denotes concatenation and $LeakyReLU$ denotes the LeakyReLU activation function. The attention scores are then normalised by a softmax operation and used to aggregate and compute the input of the next layer:

$$\begin{aligned} \alpha_{ij}^{(l)} &= \frac{exp(e_{ij}^{(l)})}{\Sigma_{k \in \mathcal{N}(i)} exp(e_{ik}^{(l)})} \\ h_i^{(l+1)} &= \sigma\left(\sum_{k \in \mathcal{N}(i)} \alpha_{ik}^{(l)} z_k^{(l)}\right) \end{aligned} \tag{2.25}$$

where σ denotes the sigmoid activation function. After L layers of computation, we obtain an embedding for each node i: $h_i^{(L)}$. The graph-level embedding is usually computed with the pooling operation (e.g. max pooling or mean pooling). In the case of knowledge infusion, the knowledge embeddings are often concatenated with the token-level embeddings to get the knowledge-enhanced representations and jointly trained with the main task objective. The embeddings can also be trained in a self-supervised manner via techniques such as graph auto-encoders or graph contrastive learning [102].

For generative knowledge sources, the knowledge is usually combined at the sentence level. Instead of decoding the knowledge, current methods mainly concatenate the hidden representations of the knowledge encoder to the sentence representations [103], [104] or combine them in the dialogue-level graph [105].

Besides directly leveraging external knowledge to help the task, many works are devoted to knowledge infusion to the PLMs, which aims to enhance its task-specific representation learning ability. Most of these works infuse knowledge in the pre-training phase, and the main difference lies in the ways of incorporating the knowledge. One line of work infuses knowledge as input and trains the PLM to reconstruct the knowledge. For example, Sun et al. [106] input factual knowledge to the PLM in natural language format but with key parts masked. Then the PLM is trained to predict the masked words to learn the knowledge. Ke et

al. [107] explicitly sum the sentiment polarity embeddings to the input representations, and new loss functions are designed to learn sentiment-related information. Other methods do not explicitly input the knowledge but use the knowledge as supervision signals. For example, LIBERT [108] takes entity pairs as training instances to enable BERT to understand the lexical-semantic relations. Some other works avoid the high-cost pre-training by infusing the knowledge in the fine-tuning process. For example, Xie et al. [44] introduce a sentiment polarity intensity prediction task, which predicts the sentiment scores obtained from SenticNet, and the task is combined with the emotion detection main task in a multi-task learning manner during fine-tuning. Chen et al. [109] incorporate factual knowledge during the prompt construction process for prompt tuning of the PLM.

All the above works infuse knowledge to the PLMs by tuning their parameters for each knowledge source, which is inefficient considering the high cost of pre-training. Wang et al. [52] propose to add a knowledge adapter to the PLM in a plug-in manner, which takes the introduced knowledge and the PLM output of particular layers as input. During pre-training, only the parameters of the knowledge adapter are optimised, and the PLM weights are fused. This way of training reduces the computational cost to a large extent as the knowledge adapter has much fewer parameters than the PLM. Another key advantage is that the PLM avoids re-training with each knowledge source incorporated. Only a knowledge adapter is required, and the knowledge can be efficiently utilised for the PLM.

2.2 Emotion Recognition in Conversations

ERC aims at identifying emotions from a pre-defined emotion category set. A critical difference between ERC and the vanilla emotion recognition of a sentence is that the emotion of the target utterance is influenced by both previous utterances from other participants and the speaker himself, which is denoted as inter- and intra-speaker influence [21]. The complex relations in multi-party conversations bring more challenges to ERC. Therefore, most previous works in the literature focus on developing appropriate context modelling techniques to deal with these challenges. Another research direction aims to leverage task-related knowledge to help the emotion reasoning process, including sentiment-related knowledge, factual knowledge and linguistic knowledge. In this section, we introduce the development of ERC methods from the above two perspectives.

2.2.1 Context Modelling

There are mainly two directions for improving the context modelling ability: (a) Leverage superior neural network architectures to obtain better utterance-level representations, and (b) Design an appropriate dialogue modelling structure to facilitate cross-utterance emotion reasoning. We introduce these techniques in the following two sections.

Utterance Modelling

Early works in ERC utilise CNN to obtain utterance-level representations [110], [111]. For example, Majumder et al. [112] use a textual-CNN to encode the text modality of each utterance and a 3D-CNN for visual and acoustic modalities. For text-based ERC, some works leverage the strong text modelling ability of RNNs for representation learning of each utterance [76], [113]. For example, Hazarika et al. [76] pre-train a standard hierarchical recurrent encoder-decoder framework on large-scale conversation data generatively and transfer the pre-trained weights of the RNN-based encoder to ERC task.

As the Transformer architecture is proved effective in NLP, some works utilise a self-attention-based structure to model the utterances. Zhong et al. [114] and Zhang et al. [115] design multi-head self-attention-based networks to encode the word embeddings and incorporate factual knowledge at the word level. Some recent works leverage the strong Transformer-based PLMs to encode each utterance and obtain an informative representation separately. Commonly leveraged PLMs include BERT [113], [116], [117], RoBERTa [118], [119], XL-Net [44], [120] and BART encoder [84]. While most of these methods jointly fine-tune the PLMs during training, some works fuse the PLM weights to enable a faster optimisation process and still achieve good performance [121].

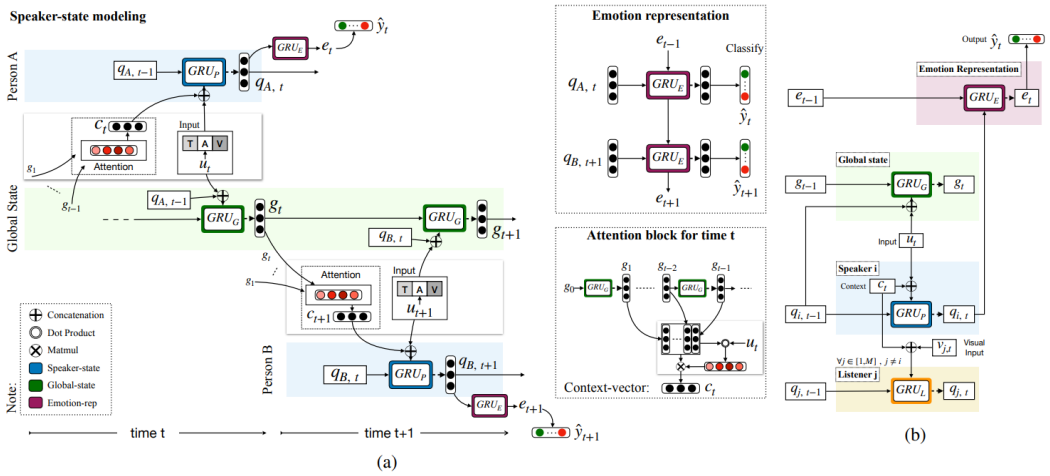


Figure 2.15. An overview of the DialogueRNN architecture. The figure is adapted from Majumder et al. [112]

Dialogue Modelling

In dialogue modelling, a natural approach is to divide the relations into intra- and inter-speaker dependencies and separately model them. A significant line of work usually utilises RNNs to model these relations [103], [110], [111]. For example, as shown in Figure 2.15, Majumder et al. [112] model the intra-speaker dependencies by devising a GRU to encode through the utterance sequence for each of the dialogue participants. Considering the inter-speaker dependencies, a global state RNN is proposed to encode the whole dialogue, which aims to model multi-party relations and emotional dynamics. The representations at previous time steps are stored as a memory bank. The memory bank is accessed via an attention mechanism and fused for emotion recognition at the current time step.

Self-attention-based models are also utilised for dialogue-level modelling. Zhong et al. [114] concatenate the utterance-level representations of previous utterances as contexts and design a hierarchical self-attention module to capture the context information. Then another multi-head dot-product attention is used to allow context reasoning, where the representation of the current utterance is used as the query, and the context representations are used as keys and values. Zhang et al. [115] achieve similar goals by introducing an incremental Transformer structure, which combines the utterance encoder and the cross-attention between utterance representations in different layers. Another branch of work leverages the strong context modelling ability of the Transformer-based PLMs to model the dialogue as a whole. For example, Li et al. [122] concatenate all utterances within the dialogue in sequential order and insert the "[CLS]" token at the beginning of each utterance. The dialogue is then fed into BERT as a single sequence. Kim et al. [119] follow a similar approach but combine speaker information by explicitly pre-pending the speaker name for each utterance. Shen et al. [120] exploit the XLNet [51] to model the dialogue sequentially, which solves the problem of the limited sequence length for previous methods. They further enhance XLNet by introducing four types of masks for the self-attention mechanism to focus on different aspects of dialogue modelling: (a) Global self-attention: global self-attention performs attention on all the dialogue contexts, which is the same as the vanilla self-attention; (b) Local self-attention: local self-attention only keeps a reception field of ω most recent historical utterances, where ω is a pre-defined context window and masks all representations before the field. It is motivated by the fact that the most recent dialogue histories influence the emotion of the current utterance more; (c) Speaker self-attention: speaker self-attention masks all utterance representations uttered by the listeners of the current speaker. It is designed to focus on intra-speaker dependency; (d) Listener self-attention: listener self-attention masks all previous utterance representations uttered by the current speaker and focuses on the listeners' utterances. It is designed to focus on inter-speaker dependency. These four attention results are concatenated for emotion reasoning.

Though rich information is available for ERC in the dialogue context, the extraction and reasoning process remains challenging. To introduce more priors and interpretable models, there are also many works [105], [117], [121], [123] that regard each utterance as a node and manually construct graphs for the dialogue. State-of-the-art GNNs are devised to learn node-level representations, and ERC is modelled as a node-classification task. For example, Shen et al. [121] design a Directed Acyclic Graph (DAG) on the dialogue. DAG denotes the set of graphs that have directed edges and no directed cycles. The DAG is built considering three principles: (a) A past utterance can pass the message to a future utterance, but the reverse is prohibited. Therefore, the edges can only point from past utterances to future utterances; (b) Utterances before the last utterance spoken by the current speaker are considered remote information. The influence of remote information is limited to the current utterance. Therefore, no edges are constructed from remote information to the current utterance; (c) Utterances between the last and current utterances spoken by the current speaker are considered local information. Local information is expected to have more influence on the current utterance, and each local utterance should connect to the current utterance. Based on these

principles, an example of DAG for ERC is given in Figure 2.16.

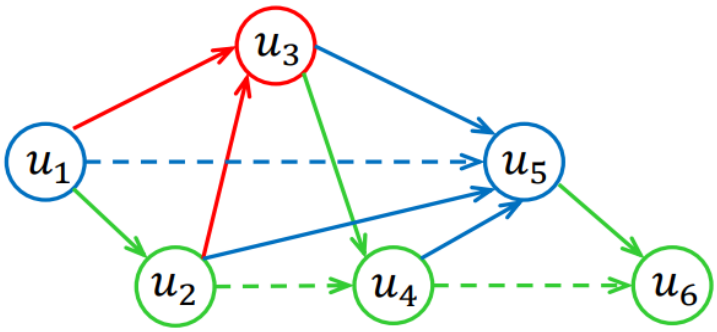


Figure 2.16. An example of the directed acyclic graph built for ERC.

In the example, each colour denotes the utterances of a dialogue participant, solid lines denote the edges from all local information to the current utterance, and dashed lines denote the connection from the last utterance of the current speaker to the current utterance. The remote information is anticipated to pass through the dashed lines to the current utterance. During the aggregation process, a multi-layer Directed Acyclic Graph Neural Network (DAGNN) is utilised for encoding the graph. DAGNN works like a combination of GNNs and RNNs. They aggregate information for each node in temporal order and allow all nodes to gather information from neighbours.

2.2.2 Knowledge Infusion

Constrained by the size of available datasets, ERC models cannot learn all the information required through the training process. Therefore, many works use external knowledge sources and infuse task-related knowledge to aid emotion reasoning. Common methods include direct knowledge infusion via the combination of knowledge representations, knowledge infusion via transfer learning and knowledge infusion via fine-tuning. Effective knowledge sources include factual knowledge, mental state knowledge, topic information and sentiment lexicons.

Knowledge Representation

The knowledge representations are incorporated into the model in different granularity for explicit knowledge infusion methods. Some works infuse knowledge from commonsense knowledge graphs to word-level representations [44], [115], [124]. The commonsense knowledge is mainly utilised to enrich the semantic space of the representations. For example, Xie et al. [44] introduce knowledge from ConceptNet [45]. Specifically, for each token in an utterance that is a concept, a sub-graph is extracted with each of its direct neighbours, and a GAT is utilised to aggregate the sub-graph, as introduced in Sec. 2.1.3. For each token representation h, we obtain a corresponding knowledge representation k. Tokens that are not in the knowledge graph are also assigned an average of all node representations in the graph.

The knowledge enriched embedding u is obtained via concatenation:

$$u = [h; k] \tag{2.26}$$

where $[;]$ denotes concatenation. The work then introduces another module to enable a full interaction between the token and knowledge embeddings, called self-matching. For two token representations within one utterance, u_i and u_j, their similarity is computed via a trilinear function:

$$r_j^i = \mathbf{W}^\top [u_i; u_j; u_i \odot u_j] \tag{2.27}$$

where W is a learnable parameter matrix, and $\odot$ denotes the element-wise multiplication operation. A self-attention matrix $\mathbf{Q}$ is obtained with the softmax operation, where q_j^i is its ij-th entry:

$$q_j^i = \frac{exp(r_j^i)}{\sum_{k=1}^{N} exp(r_k^i)} \tag{2.28}$$

where N denotes the token number within the utterance. In addition, indirect interaction allows the model to learn deeper semantic relations within the knowledge-enriched representations. To achieve the indirect interaction, a self-multiplication of the attention matrix Q is calculated:

$$\hat{\mathbf{Q}} = \mathbf{Q}\mathbf{Q}^\top \tag{2.29}$$

With $\hat{\mathbf{Q}}$, each token pair can interact via another token. Two attended vectors are computed with the matrices:

$$\begin{aligned} v_i &= \sum_{k=1}^{N} q_k^i u_k \\ \hat{v}_i &= \sum_{k=1}^{N} \hat{q}_k^i u_k \end{aligned} \tag{2.30}$$

The two attended vectors are concatenated in various means to allow rich interactions:

$$c_i = [u_i; v_i; u_i - v_i; u_i \odot v_i; \hat{v}_i; u_i - \hat{v}_i] \tag{2.31}$$

where c_i denotes the final knowledge enriched representation for token i. The self-matching process introduces knowledge purposefully instead of acting as noise. Self-matching is one of the effective knowledge interaction methods proposed by many ERC works, and experiments show their effectiveness.

In ERC, Commonsense knowledge is also introduced on the utterance level. A representative work is COSMIC [103], which utilises the generative knowledge source COMET [54] trained on the knowledge graph ATOMIC (introduced in Sec. 2.1.3). The utterance u is concatenated with each relation type as input of COMET. The representations of the COMET encoder for each relation type are regarded as the knowledge representation and directly concatenated with the utterance-level representations of the corresponding utterance. The

knowledge-enriched representations are combined via the attention mechanism and used for final emotion prediction. In another work, Li et al. [105] build a graph on the dialogue and utilise the COMET knowledge representations as the edge representations of the graph. A graph transformer is used to propagate the information of the dialogue graph.

Transfer Learning

Explicit knowledge infusion usually requires structured knowledge sources, which require much human processing. However, a large amount of helpful knowledge exists in unstructured natural language data and demands specific techniques to leverage. Therefore, some works devise the transfer learning method, which designs relevant pre-training tasks and transfers the pre-trained weights to ERC [76], [113]. For example, Hazarika et al. [76] design a conversation modelling pre-training task and transfer the weights of the dialogue encoder to ERC, as shown in Figure 2.17.

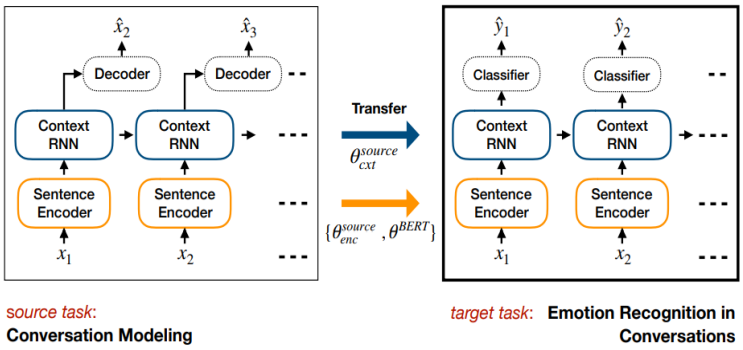


Figure 2.17. The transfer learning framework for ERC. The figure is adapted from Hazarika et al. [76].

Two large-scale conversation datasets are collected for the source conversation modelling task, and an RNN-based Hierarchical Recurrent Encoder-Decoder (HRED) architecture is trained on these data in the response generation style. The HRED contains a sentence encoder (usually an RNN or BERT structure), an RNN-based context encoder and an RNN-based utterance decoder. The sentence and context encoder weights are transferred to the ERC task. In ERC, the transferred HRED encoders are used to encode the dialogue history, and an FFN-based classifier is devised to predict emotions on top of the context-aware representations.

Fine-Tuning

Though transfer learning brings helpful information to ERC, it usually requires large-scale datasets and high computational costs for pre-training. Therefore, some works utilise PLMs and infuse useful information via fine-tuning, such as sentiment scores [44], topic information [116], [125] and speaker-utterance relations [122]. The knowledge-aware fine-tuning task is usually jointly trained with the ERC task in a multi-task learning manner. For example, Xie et al. [44] propose an auxiliary Sentiment Polarity Intensity Prediction (SPIP) task, which assigns a sentiment score to each phrase within the dataset that exists in the emotion lexicon SenticNet [95]. The model is trained to predict the sentiment score of each labelled

phrase. The SPIP task enables the model to be aware of the sentiment intensity of key phrases and helps determine the emotion of the utterance.

More fine-grained sentiment information is also proved effective in ERC. For example, Valance-Arousal-Dominance is an effective dimensional emotion representation model in psychology [126]. Therefore, VAD information is also incorporated to facilitate categorical emotion detection [124], [127], [128], which considerably boosts the model performance. In ERC, Zhong et al. [124] utilise the human-labelled VAD scores from the lexicon NRC-VAD [53] to help determine the weights of each term during the factual knowledge infusion process, which aims to attend more on emotion-intense concepts.

2.3 Mental Health Analysis

2.3.1 Foundations in Psychology

The high prevalence of mental health disorders worldwide has been one of the most severe public health concerns. Therefore, developing convenient early detection methods for mental health problems has attracted growing research interests. In psychology, early works have noticed the theoretical relations between mental health conditions and certain linguistic features such as "depressive language", and try to leverage these features to aid mental health analysis. For example, Beck et al. [24] develop cognitive therapy and consider the frequency of negatively-valenced words expressed during the therapy. The result shows that people with a higher frequency of negatively-valenced words tend to face higher risks of mental health issues. Pyszczynski et al. [25] focus on studying the expression of depression. They collect the different judgments of patients diagnosed as depressed and non-depressed people on the probability of future positive and negative life events occurring to themselves and others. The results show an apparent difference in positive levels that depressed people are much less optimistic in their anticipation of the future. The study also finds that depressed patients are usually more self-focused than non-depressed people and hypothesises that inducing depressed subjects to focus externally would attenuate the pessimistic tendencies.

Further empirical studies verify these hypotheses and further focus on validating the connections between certain linguistic features and the patients' mental states. Rude et al. [26] collect essays written by current depressed and non-depressed college students and examine features that reflect the cognitive operations associated with depression and depression-vulnerability. The study discovers several patterns. For example, depressed people use negatively-valenced words and the word "I" more frequently than non-depressed or previously depressed people, which corresponds to early hypotheses of Pyszczynski et al. [25]. Ramirez-Esparza et al. [27] collect multi-lingual posts (English and Spanish) from forums on the Internet to examine the above results in broader views and different cultural backgrounds. The results show that though depressed people with different cultural backgrounds tend to be concerned about different aspects of depression, linguistic cues associated with depression are higher in

depressed than in non-depressed posts in both English and Spanish. With the connection between mental health issues and language expression assured, some works utilise social media as a rich source of text data and use these online user-generated posts to manually analyse mental health conditions [129]–[131] and detect mental disorders such as depression, PTSD and eating disorders.

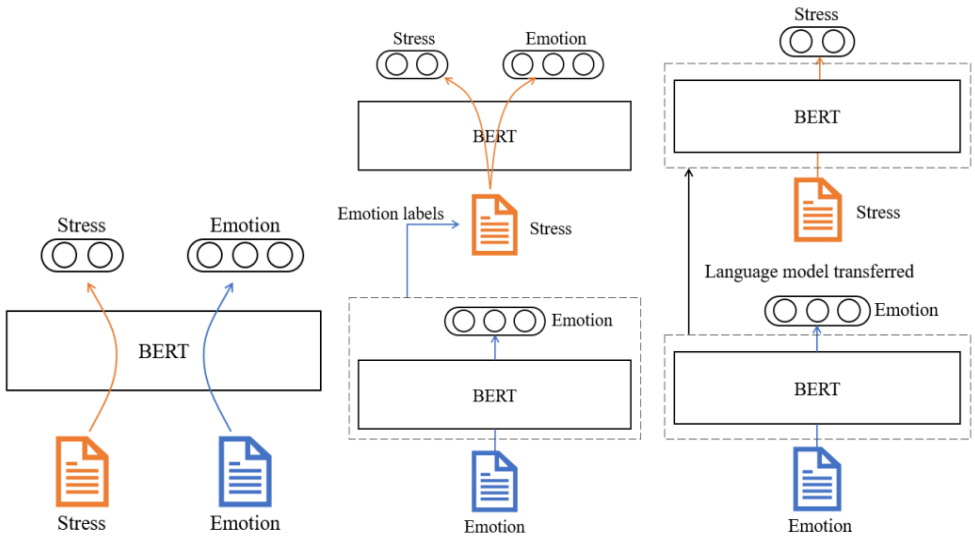
2.3.2 NLP-Based Approaches

With the fast-growing numbers of online texts and the sensitivity of mental health conditions, manual analysis of texts and timely psychiatric treatment on a large scale is no longer practical. Therefore, the artificial intelligence community pays attention to mental health analysis and tries to leverage NLP and text mining techniques for automated mental health analysis from social media data. However, given the non-experts nature of NLP researchers on psychology, these methods are not expected to make an actual diagnosis but offer assistance for early detection. This claim is often stated in the ethical considerations of previous related works.

Early methods extract statistical features such as Bag-of-Words [132], [133] and TF-IDF [134] from the collected data, then employ them in traditional machine learning methods such as Support Vector Machine [135], Random Forest [136] and Logistic Regression [137], [138] to predict depression, suicide tendency [131], etc. For example, Saleem et al. [135] collect data from an online forum for veterans with post-combat psychological issues and annotate the dataset with distress labels. The distress identification task is divided into a two-stage text classification problem. In the first stage, a support vector machine is utilised to classify relevant versus irrelevant messages, where each post is classified as whether it bears useful information. An ensemble of multiple machine learning methods is used for the second-stage distress label classification task. The work utilises several linguistic features, including bag-of-words representations, normalised count of punctuations and pronouns, average sentence lengths and sentiment-bearing word features, etc. The experiments show the effectiveness of the proposed features and the ensemble methods.

Advances in deep learning also boost mental health-related tasks. Most current methods employ deep learning models to capture latent semantic information automatically without complex feature engineering. Some works utilise CNN [139] or RNN, including LSTM [140] and GRU [141] to detect depression based on the posted text. For example, Ghosh et al. [140] collect data from Twitter and extract features from each post, such as the 12-dimensional emotional features, topic model features, and user information. Then a 3-layer LSTM is used to encode these extracted features and predict the depression intensity of the posts. Researchers also explore hybrid architectures of CNN and RNN to capture both local and long-dependency features [142], [143]. Furthermore, an attention mechanism [141], [144], [145] is used to make models focus on the most significant parts of the input. In addition, multi-task learning is utilised to jointly train with other auxiliary tasks such as statistical feature classification [146], and depression cause prediction [147], which provide additional information for depression detection. With the development of Transformer-based models,

PLMs such as BERT [148], [149], RoBERTa [150] and GPT [151] are also widely applied to detection of many mental health issues such as suicide and PTSD, due to their strong context modelling ability. The representations of the PLMs are usually fine-tuned on the target dataset to adapt to the mental health domain. The representations are sometimes used as input features to classification algorithms such as logistic regression and outperform the statistical feature-based methods.



(a) The Multi^{Alt} model. (b) The Multi model. (c) The Fine-Tune model.

Figure 2.18. The emotion information-enriched models for the stress detection task. The figure is adapted from Turcan et al. [152].

In short-context scenarios, identifying mental health issues can be difficult due to the lack of information. Therefore, another branch of work infuses external knowledge to aid mental health-related tasks. Some works notice the close relations between mental states and emotion expressions, and leverage history context modelling [153], [154], multi-task learning [155] or transfer learning [152], [156] to infuse emotion information. As an example, Turcan et al. [152] develop three emotion information-enriched model architectures for the stress detection task, as shown in Figure 2.18. The designed architectures are: (a) a multi-task learning setting where the emotion detection and stress detection tasks share the BERT parameters; (b) the emotion detection task is performed first, and the detected emotion features are combined with stress detection features; (c) the model is pre-trained on the emotion detection task first, and the pre-trained weights are transferred to the stress detection task. Experiments show that all three settings perform well. In addition, Ji et al.[157] collect a large amount of data on mental health from social media platforms and fine-tune them on the pre-trained BERT for a new domain-specific model, namely MentalBERT. The knowledge introduced during post-training boosts MentalBERT to achieve state-of-the-art performance on several stress and depression detection datasets. In addition, some works focus on qualitative analysis of the bias of these models towards gender and racial/ethnic groups [158].

2.4 Summary

This chapter provided an overview of the related literature with our methodology. Specifically, we first introduce state-of-the-art representation learning methods for NLP, starting with neural networks such as CNN, RNN, and Transformer. We also present the widely utilised PLM-based methods. Then we introduce the contrastive learning methods and their applications in representation learning. Knowledge-enhanced representation learning methods are also briefly explained. For ERC, we first introduce popular architectures for context modellings, such as graph-based and recurrent-based structures. Then we present current knowledge sources and infusion methods for ERC, such as knowledge graphs and pre-training-based knowledge infusion. For mental health analysis, we also briefly introduce previous works in the NLP research community.

Chapter 3

Cluster-Level Contrastive Learning

3.1 Overview

Context modelling is a crucial challenge for ERC. The emotion of each utterance is influenced both by the previous utterances of the speaker and the responses of other participants [120]. Current methods mainly utilise the Transformer-based PLMs [73] to deal with this challenge. However, PLMs are found to poorly capture the semantics of sentences without careful fine-tuning [159], which also raises difficulties for the identification of semantically similar emotions (e.g., *excited* and *happy*). Since previous works utilise unsupervised contrastive learning to alleviate this problem [82], [159] and obtain promising results in several text classification tasks, Li et al. [84] manage to introduce supervised contrastive learning, where utterances with the same emotion label are considered as positive pairs, and the instance-level utterance representations are directly utilised for contrastive learning. SCL decouples the overlap between samples with similar emotions in the semantic representation space and facilitates learning the decision boundary.

However, SCL treats two samples as a negative pair as long as they are with different labels, regardless of the quantitative semantic similarity between emotions (e.g., *happy* is closer to *excited* than *sad*). This negligence is manifested by the fact that all negative samples are equally pushed apart in the semantic space during SCL. In addition, the success of works with manual feature selection [160] (e.g., pleasantness and emotion intensity of the current utterance) shows that ERC is not equally dependent on all features embedded in the high-dimensional utterance representations. We expect a low-dimensional prototype for each emotion, defined as *a representative embedding for a group of similar instances* [78], to be more efficient in contrastive learning. High-dimensional SCL space also leads to other limitations: (a) Euclidean distance becomes less meaningful due to the curse of dimensionality [80]; (b) the results are hard to interpret and visualise. Previous works mainly utilise t-SNE [161] to reduce the dimensions, but it may lead to mis-interpretations[1], and the reduced dimensions have no practical significance; (c) stable SCL requires large batch sizes [40], which leads to high computational costs. This requirement limits the application of SCL-based methods in low computational resource scenarios.

To tackle the above challenges, we propose a novel low-dimensional Supervised Cluster-

[1] http://deeplearning.csail.mit.edu/slide_cvpr2018/laurens_cvpr18tutorial.pdf

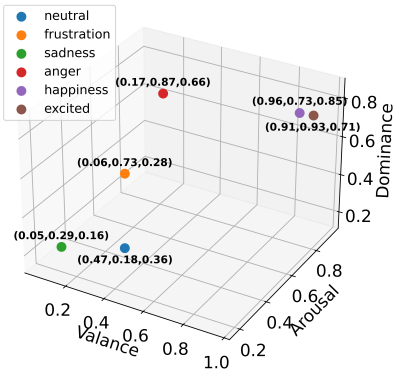


Figure 3.1. An example of appropriate emotion prototypes in VAD space bringing quantitative information.

level Contrastive Learning (SCCL) method for ERC. With a PLM-based context-aware utterance encoder, we improve SCL as follows: Firstly, we reduce the high-dimensional contrastive learning space to a three-dimensional space called Valence-Arousal-Dominance, a widely explored affect representation model in psychology [3], [4]. Secondly, we introduce a prototype for each emotion in VAD space from a sentiment lexicon called NRC-VAD [53], which brings quantitative information between all emotion labels. We provide an example for some emotions in Figure 3.1, where the emotions within the same sentiment polarity lie closer, and their relative positions are reasonable. Regarding each emotion category as a cluster centre, SCCL predicts the cluster-level VAD for each emotion, transfers the instance-level emotion labels to cluster level with the emotion prototypes, and performs cluster-level contrastive learning. Meanwhile, Liu et al. [162] argue that current PLMs lack fine-grained linguistic knowledge, which is proved useful to help to model the utterances in sentiment-related tasks [107]. Factual knowledge is also widely leveraged in ERC [44], [103], [114] and proved effective in enriching the context and providing relevant knowledge for emotion reasoning. Therefore, we infuse linguistic and factual knowledge leveraging the pre-trained knowledge adapter in a plug-in manner, which avoids modification of the PLM weights. An overview of our model is presented in Figure 3.2, where in ERC and SCCL, each colour denotes an emotion category. The function of each part is described: (a) linguistic and factual knowledge is infused into the knowledge adapters through pre-training and combined with the utterance encoder in a plug-in manner; (b) the one-hot label matrix is mapped to the VAD space with emotion prototypes; (c) cluster-level representations are aggregated from the VAD predictions, and contrastive learning is performed in the VAD space. We conduct experiments on four widely used ERC benchmark datasets. The results show that our method achieves competitive performance on all datasets and state-of-the-art outcomes on three: IEMOCAP, MELD, and DailyDialog. An ablation study proves the effectiveness of each proposed module, and further comparisons analyse their property and visualise the contrastive results.

To summarise, this work mainly makes the following contributions:

- We reduce the high-dimensional SCL space to a three-dimensional space VAD, which improves model performance and facilitates the interpretability of contrastive learning.
- For the first time in ERC, we incorporate VAD prototypes into SCL by proposing a novel supervised cluster-level contrastive learning method. Analysis shows that SCCL

remains stable with both large and small batch sizes, which facilitates its application in low computational resource scenarios.

- We infuse linguistic and factual knowledge into the utterance encoder by utilising the pre-trained knowledge adapters and analysing their benefits via the ablation study and empirical comparisons.

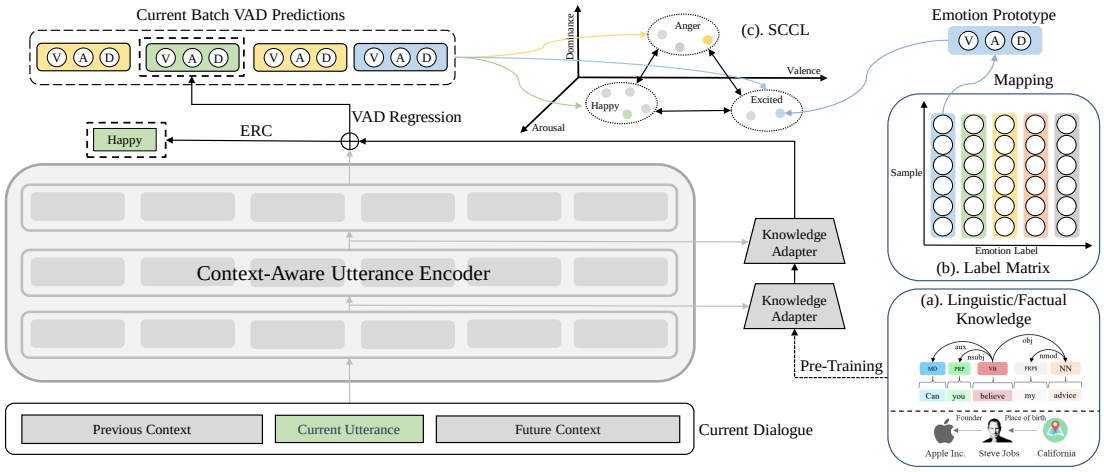


Figure 3.2. An overview of our model architecture.

3.2 Pre-trained Knowledge Adapter

3.2.1 Context-Aware Utterance Encoder

To introduce speaker information, we pre-pend the speaker's name $P(D_j)$ for each utterance D_j as $\hat{D}_j$. Then the current utterance $\hat{D}_i$ is concatenated with both past and future contexts to get the context-aware input R_i: $R_i = \{[CLS]; \hat{D}_{i-W_p}; ...; \hat{D}_i; ...; \hat{D}_{i+W_f}; [EOS]\}$, where W_p and W_f denotes past and future context window size, $[CLS]$ and $[EOS]$ denote the start-of-sentence and end-of-sentence token in PLMs. Then we use R_i to obtain the context-aware utterance embeddings:

$$H_i^L = Encoder(R_i) \tag{3.1}$$

where $Encoder$ denotes the RoBERTa [50] encoder, $H_i^L \in \mathbb{R}^{S \times D_h}$ denotes the final output of the L-th layer, S denotes sequence length and D_h is the hidden size of the encoder. H_i^L is used as the context-aware representation for the i-th utterance in the next methods.

3.2.2 Knowledge-infusion with Adapter

We incorporate external knowledge into the utterance encoder by injecting pre-trained knowledge adapters. The knowledge adapter is a multi-layer Transformer-based model separately initialised and pre-trained for each knowledge source. During pre-training, the weights of the PLM are frozen, and only the knowledge adapter weights are updated. Compared with normal pre-training or explicit incorporation methods of knowledge infusion, this training paradigm

has three advantages: (a) The weight fusion prevents the catastrophic forgetting [163] problem of PLMs when multiple knowledge sources are infused; (b) The training process saves memory and speeds up since the knowledge adapter is smaller in size than the PLM; (c) With a new knowledge source to incorporate, the weights of the PLM do not need retraining.

As shown in Figure 3.2(a), we follow the methodology of Wang et al. [52] and pre-train two knowledge adapters with commonsense knowledge from T-REx [164] (FacAdapter) and linguistic knowledge provided by Stanford Parser[2] (LinAdapter). T-REx is a large-scale factual knowledge graph built from over 11.1M alignments between statements and triples of Wikipedia, which provide relevant knowledge to enrich the context and aid emotion reasoning. For example, the statement "Vincent van Gogh and other late 19th century painters used blue not just to depict nature, but to create bad moods and emotions" is aligned with triples <Vincent van Gogh, occupation, painters> and <blue, represent, bad moods and emotions>. Given the statements and entities as input during the pre-training process, the FacAdapter predicts the relation type of the aligned triples. Linguistic knowledge is naturally embedded in language texts, which benefits sentence modelling. It can be obtained by running a dependency parser to get semantic and syntactic information. Therefore, for pre-training on linguistic knowledge, the LinAdapter takes the texts as input and predicts the syntactic and semantic relations annotated by the parser.

The adapter is utilised on the utterance encoder in a plug-in manner as follows: let $Encoder^l$ denote the l-th hidden layer of the utterance encoder. LinAdapter, denoted as *Adapter*, has n_k Transformer-based layers, where $n_k \leq L$ and $Adapter^j$ denotes j-th layer of the adapter. LinAdapter is pre-defined as an interactive layer set $\hat{L} = \{l_1, l_2, ..., l_{n_k}\}$, where the hidden states of $Encoder^{l_j}$ will be combined in $Adapter^j$. Specifically, for i-th utterance and each $l_j \in \hat{L}$, this process can be formalised as:

$$H_f^j = H_i^{l_j} \oplus H_a^{j-1} \tag{3.2}$$

$$H_a^j = Adapter^j(H_f^j) \tag{3.3}$$

where $H_a^j \in \mathbb{R}^{D_h}$ denotes the j-th layer output of the knowledge adapter, $H_i^{l_j}$ is the l_j-th layer output of the utterance encoder, $\oplus$ denotes element-wise addition, and H_a^1 is initialised with an all-zero matrix. The final layer output $H_a^{n_k}$ of the adapter is combined with the PLM embeddings as the final utterance representations:

$$\hat{H}_i = Tanh((H_i^L \oplus H_a^{n_k})W_1 + b_1) \tag{3.4}$$

where $\hat{H}_i \in \mathbb{R}^{S \times D_h}$ denotes the knowledge-enhanced utterance embeddings, $Tanh$ denotes the tanh activation function, and $W_1 \in \mathbb{R}^{D_h \times D_h}$, $b_1 \in \mathbb{R}^{D_h}$ are learnable parameters.

[2]https://nlp.stanford.edu/software/lex-parser.html

3.3 Supervised Cluster-Level Contrastive Learning

3.3.1 Emotion Prototypes

Valence-Arousal-Dominance (VAD) maps emotion states to a three-dimensional continuous space, where Valence reflects the pleasantness of a stimulus, arousal reflects the intensity of emotion provoked by a stimulus, and dominance reflects the degree of control exerted by a stimulus [5]. Instead of directly leveraging the one-hot categorical emotion labels for supervision, VAD allows each categorical emotion to be projected into the space with measurable distances. A few ERC resources [14] are human-labelled with a context-dependent VAD score for each utterance j: $\textit{H-VAD}_j \in \mathbb{R}^3$, which can be leveraged for accurately computing emotion prototypes.

However, utterance-level VAD labels are expensive and unavailable in most cases. For application in such scenarios, we consider the context-independent word-level VAD information from sentiment lexicons. We utilise NRC-VAD [53], a VAD sentiment lexicon that contains reliable human-ratings of VAD for 20,000 English words. All the terms in NRC-VAD denote or connote emotions, and are selected from commonly used sentiment lexicons and tweets. Each of these terms is first strictly annotated via best-worst scaling with crowdsourcing annotators. Then an aggregation process calculates the VAD for each term ranging from 0 to 1. With the pre-defined categorical emotion set E, we extract the VAD for each of the emotion $e \in E$ from NRC-VAD: $\textit{NRC-VAD}_e \in \mathbb{R}^3$. For example, the emotion *happiness* is assigned: $[0.9600, 0.7320, 0.8500]$. The VAD information from either of the above methods is utilised to obtain cluster-level emotion representations. We expect utterance-level H-VADs to outperform word-level NRC-VADs since they are context-dependent and bear more fine-grained VAD information.

3.3.2 Cluster-Level Contrastive Learning

Though VAD prototypes provide useful quantitative information, they are difficult to be infused to enhance SCL since infusion during inference leads to the leakage of label information. Therefore, we propose to perform SCL at cluster level instead of instance level with a novel SCCL method. Regarding each emotion category as a cluster centre, we perform SCCL with cluster-level representations separately obtained from emotion labels and model predictions, where both processes are introduced below.

We first compute for emotion labels using the emotion prototypes. For a batch of utterances, as shown in Figure 3.2(b), the emotion labels are projected to a one-hot label matrix $M \in \mathbb{R}^{|B| \times |E|}$, where $M_i \in \mathbb{R}^{|E|}$ is the i-th row of M, denoting the one-hot emotion label of the i-th sample, and $M^j \in \mathbb{R}^{|B|}$ is the j-th column of M, denoting the samples with the label

$e_j \in E$. For the j-th cluster e_j, we map M^j to the VAD space as follows:

$$\hat{M}^j = \frac{\sum_{k=1}^{B} M^{jk} \times \textit{VAD}_{e_j}}{\sum_{k=1}^{B} M^{jk}} \tag{3.5}$$

where M^{jk} denotes k-th element of M^j, $\hat{M}^j \in \mathbb{R}^3$ denotes the cluster-level representation of e_j. When utterance-level VAD information is available, $\textit{VAD}_{e_j} = \textit{H-VAD}_j$. When NRC-VAD information is utilised, $\textit{VAD}_{e_j} = \textit{NRC-VAD}_{e_j}$ and $\textit{NRC-VAD}_{e_j}$ is directly regarded as the cluster-level emotion representation for e_j.

Then we compute for the model predictions. One choice is to adopt a similar approach as the emotion labels, which utilises the normalised categorical predictions with Softmax, and maps them to the VAD space using Eqn.3.5. However, it may deteriorate SCCL to the vanilla case where the model only learns the one-hot label information and ignores the emotion prototypes. Therefore, we utilise a neural network to parameterise the dimension reduction process from the semantic space to the VAD space. Specifically, for $\hat{H}_i$, we regard the embedding of the start-of-sentence token at position 0 $\hat{H}_i^{[CLS]}$ as its utterance-level embedding, and map $\hat{H}_i^{[CLS]}$ to the VAD space:

$$H_i^{VAD} = \frac{1}{1 + e^{-(\hat{H}_i^{[CLS]} W_2 + b_2)}} \tag{3.6}$$

where $\hat{H}_i^{[CLS]} \in \mathbb{R}^{D_h}$, $H_i^{VAD} \in \mathbb{R}^3$, and $W_2 \in \mathbb{R}^{D_h \times 3}$, $b_2 \in \mathbb{R}^3$ are learnable parameters.

As shown in Figure 3.2(c), following the idea of *labels as representations*, for each batch, we calculate the SCCL loss as follows:

$$\hat{H}_j^{VAD} = \frac{1}{|\mathcal{A}(j)|} \sum_{i \in \mathcal{A}(j)} H_i^{VAD} \tag{3.7}$$

$$sim(j) = log \frac{exp(\hat{H}_j^{VAD} \cdot \hat{M}^j)/\tau}{\sum_{e_k \in E} exp(\hat{H}_j^{VAD} \cdot \hat{M}^k)/\tau} \tag{3.8}$$

$$\mathcal{L}_{SCCL} = -\frac{1}{|E|} \sum_{e_j \in E} sim(j) \tag{3.9}$$

where $\hat{H}_j^{VAD} \in \mathbb{R}^3$ denotes the cluster-level embedding for e_j from model predictions, $\mathcal{A}(j) = \{i | Y_i = e_j, i \in [1, |B|]\}$ records the samples $D_i \in B$ labelled with the emotion e_j, $\cdot$ denotes dot-product operation, $\tau \in \mathbb{R}^+$ is the temperature coefficient, and $\mathcal{L}_{SCCL}$ denotes the SCCL loss.

3.4 Model Training

We combine SCCL with ERC in a multi-task learning manner. For the i-th utterance, we still utilise $\hat{H}_i^{[CLS]}$ as the utterance-level embedding, and compute the final classification

probability as follows:

$$\hat{Y}_i = Softmax(\hat{H}_i^{[CLS]} W_3 + b_3) \tag{3.10}$$

where $\hat{Y}_i \in \mathbb{R}^{|E|}$, and $W_3 \in \mathbb{R}^{D_h \times |E|}$, $b_3 \in \mathbb{R}^{|E|}$ are learnable parameters. Then we compute the ERC loss using the standard cross-entropy loss:

$$\mathcal{L}_{ERC} = -\frac{1}{B} \sum_{i=1}^{B} \sum_{j=1}^{|E|} Y_i^j log \hat{Y}_i^j \tag{3.11}$$

where Y_i^j and $\hat{Y}_i^j$ are the j-th element of Y_i and $\hat{Y}_i$. Finally, we combine the ERC loss and SCCL loss in the following manner:

$$\mathcal{L} = \mathcal{L}_{ERC} + \alpha \mathcal{L}_{SCCL} \tag{3.12}$$

where $\alpha \in [0, 1]$ denotes the pre-defined weight coefficient of $\mathcal{L}_{SCCL}$.

3.5 Experimental Settings

3.5.1 Datasets

In our work, we follow the setting of all previous ERC works and assume that each utterance has a single categorical emotion label, due to the limitation of most ERC datasets. We evaluate our method on the following four benchmark datasets. The statistics of all datasets are presented in Table 3.1.

Dataset	Conv.(Train/Val/Test)	Utter.(Train/Val/Test)	Utter./Conv
IEMOCAP	100/20/31	4,778/980/1,622	49.2
MELD	1,038/114/280	9,989/1,109/2,610	9.6
EmoryNLP	713/99/85	9,934/1,344/1,328	14.1
DailyDialog	11,118/1,000/1,000	87,170/8,069/7,740	7.9

Table 3.1. Statistics of the datasets. Conv. and Utter. denotes the conversation and utterance number. Utter./Conv denotes the average utterance number per dialogue.

IEMOCAP [14]: A two-party multi-modal conversation dataset derived from the scenarios in the scripts of the two actors. For all datatsets, we only utilise the text modality in our experiments. The pre-defined categorical emotions are *neutral, sad, anger, happy, frustrated, excited*.

MELD [15]: A multi-party multi-modal dataset enriched from *EmotionLines* dataset, collected from the scripts of American TV show *Friends*. The pre-defined emotions are *neutral, sad, anger, disgust, fear, happy, surprise*.

EmoryNLP [16]: Another dataset collected from TV show *Friends*, but annotated with different emotion label categories. The pre-defined emotions are *neutral, sad, mad, scared, powerful, peaceful, joyful*.

DailyDialog [17]: A dataset compiled from human-written daily conversations with only two parties involved and no speaker information. The pre-defined emotion labels are the Ekman's emotion types: *neutral, happy, surprise, sad, anger, disgust, fear*.

IEMOCAP	neutral	frustrated	sad	anger	excited	happy	–
Valence	0.469	0.060	0.052	0.167	0.908	0.960	–
Arousal	0.184	0.730	0.288	0.865	0.931	0.732	–
Dominance	0.357	0.280	0.164	0.657	0.709	0.850	–
MELD	neutral	joy	surprise	anger	sad	disgust	fear
Valence	0.469	0.980	0.875	0.167	0.052	0.052	0.073
Arousal	0.184	0.824	0.875	0.865	0.288	0.775	0.840
Dominance	0.357	0.794	0.562	0.657	0.164	0.317	0.293
EmoryNLP	joyful	neutral	powerful	mad	sad	scared	peaceful
Valence	0.990	0.469	0.865	0.219	0.225	0.146	0.867
Arousal	0.740	0.184	0.830	0.873	0.333	0.828	0.108
Dominance	0.667	0.357	0.991	0.277	0.149	0.185	0.569
DailyDialog	neutral	anger	disgust	fear	happy	sad	surprise
Valence	0.469	0.167	0.052	0.073	0.960	0.052	0.875
Arousal	0.184	0.865	0.775	0.840	0.732	0.288	0.875
Dominance	0.357	0.657	0.317	0.293	0.850	0.164	0.562

Table 3.2. The NRC-VAD assignments to all emotions in the four datasets.

Among the above datasets, human-labelled utterance-level VAD scores are only available in IEMOCAP, where the aggregation process calculates the VAD for each utterance ranging from 1 to 5. To cope with the SCCL method, we linearly transform all VAD scores to the range $[0, 1]$ during inference.

When NRC-VAD is utilised, the emotion prototypes of the labels for all datasets are listed in Table 3.2. According to the assignments, most of the cluster centres (VAD assignments) reflect appropriate positions of the corresponding emotions in VAD space, where similar emotions are measurably closer to each other while maintaining a fine-grained difference to facilitate the model to distinguish them. For example, *happy* stays closer to *excited* than *anger* in IEMOCAP. In addition, for all four datasets, positive and negative emotions are mostly separated by *neutral* in the dimension Valence, while the emotions within each sentiment polarity mostly differs in Arousal and Dominance.

3.5.2 Baselines

We select the following strong baseline models to compare with our model:

BERT-Large [49]: The model initialises from pre-trained weights of BERT-Large and is fine-tuned on the ERC task. The $[CLS]$ embedding at position 0 of the BERT output is passed through an FFN to predict the emotion.

DialogXL [120]: This work is based on the PLM XLNet [51]. It proposes four types of dialogue-aware self-attention (global self-attention, local self-attention, speaker self-attention,

listener self-attention) to model inter- and intra-speaker dependencies and uses an utterance recurrence mechanism to model the long-range contexts.

RGAT [117]: The model constructs a graph on each dialogue to introduce prior knowledge in context modelling and combines a relation position encoding to introduce sequential information in the graph. GNNs are used to summarise and aggregate the graphs. The manually designed graph structure guides the message passing process.

COSMIC [103]: This work uses the RNN structures to model the dialogue history for each participant and the context information. It also extracts utterance-level commonsense knowledge to model several aspects of the speakers' mental states and attentively infuses the knowledge into the utterance representations.

KI-Net [44]: This work leverages token-level commonsense knowledge from knowledge graphs and explicitly infuses the knowledge into token-level dialogue representations. It also implicitly introduces sentiment scores from the sentiment lexicon SenticNet [95] via multi-task learning to guide emotion reasoning.

DAG-ERC [121]: Utilising RoBERTa-Large as the single utterance encoder, this model builds a directed acyclic graph on the dialogue and uses a multi-layer DAGNN to aggregate the information on the graph. The outputs of all layers are concatenated for ERC classification.

SGED [165]: This method proposes a speaker-guided encoder-decoder framework to exploit speaker information for ERC.

SKAIG [105]: This work extracts psychological commonsense knowledge from COMET [54], builds a graph on the dialogue according to different aspects of the knowledge, and uses the corresponding knowledge representations as the edge representations.

CoG-BART [84]: Based on BART-Large [166], this work utilises SCL and a response generation auxiliary task to distinguish semantics of utterances with similar emotions. The tasks are trained in a multi-task learning manner.

3.5.3 Implementation Details

We conduct the experiments on IEMOCAP, MELD, and EmoryNLP using a single Nvidia Tesla V100 GPU with 16GB of memory, and set the batch size to 4. For large-scale dataset DailyDialog, we conduct the experiments using a single Nvidia Tesla A100 GPU with 80GB of memory, and set the batch size to 16. We initialise the pre-trained weights of PLMs and use the tokenization tools both provided by Huggingface[3]. The pre-trained knowledge adapter weights are from Wang et al.[52], and these weights are fused during training. We leverage AdamW optimiser [167] to train the model, with a linear warm-up learning rate scheduling [168] of warm-up ratio 20% and peak learning rate 1e-5. Due to the limitation of computation memory, we use mixed floating point precision [169] during training. Hyper-parameters are tuned on the validation set. α is tuned on the interval $[0.5, 1.0]$ and set to 1.0

[3] https://huggingface.co/

for MELD and 0.8 for all other datasets. $S = 512$, $D_h = 1024$, $L = 24$ for RoBERTa-Large, and $D_h = 768$, $L = 12$ for RoBERTa-Base. We set a dropout rate 0.1, a L^2 regularisation rate 0.01 to avoid over-fitting. We use the weighted-F1 measure as the evaluation metric for IEMOCAP, MELD and EmoryNLP. Since *neutral* is the dominant tag in DailyDialog, we use micro-F1 for this dataset, and ignore the label *neutral* when calculating the results as in the previous works [84], [121]. All reported results are averages of five random runs.

3.6 Results and Analysis

3.6.1 Overall Performance

Table 4.3 presents the performance of our method, and compares it to the strong baseline models.

Model	IEMOCAP	MELD	EmoryNLP	DailyDialog
BERT-Large [49]	60.60	62.83	33.73	54.09
DialogXL [120]	65.94	62.41	34.73	54.93
COSMIC [103]	65.28	65.21	38.11	58.48
KI-Net [44]	66.98	63.24	–	57.30
SGED [165]	68.53	65.46	**40.24**	–
SKAIG [105]	66.96	65.18	38.88	59.75
RGAT [117]	65.22	60.91	34.42	54.31
DAG-ERC [121]	68.03	63.65	39.02	59.33
CoG-BART [84]	66.18	64.81	39.04	56.29
HVAD-SCCL	**69.88***($\pm$0.50)	–	–	–
NRC-SCCL	69.81*($\pm$0.63)	**65.70***($\pm$0.82)	38.75($\pm$0.49)	**62.51***($\pm$0.20)

Table 3.3. The test results on IEMOCAP, MELD, EmoryNLP and DailyDialog datasets. HVAD-SCCL denotes our SCCL method utilising the utterance-level VAD labels, and NRC-SCCL denotes SCCL with the NRC-VAD supervision signals. All SCCL results are with LinAdapter. Best values are highlighted in bold. The numbers with $*$ indicate that the improvement of our model over all baselines is statistically significant with p $<$ 0.05 under t-test.

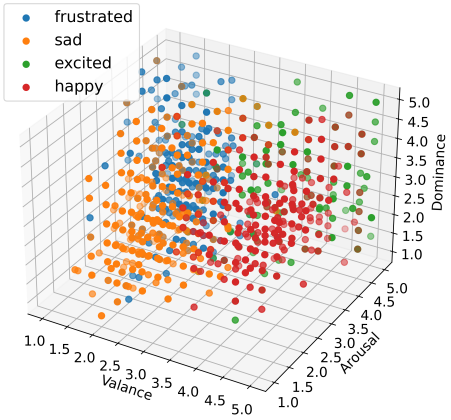


Figure 3.3. Visualisation of HVAD annotations in IEMOCAP training set.

According to the results, BERT-Large and DialogXL achieved competitive results on all datasets. These PLM-based methods are usually used as foundations for other works. KI-Net,

COSMIC and SKAIG all explicitly incorporate factual knowledge at the token level or mental state knowledge at the utterance level, and achieve competitive performance especially on short-context datasets, such as over 57% on DailyDialog (7.9 utterances per dialogue). SGED also implicitly models speaker information via an encoder-decoder framework, which leads to a balanced improvement on all datasets and the best performance 40.24% on EmoryNLP. These results demonstrate that infusing task-related knowledge and information is beneficial for ERC task. Though RGAT and DAG-ERC both utilise graph structure to model the context, DAG-ERC significantly outperforms RGAT with over 3% gain on all datasets, showing the importance of more reasonable dialogue modelling structures. The competitive performance of CoG-BART also shows the effectiveness of other representation learning techniques such as supervised contrastive learning and response generation.

We can only test HVAD-SCCL on IEMOCAP since all other datasets do not provide human-labelled utterance-level VAD scores. According to the results, HVAD-SCCL achieves a new state-of-the-art result of 69.88%, but outperforms NRC-SCCL slightly on IEMOCAP, which does not correspond to our early hypothesis. We notice that NRC-VAD follows strict best-worst scaling annotation and aggregation processes with a minimum of 6 annotators per word. In contrast, the IEMOCAP VAD (HVAD) annotation process follows a rough scheme, which brings inaccuracy to the annotated labels and only provides coarse-grained VAD information within each emotion. According to the visualisation results in Figure 3.3, the VAD shifts within each emotion are mostly discrete, which leads to a limited advantage over the fixed NRC-VAD prototypes. In addition, the VAD distributions of semantically similar emotions (e.g. *Frustrated* and *Sad*) appear to be more entangled, which increases confusion during the training process.

On the other hand, NRC-SCCL obtains competitive results on all datasets, and achieves new state-of-the-art results 69.81% on IEMOCAP, 65.70% on MELD and 62.51% on DailyDialog. Specifically, NRC-SCCL outperforms all information infusion-based models on three datasets with linguistic knowledge and NRC-VAD emotion prototypes, showing the effectiveness of these information. It also improves the performance of CoG-BART by over 3% on IEMOCAP and 6% on DailyDialog, indicating the advantage of SCCL over vanilla supervised contrastive learning and response generation. However, on EmoryNLP, NRC-SCCL fails to outperform the baseline models as in the other datasets. A possible reason is that EmoryNLP defines fuzzy emotions *powerful* and *peaceful*. Though appearing highly positive in NRC-VAD (*Powerful*: [0.865,0.830,0.991], *peaceful*: [0.867,0.108,0.569], as listed in Table 3.2), we find that many utterances labelled with these emotions do not yield positive sentiments. Therefore, unified VAD prototypes of the fuzzy emotions are misleading for many samples.

We provide some cases in Table 3.4 to explain the above hypothesis. All examples are from the training set of EmoryNLP. In the samples of *peaceful*, utterance #1 expresses no apparent emotions with a moderate Valance score 0.460, utterance #2 conveys weak sadness and anger with lower Valance 0.317 and higher Dominance 0.470, and utterance #3 shows implicit happiness with higher Valance 0.752. In the samples of *powerful*, though all utterances

Emotion	Utterances
peaceful **(0.867,0.108,0.569)**	1. Well...you never know. How's. um... how's the family? **(0.460,0.249,0.355)** 2. Warden, in five minutes my pain will be over. But you'll have to live with the knowledge that you sent an honest man to die. **(0.317,0.470,0.261)** 3. Yeah, I'm sorry too. But, I gotta tell you. I'm a little relieved. **(0.752,0.696,0.410)** 4. Oh, like you've never gotten a little rambunctious with Ross. **(0.341,0.527,0.497)** 5. Yeah, I'm thinking, if we put our heads together, between the two of us, we can break them up. **(0.770,0.194,0.712)**
powerful **(0.865,0.830,0.991)**	1. ..Dammit, hire the girl! Okay, everybody ready? **(0.483,0.936,0.898)** 2. Okay, everybody, we'd like to get this in one take, please. Let's roll it... water's working... and... action. **(0.640,0.794,0.519)** 3. I'm on top of the world, looking down on creation and the only explanation I can find, is the wonders I've found ever since... **(0.720,0.584,0.716)** 4. Alright, I looked all over the building and I couldn't find the kitty anywhere. **(0.325,0.764,0.372)** 5. My God, you're choking! That better? **(0.322,0.905,0.569)**

Table 3.4. Some samples of the fuzzy emotion *peaceful* and *powerful* that shift in Valance-Arousal-Dominance. We provide the NRC-VAD emotion prototypes for the two emotions, and the VAD predictions of each utterance in a random run of Lin-SCCL.

express high Arousal (emotion intensity) which corresponds to the NRC-VAD emotion prototypes, these examples have different Valance and Dominance levels. For example, utterance #1 has high dominance 0.898 with a strong sense of control, but utterance #2 and #4 show relatively low dominance 0.519 and 0.372. On the other hand, utterance #3 conveys high Valance 0.722 with apparent pleasantness, while utterance #4 and #5 express sadness and fear with low Valance 0.325 and 0.322. Therefore, the model is unable to learn fine-grained shifts in fuzzy emotions with the unified NRC-VAD emotion prototypes. One direction of our future work is leveraging more fine-grained supervision signals to handle the change of situations for fuzzy emotions.

3.6.2 Ablation Study

We investigate the performance of each proposed module via an ablation study on Lin-SCCL in Table 3.5. According to the results, Lin-SCCL outperforms the context-aware utterance encoder by over 3% on IEMOCAP and DailyDialog, and over 2% on MELD and EmoryNLP. These improvements show the joint contribution of linguistic knowledge and SCCL. While the removal of either SCCL or LinAdapter decreases the model performance, removing SCCL leads to a more serious over 1.5% drop for IEMOCAP, MELD and DailyDialog. According to the previous analysis in NRC-VAD emotion prototypes, SCCL is expected to be beneficial

Model	IEMOCAP	MELD	EmoryNLP	DailyDialog
Fac-SCCL	69.66	65.10	37.85	61.89
-SCCL	68.25(↓1.41)	64.20(↓0.90)	37.10(↓0.75)	60.64(↓1.25)
Lin-SCCL	**69.81**	**65.70**	**38.75**	**62.51**
-SCCL	68.21(↓1.60)	63.70(↓2.00)	38.53(↓0.22)	60.38(↓2.13)
-Adapter	69.23(↓0.58)	64.72(↓0.98)	37.45(↓1.30)	61.53(↓0.22)
-SCCL,Adapter	66.52(↓3.29)	63.44(↓2.26)	36.68(↓2.07)	59.32(↓3.19)
Lin-RL	68.70(↓1.11)	64.65(↓1.05)	38.12(↓0.63)	61.24(↓1.27)

Table 3.5. Results of ablation study for two knowledge types. Lin-SCCL denotes the SCCL method with LinAdapter and Fac-SCCL is with FacAdapter. Lin-RL replaces the SCCL with a correlation-based regression loss on the VAD scores. All experiments use the NRC-VAD supervision signals. "-" denotes the removal of one or several modules, and "Adapter" denotes the adapter module. The values in parentheses indicate the relative change with respect to Lin-SCCL and Fac-SCCL. We omit the repeated results for Fac-SCCL. Best values are highlighted in bold.

in distinguishing similar emotions, which is crucial for ERC and leads to more improvement than LinAdapter on these datasets. On EmoryNLP, LinAdapter benefits model performance more significantly than SCCL since the fuzzy emotions affect the contrastive learning process in VAD space, as analysed in Sec. 3.6.1. Lin-SCCL outperforms Lin-RL by over 1% on most datasets, showing SCCL as more appropriate for leveraging VAD information. A possible reason is that regression loss only introduces the current emotion's cluster-level representation, while SCCL also introduces and pushes apart all other emotion prototypes. SCCL further enables the model to be aware of the quantitative information between each pair of emotions.

3.6.3 Empirical Comparison of Knowledge Adapters

We analyse the effect of linguistic knowledge and factual knowledge on SCCL and the utterance encoder by comparing their performance in ERC, where the results are shown in Table 3.5. According to the results, both LinAdapter and FacAdapter contribute to the performance positively, denoting the effectiveness of both knowledge types. Lin-SCCL outperforms Fac-SCCL on all four datasets, because linguistic knowledge provides utterance structure information to help discover the linguistic patterns for emotion expression, which benefits the contrastive learning process. On the other hand, much factual knowledge is unrelated to affect and brings noise to the fine-grained emotion reasoning in SCCL. With the removal of SCCL, the utterance encoder achieves superior results on MELD and DailyDialog with FacAdapter, since the factual knowledge enriches the semantics of utterances, which benefits the dialogues with short contexts. This hypothesis is further indicated by the more significant improvement with LinAdapter on the other two rich-context datasets IEMOCAP and EmoryNLP. Overall, the empirical comparison of both knowledge adapters verifies the more benefits of linguistic knowledge on SCCL, and factual knowledge provides more information to the utterance encoder in short-context scenarios.

3.6.4 Comparison of Contrastive Learning Methods

We compare the results of different contrastive learning methods with RoBERTa encoder in Figure 3.4.

Both large and base-size encoders are leveraged to compare the performance of encoders with different context modelling ability. "VADCL" denotes performing SCL directly on VAD space without introducing emotion prototypes, and "Random-SCCL" utilises the same structure as SCCL except *randomly initialising* the prototype for each emotion instead of utilising NRC-VAD.

For the results on RoBERTa-Large, SCCL outperforms the RoBERTa baseline with an improvement of 2.71% on IEMOCAP and 1.28% on MELD. VADCL achieves comparable performance with SCL on both datasets, proving the viability of performing contrastive learning on a low-dimensional space instead of the semantic space, which also provides useful information to facilitate the identification of emotions. SCCL also outperforms VADCL on both datasets, denoting that emotion prototypes guide samples of each emotion to cluster towards proper positions and maintain appropriate quantitative relations. To further analyse this hypothesis, we experiment on RoBERTa+Random-SCCL, and Random-SCCL yields worse outcomes than RoBERTa on both datasets. These results indicate that SCCL relies on emotion prototypes instead of merely clustering the same emotion as in SCL. The quantitative information embedded in the prototype of each emotion is eliminated as the consequence of the random initialisation, and these false relations mislead SCCL.

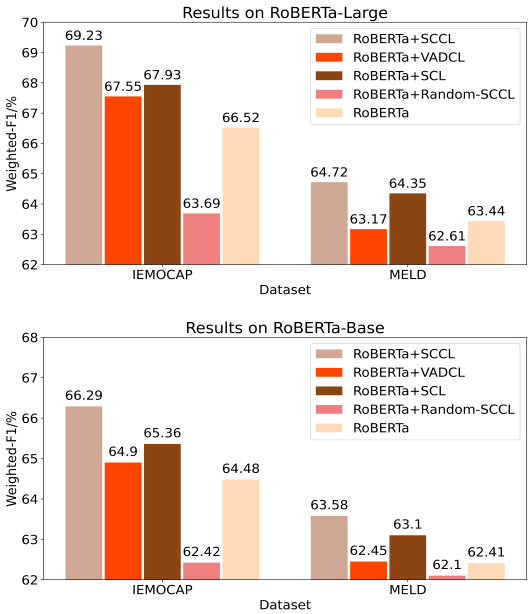


Figure 3.4. Performance of different contrastive learning methods with RoBERTa-Large and RoBERTa-Base encoder. Test performance is reported with tuning on the dev set.

We also present the results with RoBERTa-Base encoder. As expected, RoBERTa-Large outperforms RoBERTa-Base with all contrastive learning methods. Similar conclusions about the comparisons of contrastive learning methods are drown from the results of RoBERTa-Base, showing that our above conclusions are robust with utterance representations of vary-

ing quality. In addition, the advantage of SCCL is more apparent on IEMOCAP, showing the consistent benefits of rich context on SCCL with different utterance encoders.

3.6.5 Batch Size Stability

With the change in batch size, we compare the training stability of SCCL, VADCL and SCL in Weighted-F1 scores on IEMOCAP. The results are shown in Figure 3.5. Due to the limitation in computational resources, we utilise RoBERTa-Base as encoder and range the batch size from 2^0 to 2^4.

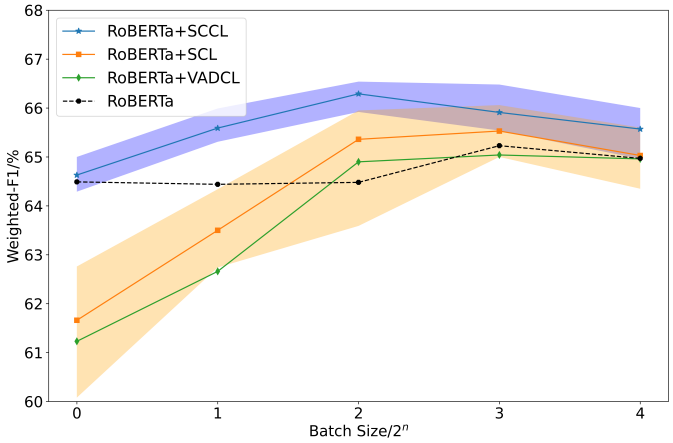


Figure 3.5. Change of F1 scores with different batch sizes on IEMOCAP, using RoBERTa-Base as the encoder.

According to the results, RoBERTa achieves the most stable outcomes as the batch size changes, with a Standard Derivation (SD) of 0.32%. SCCL obtains 0.82% better results than RoBERTa on average, and performs stable as the batch size change, with a SD of 0.66%. This result shows that emotion prototypes obtained from NRC-VAD provide a fixed clustering direction for samples of each emotion. Therefore, the model does not need a large amount of observations at each training step for a stable convergence.

For SCL, while the results remain competitive and stable with large batch sizes, the performance drops fast below the RoBERTa baseline as the batch size decreases, leading to a high SD of 1.40%. In the extreme case where the batch size drops to 1, SCL fails to converge and brings noise to the training process, resulting in a severe 2.83% drop compared to RoBERTa. We also provide the variation scale at each batch size for SCCL and SCL. The results show that SCCL has relatively low variances compared to SCL, especially with small batch sizes. This result shows the benefit of NRC-VAD emotion prototypes and the low-dimensional contrastive space, which relieves the curse of dimensionality problem.

VADCL suffers from the similar problems as SCL, with the highest SD of 1.67%. In addition, VADCL performs worse than SCL with small batch sizes. When observing only a few samples at each training step, the model fails to extract effective features in the three-dimensional space without emotion prototypes as the guidance.

3.6.6 Visualisation in VAD Space

With the three-dimensional VAD space, we are able to directly visualise the predictions instead of utilising dimension reduction techniques. Each VAD prediction also reflects the model's corresponding emotion reasoning process from the Valence-Arousal-Dominance perspective, which benefits interpretability. We present the key elements of the visualisation results on all four test sets in Figure 3.6.

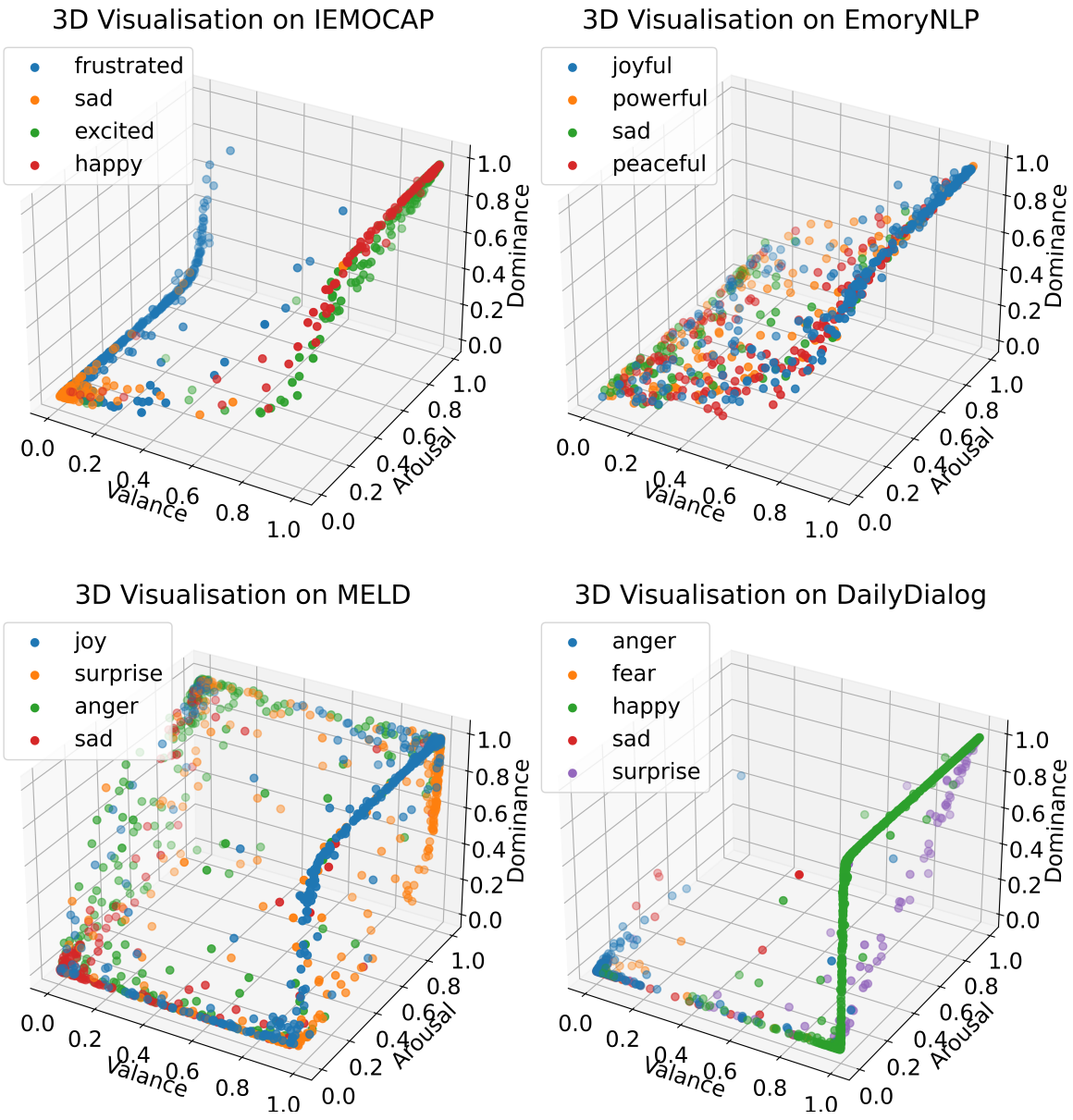


Figure 3.6. Key elements of the VAD visualisation results on all test sets. We only present the samples of representative emotions to provide a more intuitive view.

For IEMOCAP, we select and present semantically similar emotions (e.g., *excited* and *happy*) to gain clearer insights to the effect of SCCL, demonstarting their relationships to each other. For dissimilar emotions such as *happy* and *sad*, Valance alone separates them well enough. In addition, similar emotions are also well distinguished in VAD space by Arousal and Dominance, which corresponds with our early hypothesis. For example, *frustrated* and *sad* significantly vary in terms of Arousal, and *happy* and *excited* are jointly divided by Arousal and Dominance.

In section 3.6.1, we speculate that SCCL provides less improvement to EmoryNLP due to the fuzzy emotions where the VAD prototypes vary in different situations. In the visualisation on EmoryNLP, we present the two fuzzy emotions *powerful*, *peaceful* and two relatively invariant emotions *joyful*, *sad* to provide an intuitive comparison. According to the results, the model makes accurate and well-clustered VAD predictions for samples of *joyful* and *sad*, while the predictions of *peaceful* and *powerful* spread across the VAD space and fail to cluster.

The visualisation results of MELD and DailyDialog shows similar well-separated samples of emotions, such as *joy*/*happy* and *surprise*. However, the predictions of several emotions are inaccurate and not well clustered (e.g., *anger* and *sad* in MELD, *fear* in DailyDialog). We notice that the label distribution of both MELD and DailyDialog is highly imbalanced. Training samples of *sad* cover merely 6.8% in MELD. In DailyDialog, over 60% of utterances are labelled with *neutral* or *happy*, while the ratio of *fear* and *sad* are both below 5%. Therefore, another direction of future work is to handle the lack of training samples caused by label imbalance for SCCL. In addition, emotions such as *anger* and *sad* are often expressed implicitly, which is closely dependent on the context. Therefore, the lack of contextual information in MELD and DailyDialog brings more challenges to the prediction of these emotions. Overall, the above visualisation results correspond with other experimental outcomes.

3.7 Summary

In this chapter, based on a PLM utterance encoder, we propose a low-dimensional supervised cluster-level contrastive learning model for emotion recognition in conversations. We reduce the high-dimensional supervised contrastive learning space to a three-dimensional space Valance-Arousal-Dominance, and incorporate VAD prototypes from the emotion lexicon NRC-VAD by proposing the novel SCCL method. In addition, we infuse linguistic knowledge and factual knowledge into the context-aware utterance encoder by utilising the pre-trained knowledge adapters. Though pre-trained knowledge adapters are not modified in anyway, we are the first to successfully apply them to ERC.

Experimental results show that our method achieves new state-of-the-art results on three datasets IEMOCAP, MELD, and DailyDialog. Ablation study proves the effectiveness of each proposed module, and further analysis indicates that VAD space is an appropriate and interpretable space for SCCL. Emotion prototypes from NRC-VAD provide useful quantitative information to guide SCCL, which improves model performance and stabilises the training process. The knowledge infused by pre-trained knowledge adapters also enhances the performance of the utterance encoder and SCCL. In the future, we will leverage more fine-grained supervision signals to handle fuzzy emotions, and develop efficient methods to alleviate label imbalance and lack of context problems for SCCL.

Chapter 4

Mental State Knowledge Infusion

4.1 Overview

Similar to other text mining tasks dealing with long sequences, context modelling ability is crucial for stress and depression detection. Early works utilise CNN[139] or RNN[140], [141] to capture long-dependency semantic information from posts. In recent years, the Transformer-based PLMs [49], [75], [170] have shown their strong context modelling ability, leading to the popularity of the pretraining-finetuning paradigm in mental health conditions detection. However, stress and depression detection are still more complicated than other related tasks, such as vanilla emotion recognition, since it requires fine-grained modelling of the speaker's mental states. Mental states are defined as the states of mind of a person, such as intention, reaction and belief. For example, with the post of a depressed speaker *I honestly have no idea how a day is gonna go anymore*, the reaction *feel sad* and intention *intend to complain about life* clearly reflect the depression tendency of the speaker. In psychology, researchers [171], [172] use questionnaires to evaluate an individual's mental status and tendency to depression. For example, the Center for Epidemiologic Studies Depression Scale [173] utilises questions related to mental states such as *how often do you feel lonely?*. In deep learning-based methods, existing research has leveraged external knowledge such as emotional information [152], [153] and user intention features [174] to aid the mental state modelling process. In addition, Ji et al.[157] collect a large amount of data on mental health from social media platforms and fine-tune them on the pre-trained BERT to implicitly infuse mental state knowledge.

Although previous methods have presented promising results, they overlook several vital factors. Firstly, existing methods mainly focus on leveraging semantic information or emotional features to model mental states that are not explicable or controllable implicitly. Secondly, even if we have explicitly modelled mental states, the model can still lack mentalisation [175], [176] ability (the ability to understand the mental states of others) to select and interpret mental states correctly. For example, with reaction *feel lonely* and intention *intend to ask questions*, the model needs to mentalise that *feel lonely* is more related to mental health conditions. A third challenge is developing more explicit ways to capture class-specific features from the post embeddings since the post encoders are found to poorly capture the semantics of sentences [159] without carefully fine-tuning in many other NLP tasks.

To tackle the above challenges, we propose a mental state **K**nowledge-aware and **C**ontrastive **Net**work (KC-Net) for early stress and depression detection. Based on a context-aware post encoder for the first challenge, we leverage a generative commonsense knowledge base called COMET [54] that provides the participants' mental state descriptions of various aspects given a post as input. We call these descriptions as mental state knowledge. It is expected to model the mental states of speakers explicitly. Mental state knowledge is infused in post embeddings using GRU models. For the second challenge, we improve the model's mentalisation ability by introducing knowledge-aware dot-product attention, allowing the model to attentively select mental state knowledge aspects most relevant to the current reasoning process. To solve the third problem, we employ supervised contrastive learning, which pushes together representations with the same label, and repels those with different labels. We thoroughly leverage label information to mine the class-specific features. At the same time, we expect the more representative post embedding, also used for querying in knowledge-aware dot-product attention, will perform better during mentalisation. We validate our method on a depression detection dataset, a stress detection dataset and a stress factors recognition dataset. Experimental results show that our model consistently outperforms the strong baselines and achieves new state-of-the-art results on all three datasets.

In summary, this paper makes the following contributions:

- For the first time on stress and depression detection, we propose to explicitly model the mental states of speakers by leveraging mental state knowledge from COMET explicitly.
- We introduce a mentalisation module based on knowledge-aware dot-product attention to enhance the model's ability to understand and utilise the introduced mental state knowledge.
- We discuss the necessity to capture class-specific features and utilise supervised contrastive learning to leverage label information for this purpose fully.
- Our method achieves new state-of-the-art results on three stress and depression detection datasets and each of the proposed modules is proved effective. Further analysis also explains and visualises the mechanism of these modules.

We provide an overview of our method in Figure 4.1. The framework mainly contains data pre-processing, post encoding, mental state feature extraction, knowledge-aware mentalisation based on dot-product attention, and joint training of stress and depression detection and contrastive learning. We first (a) collect and (b) pre-process the raw data. Then we (c) employ the context-aware post encoder to obtain deep bidirectional word embeddings of the input post. Thirdly, we (d) extract mental state knowledge and infuse it into the network to aid the (e) mentalisation process. Specifically, after extracting the knowledge, we combine it with the post embeddings using GRUs and utilise knowledge-aware dot-product attention to focus on the most relevant knowledge aspects. Finally, we employ (f) supervised contrastive learning as the auxiliary task, together with the (g) stress and depression detection main task, for (h) jointly training in a multi-task learning manner.

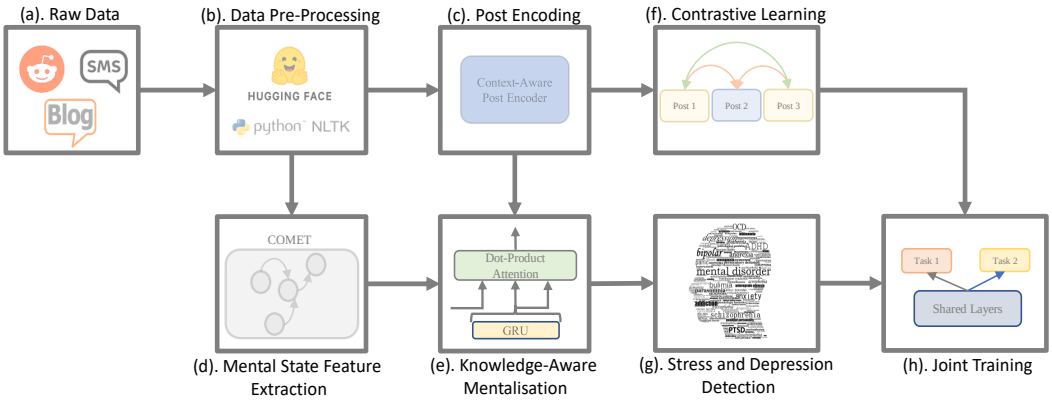


Figure 4.1. Overview of our stress and depression detection framework.

4.2 Post Encoding

4.2.1 Data Pre-Processing

Each post in the training data consists of multiple sentences without segmentation, while the extraction and infusion of mental state knowledge both require sentence-level representations. To facilitate the input construction of the knowledge encoder and the concatenation of mental state knowledge embeddings, we use NLTK[1] sentence tokenizer to segment the post into sentences. Thus, the ith post in the data $\mathbf{X}^i = \{\mathbf{X}^i_1, \mathbf{X}^i_2, ..., \mathbf{X}^i_{\mathbf{N}^i}\}$, where $\mathbf{X}^i_j$ is the jth sentence, and $\mathbf{N}^i$ denotes the number of sentences in $\mathbf{X}^i$.

To make the post encoder context-aware, the model input needs to cover the whole post at once. We rejoin all post elements in $\mathbf{X}^i$ with $\langle/s\rangle$, which denotes end-of-sentence token in the post encoder. We also prepend the rejoined input with a start-of-sentence token $\langle s\rangle$. The final input is $\hat{\mathbf{X}}^i = \{\langle s\rangle ; \mathbf{X}^i_1; \langle/s\rangle ; ...; \mathbf{X}^i_{\mathbf{N}^i}; \langle/s\rangle\}$, where ; denotes concatenation. Since the embedding look-up process of the encoder requires a token-level representation of the post, we use the token-level tokenizer provided by HuggingFace[2] to tokenize $\hat{\mathbf{X}}^i$: $\hat{\mathbf{X}}^i = \{\hat{\mathbf{X}}^i_0, \hat{\mathbf{X}}^i_1, ..., \hat{\mathbf{X}}^i_{K^i}\}$, where $\hat{\mathbf{X}}^i_k$ denotes the kth token, and K^i denotes the number of tokens in $\hat{\mathbf{X}}^i$. We also record the position of the start-of-sentence and each of the end-of-sentence tokens $\langle s\rangle$ and $\langle/s\rangle$ in $\hat{\mathbf{X}}^i$: $P^i = \{0, P^i_1, P^i_2, ..., P^i_{n^i}\}$, where each P^i_k is the k-th separation position, and n^i denotes the number of separation tokens in $\hat{\mathbf{X}}^i$. If the overall token numbers exceed the maximum input length, we truncate it to fit the post encoder.

4.2.2 Context-Aware Post Encoder

In recent years, we have witnessed the huge success of utilising PLMs [49], [75], [170] as sentence encoders for fine-tuning various downstream tasks, such as text classification [177] and text generation [178]. One advantage of these Transformer-based PLMs in text modelling

[1]https://www.nltk.org/
[2]https://huggingface.co/

is that they are context-aware and allow direct and full interactions between long-distance elements, which is important in long-sequence posts. The semantic information and knowledge learned during pre-training also help to enrich sentence-level representations. Furthermore, domain-specific pre-trained PLMs usually outperforms general language pre-trained PLMs on domain-related tasks [179], [180]. MentalRoBERTa [157] is trained on mental health posts crawled from social media and customised for detecting mental health conditions. It outperforms RoBERTa on several related datasets due to domain-related knowledge introduced during pre-training. Therefore, we propose a Context-Aware Post (CAP) encoder based on MentalRoBERTa to obtain token-level embeddings.

Specifically, with the pre-processed input $\hat{\mathbf{X}}^i$, the CAP encoder uses L layers of Transformer to get the input representations. For convenience, we denote the process as:

$$\mathbf{H}^i = post\_encoder(\hat{\mathbf{X}}^i, L) \tag{4.1}$$

where $post\_encoder$ denotes the CAP encoder, and $\mathbf{H}^i \in \mathbb{R}^{N \times D_h}$ denotes the outputs of Lth layer of the encoder, N denotes the sequence length, and D_h is the hidden dimension of the CAP encoder.

4.3 Mentalisation

This section introduces the extraction of the sentence-level mental state knowledge and how we infuse this knowledge into the stress and depression detection model. An overview of our approach is provided in Figure 4.2. It mainly contains two parts: (a) mental state knowledge feature extraction; (b) the mentalisation process, which includes the GRU-based knowledge combination and the attentive knowledge selection process. We utilise a generative transformer model for mental state knowledge, namely COMET [54], to extract knowledge features. Based on GPT [75], COMET has two versions, which are trained separately on ATOMIC [47] and ConceptNet [45]. ConceptNet is a word-level commonsense knowledge base, while ATOMIC is a large-scale collection of everyday inferential if-then knowledge in the form of textual descriptions. These descriptions mainly focus on the speaker and listeners of the input. In particular, there are nine different if-then aspects in ATOMIC. An input involving a speaker *S* and listeners (*others*) may include nine aspects: *intent of S*, *need of S*, *attribute of S*, *effect on S*, *wanted by S*, *reaction of S*, *effect on others*, *wanted by others*, and *reaction of others*. As an example, with the input *Person S gives him a compliment*, *intent of S* would be *S wanted to be nice*. With pre-training on ATOMIC, COMET can generate nine responses regarding the input and each aspect. These responses are similar to the knowledge in ATOMIC but are not bound by them. Considering the apparent mutual indications between mental states and mental health conditions, it is natural to select the COMET model trained on ATOMIC over ConceptNet, which mainly consists of general language concepts.

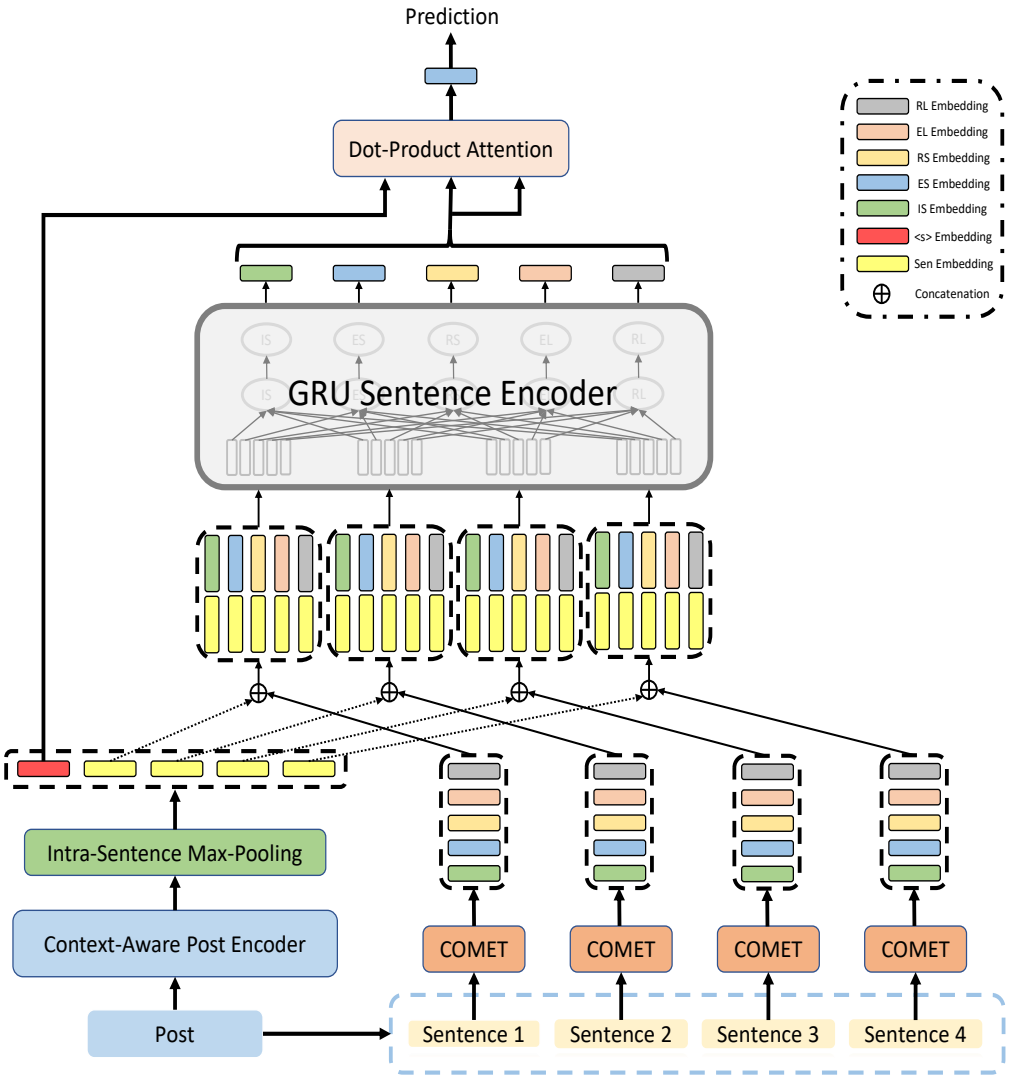


Figure 4.2. Overview of the mental state knowledge infusion process.

4.3.1 Feature Extraction

COMET generates nine if-then responses for each input, which cover most aspects of the participants' mental states. However, the nine aspects are not equally beneficial for mental state modelling. We select the following five aspects: *intent of S* (R_{IS}), *effect on S* (R_{ES}), *reaction of S* (R_{RS}), *effect on others* (R_{EL}) and *reaction of others* (R_{RL}), forming the selected aspect set $\mathbf{R}$. These aspects are selected mainly by considering the following two factors: (a) Sap et al.[47] clearly define *intent of S* (R_{IS}), *reaction of S* (R_{RS}) and *reaction of others* (R_{RL}) as mental states of the participants, which is directly related to our goal; (b) These five aspects have been selected and utilised in emotion-related tasks and achieved promising performance [103], [116] than other combinations. Considering the close relationship between emotions and mental states, this selection is expected to be appropriate.

We have obtained split sentences for post $\mathbf{X}^i$ (Section 4.2.1). Since COMET enforces each input event to be less than 17 tokens, we truncate the event after tokenisation:

$$\hat{\mathbf{X}}^i_j = \mathbf{X}^i_j[0 : min(|\mathbf{X}^i_j|, 17)] \tag{4.2}$$

Note that we slice segments of sentences that exceed the maximum position length of the CAP encoder, and we discard the corresponding knowledge of these sentences to avoid introducing noise. Sentence $\hat{\mathbf{X}}_j^i$ is regarded as input for knowledge extraction and concatenated with each of the selected aspect phrases in $\mathbf{R}$. As an example, to process with $\mathbf{R_{IS}}$, we have input $\{\hat{\mathbf{X}}_j^i; intent\ of\ \mathbf{S}\}$, which is then put into the encoder of COMET to obtain knowledge representations. We extract the activations from the final time step as the corresponding mental state embedding. Throughout the process, for $\hat{\mathbf{X}}_j^i$ we have a set of five knowledge vectors:

$$\hat{R}(\hat{\mathbf{X}}_j^i) = \{R_{IS}(\hat{\mathbf{X}}_j^i), R_{ES}(\hat{\mathbf{X}}_j^i), R_{RS}(\hat{\mathbf{X}}_j^i), R_{EL}(\hat{\mathbf{X}}_j^i), R_{RL}(\hat{\mathbf{X}}_j^i)\}$$

where $\hat{R}(\hat{\mathbf{X}}_j^i) \in \mathbb{R}^{5\times D_k}$, D_k denotes the knowledge embedding dimension. Instead of using them for response generation, we discard the COMET decoder and utilise the representations directly to enhance knowledge in post representations. We expect to adopt these mental-related variables in a unified model.

We have computed token-level CAP embeddings $\mathbf{H}$ (Section 4.2.2), with all separation positions Precorded (Section 4.2.1). In $\mathbf{H}^i$, we compute sentence-level representation $\hat{\mathbf{H}}_j^i$ for jth sentence by performing intra-sentence max-pooling with position record P^i:

$$\hat{\mathbf{H}}_j^i = max\_pooling(\mathbf{H}[P_{j-1}^i : P_j^i]) \tag{4.3}$$

where $max\_pooling$ denotes max pooling operation, $\hat{\mathbf{H}}_j^i \in \mathbb{R}^{D_h}$, and $[:]$ denotes the slicing operation. We concatenate each sentence representation separately with the extracted five mental state embeddings to preserve the semantic and knowledge information for knowledge-enhanced embeddings. As an example, $\hat{\mathbf{H}}_j^i$ produces $intent\ of\ \mathbf{S}$ embedding:

$$\mathbf{E}_{IS}^{ij} = [\hat{\mathbf{H}}_j^i; R_{IS}(\mathbf{X}_j^i)]$$

where $\mathbf{E}_{IS}^{ij} \in \mathbb{R}^{2D_h}$ is the knowledge-enhanced embedding. Therefore, each sentence X_j^i has an embedding set $\hat{\mathbf{E}}_j^i = \{\mathbf{E}_{IS}^{ij}, \mathbf{E}_{ES}^{ij}, \mathbf{E}_{RS}^{ij}, \mathbf{E}_{EL}^{ij}, \mathbf{E}_{RL}^{ij}\}$.

4.3.2 Knowledge-Aware Mentalisation

Given the sentence-level mental state knowledge-enhanced representations $\hat{\mathbf{E}}_j^i$, we propose a knowledge-aware dot-product attention to attentively select different aspects to enhance the mentalisation ability of our model. Because this process requires a post-level embedding for each aspect, we utilise 5 independent GRU models [181] to separately encode the 5 aspects of knowledge-enriched representations for a post. Each GRU model walks over one aspect representations of the sentences. For example, with embedding $\mathbf{E}^i$ and aspect $intent\ of\ \mathbf{S}$, we compute the post-level embedding:

$$\mathbf{O}_{IS}^i = GRU_{IS}(\mathbf{E}_{IS}^i, D_r) \tag{4.4}$$

where GRU_{IS} denotes the corresponding GRU model, D_r denotes the hidden state dimension of the GRU, $\mathbf{O}^i_{IS}$ denotes the hidden state of the final time step of the GRU, and $\mathbf{E}^i_{IS} = \{\mathbf{E}^{i0}_{IS}, \mathbf{E}^{i1}_{IS}, ..., \mathbf{E}^{i\mathbf{N}^i}_{IS}\}$. After the encoding process of 5 GRUs, for post X^i, we acquire a post-level embedding set with 5 aspect embeddings: $\hat{\mathbf{O}}^i = \{\mathbf{O}^i_{IS}, \mathbf{O}^i_{ES}, \mathbf{O}^i_{RS}, \mathbf{O}^i_{EL}, \mathbf{O}^i_{RL}\}$.

Each mental state aspect of the knowledge-enriched embeddings contributes to the mental state reasoning process, according to different situations of various posts. Thus, we utilise a scaled dot-product attention module on the 5 post-level knowledge-enriched embeddings. For each post, the word-level embedding $\mathbf{H}^i_{\langle s \rangle}$ of the start-of-sentence token $\langle s \rangle$ is used as the overall semantic representation. Therefore, we use $\mathbf{H}^i_{\langle s \rangle}$ as the query, and the aspect embeddings $\hat{\mathbf{O}}^i$ as keys and values:

$$\mathbf{F}^i = Softmax(\frac{\mathbf{H}^i_{\langle s \rangle} \cdot \hat{\mathbf{O}}^{i\top}}{\sqrt{D_h}}) \cdot \hat{\mathbf{O}}^i \tag{4.5}$$

where $\mathbf{F}^i$ denotes final post embedding of X^i, $Softmax$ denotes softmax operation, $\cdot$ denotes dot product operation.

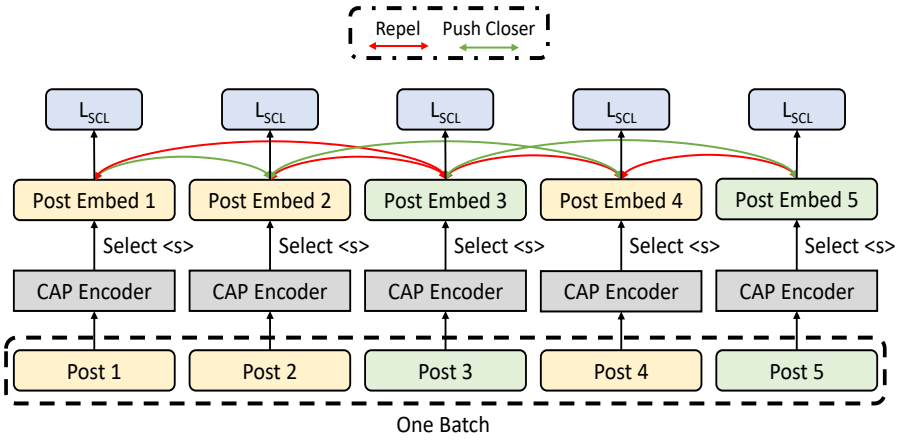


Figure 4.3. Overview of the supervised contrastive learning module.

4.4 Instance-Level Contrastive Learning

Stress and depression have specific features, which can be used for diagnosis, such as alcohol use and eating disorders [182]. Recognition of these class-specific features usually requires multi-modal information such as facial expression [183], while we only have access to text-based posts. Thus, the CAP encoder must fully utilise linguistic features that are discriminative in mental health identification [28], [184]. Unfortunately, the original post embedding $\mathbf{H}^i_{\langle s \rangle}$ from CAP encoder cannot capture the semantics of sentences [159] without careful fine-tuning. Recent works used unsupervised contrastive learning to solve this issue [42], [82]. We also employ contrastive learning, but in a supervised manner, to fully leverage the label information for capturing class-specific features. We provide an overview in Figure 4.3. The intuition of supervised contrastive learning is to make sentences with the same label cohesive and different labels mutually exclusive. As a result, we expect the model to correctly extract

key features in contrast with posts from the same and different categories. Since the post embeddings are also utilised as queries in knowledge-aware dot-product attention (Section 4.3.2), we also expect the contrasted embeddings to perform better in mentalisation.

For each post X^i, we still utilise $\mathbf{H}^i_{\langle s \rangle}$ as the semantic representation. All the posts within the same training batch take part in the contrast process of X^i, where posts with the same label as X^i are considered as positive pairs and the ones with different labels are considered as negative pairs. For multi-class datasets such as SAD, there could be a class where only one sample exists in a batch, and it cannot be directly applied for contrastive learning. To solve the problem, inspired by Li et al.[84], we copy each post embedding $\mathbf{H}^i_{\langle s \rangle}$ as $\overline{\mathbf{H}}^i_{\langle s \rangle}$, where $\overline{\mathbf{H}}^i_{\langle s \rangle}$ is detached from gradient. For a batch $\hat{\mathbf{H}}_{\langle s \rangle} = \{\mathbf{H}^1_{\langle s \rangle}, \mathbf{H}^2_{\langle s \rangle}, ..., \mathbf{H}^{N_b}_{\langle s \rangle}\}$, where N_b denotes the batch size, we obtain a new batch $\widetilde{\mathbf{H}}_{\langle s \rangle} = [\hat{\mathbf{H}}_{\langle s \rangle}, \overline{\mathbf{H}}_{\langle s \rangle}]$ of size $2N_b$, where $[,]$ denotes concatenation in the first dimension. With this copy operation, each sample within one batch has at least one sample in the same category(the detached copy of itself). We then compute contrastive loss on this new batch:

$$sim(p, i) = log \frac{exp(\widetilde{\mathbf{H}}^p_{\langle s \rangle} \cdot \widetilde{\mathbf{H}}^i_{\langle s \rangle} / \tau)}{\sum_{a \in A(i)} exp(\widetilde{\mathbf{H}}^a_{\langle s \rangle} \cdot \widetilde{\mathbf{H}}^i_{\langle s \rangle} / \tau)} \tag{4.6}$$

$$L_{SCL} = \sum_{i \in I} -\frac{1}{|P(i)|} \sum_{p \in P(i)} sim(p, i) \tag{4.7}$$

where $i \in I = \{1, 2, ..., 2N\}$ denotes the sample index, τ indicates the temperature coefficient, $P(i)$ represents samples in the same category as sample $\widetilde{\mathbf{H}}^i_{\langle s \rangle}$ except itself, and $A(i)$ denotes all sample in $\widetilde{\mathbf{H}}_{\langle s \rangle}$ except $\widetilde{\mathbf{H}}^i_{\langle s \rangle}$.

4.5 Model Training

We combine the training of the stress and depression detection and supervised contrastive learning task in a multi-task learning setting. The training loss consists of 2 parts: (a) the output of the knowledge-aware dot-product attention $\mathbf{F}^i$ passes through a feed-forward network to obtain classification logits L_{MD} for computing cross-entropy of post X^i; (b) the supervised contrastive learning loss L_{SCL}. We formalize them as follows:

$$\hat{Y}^i = Softmax(FFN(\mathbf{F}^i)) \tag{4.8}$$

$$L_{MD} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} Y^{i,c} log \hat{Y}^{i,c} \tag{4.9}$$

$$L_o = \alpha L_{MD} + (1 - \alpha) L_{SCL} \tag{4.10}$$

Dataset	Category	Data Source	Avg. Sentence	Avg. Token	Train	Validation	Test
Depression_Mixed	Depression	Reddit & Blogs	13	264	2215	474	476
Dreaddit	Stress	Reddit	5	103	2270	568	715
SAD	Stress Factors	SMS	1	17	5548	617	685

Table 4.1. Summary of the datasets. If the original data does not have a validation set, we split a portion of the training set for validation.

where FFN denotes a feed-forward network, Y^i denotes the one-hot ground-truth label of X^i, C is the number of classes, α is the weight for controlling contribution of the 2 losses, and L_o denotes the overall training loss.

4.6 Experimental Settings

4.6.1 Datasets

In this paper, we evaluate and compare our method with others on three different publicly available datasets. We explain the building process of these datasets in this section, and their statistical details are listed on Table 4.1. We also provide some examples for each of the datasets on Table 4.2.

Depression_Mixed[3][23] A weakly-supervised depression detection dataset with 2765 posts, which are collected from subreddits of Reddit[4]. Each post is labelled as depression or not, and consists of a multiple-sentence monologue stating the speaker's background and current feelings. Specifically, the authors match posts with certain protocols (e.g., posts containing *I was just diagnosed with depression*) in the Depression Support subreddit, and collect other posts written by the same speaker within one month as depressive posts. Non-depressive posts are collected in a similar way from the Breast Cancer and Family and Friends subreddits. Another 400 blog posts are also collected from English depression forums [27].

Dreaddit[5][185] A human labelled stress detection dataset. The authors select five domains belonging to three major topics: financial need, mental illness, and interpersonal conflict, where members are likely to discuss stressful topics. Then ten related subreddits are utilised to collect posts, such as homeless and PTSD. The dataset includes 3553 posts, each consists of a multiple-sentence monologue stating the feelings of the speaker, and each post is labelled as stressful or not. The annotation process is done using Amazon Mechanical Turk[6], where workers are required to label 5 posts as *stress*, *not stress* and *unknown*, until each post is labelled by at least 5 workers. Finally, the label of each post is determined by vote and those with *unknown* label are discarded.

SAD[7][186] A human-labelled stress factor detection dataset. To determine the stressors, the authors derive an original stressor set from Holmes and Rahe Scale[187], and simplify

[3]https://github.com/Inusette/Identifying-depression
[4]https://www.reddit.com/
[5]http://www.cs.columbia.edu/ eturcan/data/dreaddit.zip
[6]`https://www.mturk.com/`
[7]https://github.com/PervasiveWellbeingTech/Stress-Annotated-Dataset-SAD

Dataset	Examples
Depression_Mixed	So one effect is that I get really, really, really sad about some things. I just saw a per**n get hit by the bus I'd j*** gotten off of; when I went to help it, it g** hit by another car and I head it's skull smash. I am absolutely devastated and can't stop myself crying at work. (Depression) Looking to start a business of having de***ions with people who want or need so****dy to talk to. Pr***s to be arranged. Feel free to get in touch if you are interested (Non-Depresson)
Dreaddit	But it's been 2 mo***s already this time. We did not speak for Christmas or new year. I'm lonely, sad, angry at the si***tion (not angry at him!) and the worst part is not being able to talk or even know what's going on. We did not fight be***e this so he's not angry at me. (Stress) Maybe a couple more days will get me back to normal. Definitely quit***g the alcohol. It's an obvious trigger. But yeah, just w**ted to ask his thread on your thoughts. Thanks (Non-Stress)
SAD	All these e**ra hours at work are driving me insane. (Work) Coronavirus.I am high risk be***se of asthma so I am worried (Health, Fatigue, or Physical Pain) A person I know is man***lating money owed. (Financial Problem) All t**s coursework I've had lately. (School)

Table 4.2. Some examples of the three datasets. The posts have been paraphrased and obfuscated for user privacy.

it by labelling collected messages from chat-bot history. Further simplification and data annotation are done by two rounds of human intelligence tasks on Amazon Mechanical Turk. As a result, the stressor set includes nine stressor categories: T={'School', 'Financial Problem (Finance)', 'Family Issues (Family)', 'Social Relationships (Social)', 'Work', 'Health, fatigue, or physical pain (Health)', 'Emotional Turmoil (Emotion)', 'Other', 'Everyday Decision Making (Decision)'}. The authors also notice that some categories have low cardinality. Therefore, they randomly select messages from the low-cardinality categories, and scrape more data that have similar sentence embeddings with these messages to expand the dataset. Finally, the dataset contains 6850 SMS-like sentences, where each post is a short message (normally one sentence) divided into one of these categories.

4.6.2 Model Summary

We compare our model with the following baselines:

CNN [188]: A three channel convolution as the encoder, which has 128 dimensional features and filters of 3,4,5. Post-level embeddings are obtained using max-pooling and features of different filters are concatenated for decoding.

GRU [189]: A two-layer uni-directional GRU is utilised as the encoder. The hidden states of the final time-step are used for decoding.

BiLSTM_Att [190]: Attention mechanism is used on the output of a bidirectional LSTM network. The attended hidden states are used for decoding.

LR+Features [191]: Logistic Regression (LR) combining selected linguistic features (LIWC features, n-gram features, etc.).

EMO_INF [152]: An emotion-infused model which leverages emotion prediction as the auxiliary task to improve stress detection. The emotion prediction model is fine-tuned on GoEmotions dataset[192].

BERT [49]: Initialized from the pre-trained weights of BERT-base. The embedding of the '[CLS]' token is used for decoding.

RoBERTa [50]: Initialized from the pre-trained weights of RoBERTa-base. The embedding of the '$\langle s \rangle$' token is used for decoding.

MentalRoBERTa [157]: Initialized from the pre-trained weights of MentalRoBERTa. The embedding of the '$\langle s \rangle$' token is used for decoding.

4.6.3 Experiment Configuration

We conducted all experiments on a E5-2630L v3 CPU with 30GB of memory, and a Geforce RTX 3060 GPU with 12 GB of memory. For hyper-parameter setting, we set $D_h = D_k = 768$. The maximum input length of the post encoder is 512. To facilitate further processing, we still keep the hidden state dimension of the GRU D_r as 768, which is identical to the post and knowledge embeddings. The framework and initial weights of the PLMs come from Huggingface's Transformers [193]. We employ AdamW [194] optimizer for model training, with a batch size N_b of 8 on SAD, Dreaddit, and 4 on Depression Mixed. We use a learning rate of 1e-5, and set a dropout rate of 0.3 on all experiments. The evaluation metrics are chosen as precision(P), recall(R) and F1 measures. All the results are obtained in text modality only. The results reported in our experiments are all based on the average of 3 random runs on the test set.

4.7 Performance Comparison

This section presents the overall and factor specific experimental performances of the baseline models and our methods on the three datasets. We also provide the ablation analysis to prove the effectiveness of the proposed modules.

4.7.1 Overall Results

The overall experiment results of our model and the baselines are listed in Table 4.3 and Table 4.4, where '+RoBERTa' denotes replacing the CAP encoder with RoBERTa to explore the performance of our method on models with different levels of domain expertise, and 'C-Net' and 'K-Net' denote the **C**ontrast module and **K**nowledge infusion module separately implemented on the CAP encoder. According to the results on Depression_Mixed and Dreaddit.

PLM-based models such as BERT and RoBERTa perform a general advantage with over 90% performance on Depression_Mixed and 80% on Dreaddit. We also notice that the KC-Net based on RoBERTa outperforms MentalRoBERTa on both datasets, which shows the advantage of explicit mental state modelling and mentalisation process over implicit infusion of domain-specific knowledge. KC-Net achieves 95.4% of F1 scores on Depression_Mixed, and 83.5% on Dreaddit, which are both new state-of-the-art results, with over 2% improvements over the PLM-based models. It indicates the advantage of our method on longer sequences (with over 100 tokens per post) and complex contexts.

As for the results on SAD, PLM-based models still outperform other baselines by achieving over 74% of F1 scores, indicating the strong semantic modelling ability of deep, pre-trained models. MentalRoBERTa, the current state-of-the-art model, achieves over 75% F1 scores, since it introduces domain-specific knowledge in the post-training phase. For our methods, C-Net and K-Net both outperform previous methods on RoBERTa and CAP encoder, which separately prove the effectiveness of contrastive learning and mental state knowledge infusion. KC-Net still outperforms C-Net and K-Net on RoBERTa, but K-Net on the CAP encoder achieves a new state-of-the-art F1 result 77.8%, and outperforms KC-Net. One possible reason is that with domain-specific knowledge, CAP encoder possesses higher class-specific feature mining ability than RoBERTa, which already satisfies the need for short SAD posts with simple semantics and no contexts. Thus, the advantage of contrastive learning fades on these simple posts.

4.7.2 Factor-Specific Results

We present the F1-measure performance of our models and baselines on each stress factor in SAD on Table 4.4. Our method achieves the best results on most factors and gives a balanced performance. We believe the infusion of mental state knowledge and supervised contrastive learning both contribute to the performance. To gain a clearer view of their effect, we focus on each factor's proportion of contribution to the improvement of C-Net and K-Net over the CAP encoder.

	Depression_Mixed			Dreaddit		
Model	P	R	F1	P	R	F1
CNN[188]	85.2	85.1	85.1	70.1	68.8	68.5
GRU[189]	84.4	84.4	84.4	71.2	69.4	69.9
BiLSTM_Att[190]	90.4	95.0	92.6	72.7	72.0	72.0
LR+Features[191]	89.0	92.0	89.0	73.5	81.0	77.0
EMO_INF[152]	-	-	-	81.7	81.7	81.7
BERT[49]	91.4	91.4	91.4	80.3	79.9	79.8
RoBERTa[50]	93.2	92.4	92.9	81.2	81.3	81.3
MentalRoBERTa[157]	93.4	93.0	93.3	82.1	81.8	81.9
KC-Net+RoBERTa	93.7	93.7	93.7	82.7	82.6	82.7
KC-Net (Ours)	**95.5**	**95.3**	**95.4**	**84.1**	**83.3**	**83.5**

Table 4.3. Performance comparisons on Depression_Mixed and Dreaddit. We highlight top-1 values in bold. '-' means the original paper does not give the corresponding result.

Model	School	Finance	Family	Social	Work	Health	Emotion	Other	Decision	P	R	F1
GRU	83.5	68.8	69.7	47.6	78.1	52.2	22.9	35.8	5.4	58.3	57.1	57.1
CNN	78.0	79.4	73.2	57.4	78.3	53.9	30.8	39.3	21.1	62.6	61.5	61.0
BERT	88.9	86.5	83.0	76.8	86.0	63.1	55.1	55.7	12.5	75.3	73.8	74.4
RoBERTa	87.3	83.9	81.6	77.3	86.7	74.9	53.7	51.7	30.0	72.6	76.2	74.9
MentalRoBERTa	88.2	84.4	85.7	76.3	88.0	76.8	52.9	53.7	9.9	74.8	76.5	75.3
C-Net+RoBERTa	88.1	87.9	84.9	79.4	87.0	72.9	56.6	59.9	13.5	75.5	77.0	76.4
K-Net+RoBERTa	89.0	85.1	84.7	79.7	86.3	**79.0**	54.6	56.1	6.7	77.3	75.7	76.2
KC-Net+RoBERTa	88.5	85.7	85.0	78.1	87.9	74.0	59.1	56.3	48.3	75.4	77.4	76.8
C-Net	88.5	87.4	86.2	78.2	89.0	74.7	54.6	54.7	13.4	75.4	77.1	76.6
K-Net	87.9	85.7	85.9	**81.3**	89.5	72.5	**61.2**	**63.0**	**48.9**	**78.7**	77.2	**77.8**
KC-Net(Ours)	**89.3**	**88.3**	**86.4**	79.4	**90.9**	68.8	57.7	59.5	30.8	75.6	**77.6**	77.0

Table 4.4. Performance comparison of ours, baselines, and state-of-the-art methods for F1 measures of each stress factor and the averages of P, R, F1 on SAD. We highlight top-1 values in bold.

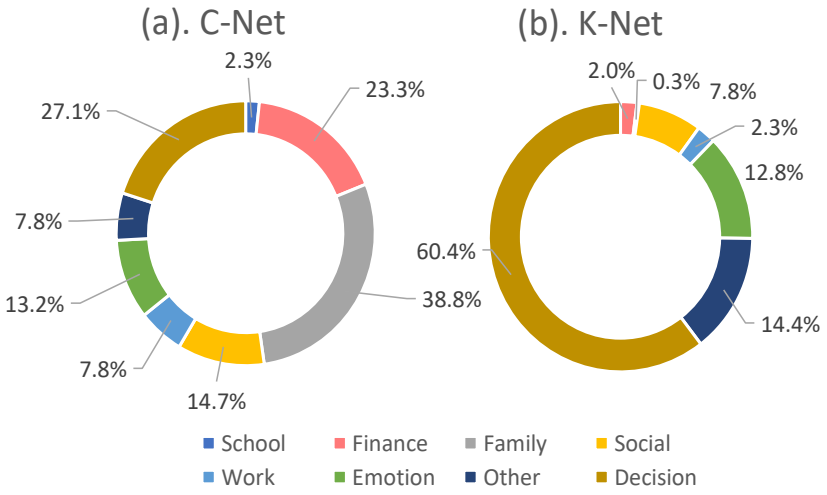


Figure 4.4. Each factor's proportion of contribution to the improvement of C-Net and K-Net over the CAP encoder. We ignore factors with no explicit improvement.

We first analyse the results on C-Net. It indicates that contrastive learning provides each factor with a relatively balanced improvement, with an average of 12.5% performance gain. This is reasonable since we expect contrast to fully leverage label information of each factor and extract their class-specific features. Compared to K-Net, contrastive learning also boosts more factors to gain solid improvement. It further shows that contrastive learning can universally increase the performance of each class. For labels with fewer training samples such as Decision, K-Net and KC-Net based models significantly outperform other models, with an average of over 20% performance gain. We believe the infusion of mental state knowledge has great benefit for lack of samples, and supervised contrastive learning can fully leverage the limited label information.

For mental state knowledge infusion, things are quite different. In contrast with C-Net, K-Net provides more imbalanced performance improvement, with an average of 16.6% performance gain and over 60% coming from the factor 'Decision'. We notice that the 'Decision' factor in the training set has the lowest proportion of sample numbers (less than 5%), which means the model receives less information from 'Decision' category during the training process. Nevertheless, we believe the infusion of COMET knowledge provides clear clues of mental states of the speakers, which are especially useful for low-resource categories in short post datasets, such as 'decision' in SAD. Therefore, we believe this advantage of K-Net is

Model	Depression_Mixed			Dreaddit		
	P	R	F1	P	R	F1
KC-Net+RoBERTa	**93.8**	**93.7**	**93.7**	**82.9**	**82.6**	**82.7**
– All Knowledge Modules	93.5	93.3	93.5(↓ 0.2)	82.8	82.3	82.5(↓ 0.2)
– Mentalisation Module	93.4	92.7	93.0(↓ 0.7)	81.5	82.0	81.7(↓ 1.0)
– Contrast Module	93.1	93.1	93.1(↓ 0.6)	82.5	82.1	82.4(↓ 0.3)
RoBERTa	93.3	92.4	92.9(↓ 0.8)	81.2	81.3	81.3(↓ 1.4)
KC-Net	**95.4**	**95.3**	**95.4**	**83.6**	**83.3**	**83.5**
– All Knowledge Modules	94.4	94.4	94.3(↓ 1.1)	82.4	82.2	82.3(↓ 1.2)
– Mentalisation Module	94.4	93.6	94.0(↓ 1.4)	81.9	82.0	81.9(↓ 1.6)
– Contrast Module	95.0	94.6	94.7(↓ 0.7)	83.2	82.6	83.0(↓ 0.5)
CAP Encoder	93.5	93.0	93.3(↓ 2.1)	81.9	81.8	81.9(↓ 1.6)

Table 4.5. The results of ablation study.

Model	Depression_Mixed			Dreaddit			SAD		
	P	R	F1	P	R	F1	P	R	F1
K-Net+RoBERTa	93.3	92.8	93.2	81.7	81.8	81.7	75.7	76.2	75.8
KC-Net+RoBERTa	92.9	93.3	93.0	81.4	82.8	82.1	75.8	75.2	75.6
K-Net	94.3	93.9	94.1	82.1	82.0	82.1	76.0	76.7	76.2
KC-Net	93.4	94.9	94.4	82.8	83.4	83.2	77.7	76.2	77.0

Table 4.6. The results of our methods with all nine knowledge aspects attended.

worth further exploration in few-shot learning scenarios.

4.7.3 Ablation Study

We perform ablation study of our model on both Depression_Mixed and Dreaddit datasets. 'All Knowledge Modules' denotes the removal of both mental state feature extraction and mentalisation module. '-Mentalisation Module' only removes the mentalisation module while keeping the mental state knowledge, and uses the average of the five knowledge-enhanced post embeddings for emotion disorders detection. '-Contrast Module' discards the contrastive learning module. Note that each "-" operation separately removes the corresponding module and there are no overlaps between operations. The results are shown on Table 4.5. The performance drops with each of the components removed. Especially, with the mentalisation module removed, the models consistently perform worse than the removal of all knowledge-related modules, which shows the importance of mentalisation in selecting the most relevant aspects of knowledge, instead of equally considering all aspects. Without the mentalisation process, the model is unable to filter out the noise brought by COMET or focus on the most useful aspects.

We also notice that in both datasets, compared with '-All Knowledge Modules' the performance drops more on KC-Net+RoBERTa when contrastive learning is removed (-Contrast Module), while on KC-Net the performance drops more when all knowledge modules is removed (-All Knowledge Modules). This proves that a higher level of domain expertise in the CAP encoder helps in both post embedding and mentalisation processes. On RoBERTa, the performance relies more on contrastive learning in understanding complex semantics and

extracting class-specific features. Besides, though KC-Net performs well on both encoders, the performance drops more on CAP encoder when KC-Net is completely removed. These results further indicate that the knowledge infusion and contrastive learning module could benefit more from encoders with rich domain-specific knowledge.

4.7.4 Empirical Analysis of Knowledge Aspects Selection

Though we have provided persuasive reasons for the selection of knowledge aspects, we perform empirical analysis by also testing our methods on three datasets with all nine aspects attended. The results are listed in Table 4.6. According to the results, all methods with all knowledge aspects attended perform slightly worse than ones with carefully chosen five aspects. A possible reason is that the remaining aspects are not closely relevant to mental state modelling, which brings noise to the mentalisation process. Specifically, short posts in SAD also do not convey enough information for complicated mental state reasoning of nine aspects, and the generative knowledge source may produce incorrect knowledge regarding the added aspects.

4.8 Discussion

In this chapter, though our proposed KC-Net achieves new state-of-the-art results on three datasets, there remains room for improvement. Therefore, we perform error analysis on the worst-performing dataset SAD. Now with a more explicable model architecture, we also evaluate the effect of each introduced module by analysing the outcomes from multiple different views.

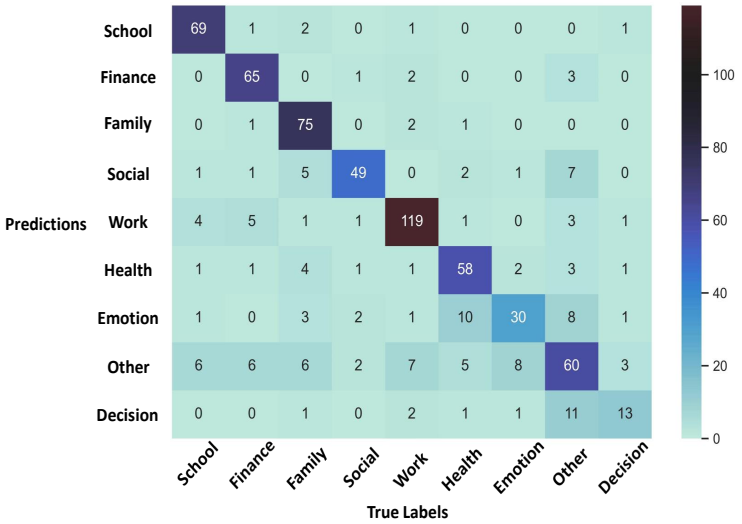


Figure 4.5. The confusion matrix on SAD dataset.

4.8.1 Error Analysis

Though our model achieves a new state-of-the-art result on the SAD dataset in stress factors recognition, the F1 scores are still below 80%. We present the confusion matrix of one random test result of KC-Net on the SAD dataset in Figure 4.5.

The confusion matrix indicates that our model is able to correctly classify most of the stress factors, while the errors mainly come from the *Other* category. Both mis-classifications to and from *Other* severely affect model performance. We believe one cause is that *Other* contains a complex set with all unidentified factors, where some of the factors even cannot be determined by human annotators. Another reason could be that some of the factors in *Other* lie close to the listed factors in SAD. We also observe that the number of mis-classifications of each factor to *Other* is almost in reverse proportion to their correct classification numbers. As an example, with the fewest correctly classified samples, *Decision* has the highest number of mis-classifications to *Other*. We infer that the larger amount of correctly classified samples denote a more accurate extraction of the key class-specific features, which helps the model distinguish these factors from those of *Other*.

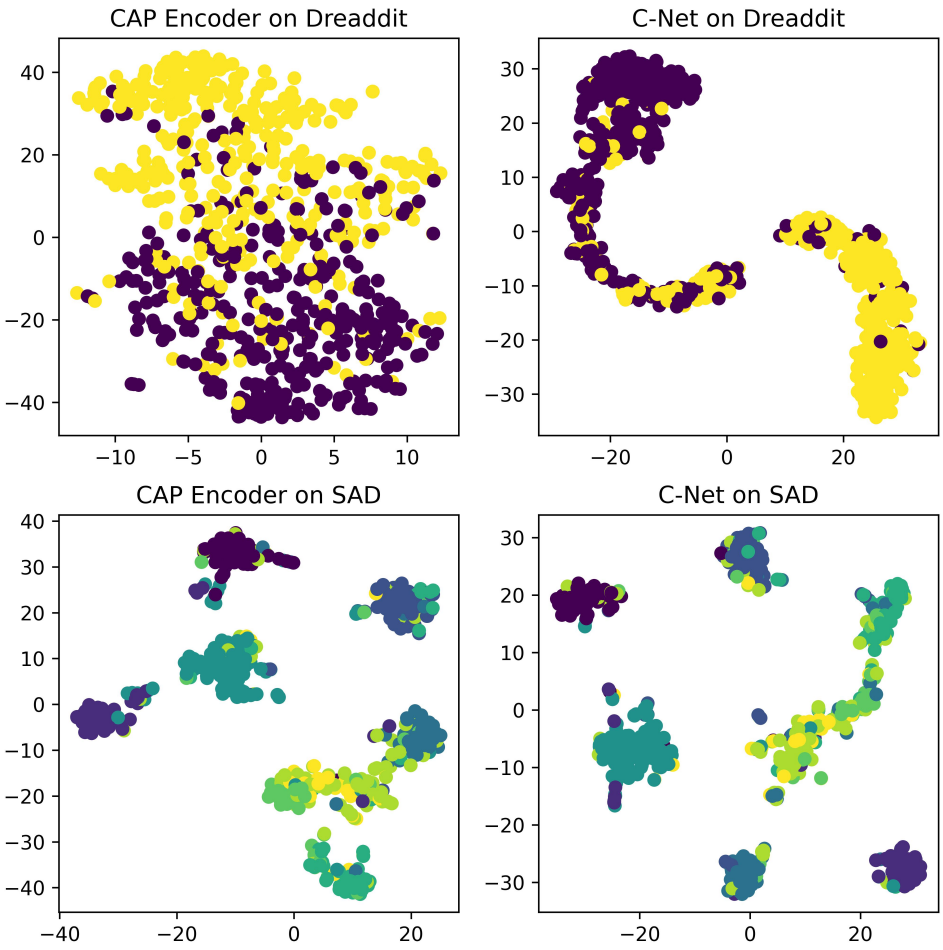


Figure 4.6. The UMAP visualization results of CAP encoder and C-Net on Dreaddit and SAD.

4.8.2 Qualitative Analysis of Contrastive Learning

Our experimental results demonstrate that supervised contrastive learning boosts the performance on all three datasets. We owe this performance to the mutually repelling process between embeddings with different labels, which enforces the model to focus on fine-grained semantics and key features. To further evaluate this inference, we utilise UMAP [195] to visualize the distribution of high-dimensional post representations obtained by training with and without supervised contrastive loss. To analyse the outcome of datasets with different categories numbers, we present the results on both Dreaddit and SAD, which are shown in Figure 4.6.

When contrastive loss is not used, the CAP encoder is trained solely on cross-entropy loss. The overlapping of both Dreaddit and SAD samples is relatively high, especially for factors with fewer samples in SAD, which increases the difficulty to the knowledge selection process of mentalisation and the learning of decision boundaries. When the CAP encoder is jointly trained with contrastive loss (C-Net), we can observe that the coupling of different classes has been distinctively enlarged, and samples of the same class gradually cohesive. On SAD, the effect shows more apparently with factors with more samples, while factors with fewer samples also have some degrees of decoupling with each other.

4.8.3 Case Study of Knowledge Infusion

We provide more insights on the effect of the mental state knowledge infusion and the mentalisation process by introducing two cases from the testing process of Dreaddit and Depression Mixed datasets, which are shown in Figure 4.7. Part of the post, the golden labels and predictions of CAP encoder, K-Net and KC-Net are listed. We also show the extracted mental state knowledge and the corresponding mentalisation attention scores. We provide key parts of the posts, the golden labels, and the predictions of different models on the post to show the effect of different modules directly. We utilised the final-layer hidden states of the COMET encoder as mental state knowledge, which is not convenient for the case study. Therefore, we leverage the COMET decoder to decode the hidden states and obtain the actual knowledge phrases, and record the corresponding knowledge-aware dot-product attention scores as evidence of which aspects the mentalisation process focuses on. For the case on Depression_Mixed, the CAP encoder failed to predict the post as depressed, since it was not aware of the negative mental states of the speaker. With the mental state knowledge, K-Net and KC-Net both correctly detected the depression by focusing on key knowledge aspects such as speaker reaction *Feel sad* and speaker intent *To be understood*. We also notice that compared with K-Net, KC-Net filtered out unrelated aspects such as effect on others *Gets asked to leave*, but pay more attention to implicit yet useful mental states: the speaker hopes to *Talk to someone*, which requires higher mentalisation ability to understand the relations between depression and loneliness. As a whole, in 476 test samples, K-Net corrects around 21 false negative samples by CAP encoder on average in Depression_Mixed with five random runs. We also

notice that these posts possess 20.7% less token numbers on average, which shows that part of the benefits of mental state knowledge comes from enriching the contexts.

For the case on Dreaddit, both the CAP encoder and K-Net fail to detect stress on the post. We notice that the attention scores of K-Net have high perplexity, which indicates that K-Net was not able to clearly distinguish important information from others. For KC-Net, we observe a much lower perplexity on attention scores. The model clearly focuses more on speaker reaction *Feel upset*, which directly reflects negative mental states, and effect on speaker *We are broke*. We believe this focus denotes that KC-Net has possessed some degree of mentalisation ability to recognize the stress factor *Financial Problem*, with the awareness that financial problems are more likely to be stress factors. To further analyse this hypothesis, we calculate the average information entropy[8] of the attention weights in the test set of Dreaddit for five random runs of both K-Net and KC-Net. The results show that K-Net has an average entropy of 1.53, while KC-Net has 1.34. KC-Net achieves over 12.4% decrease in entropy, which denotes that the model possesses higher confidence in selecting crucial knowledge aspects. Based on this idea, it would also be interesting to explicitly combine stress factors detection with mental health conditions detection in future work.

4.8.4 Ethical Considerations

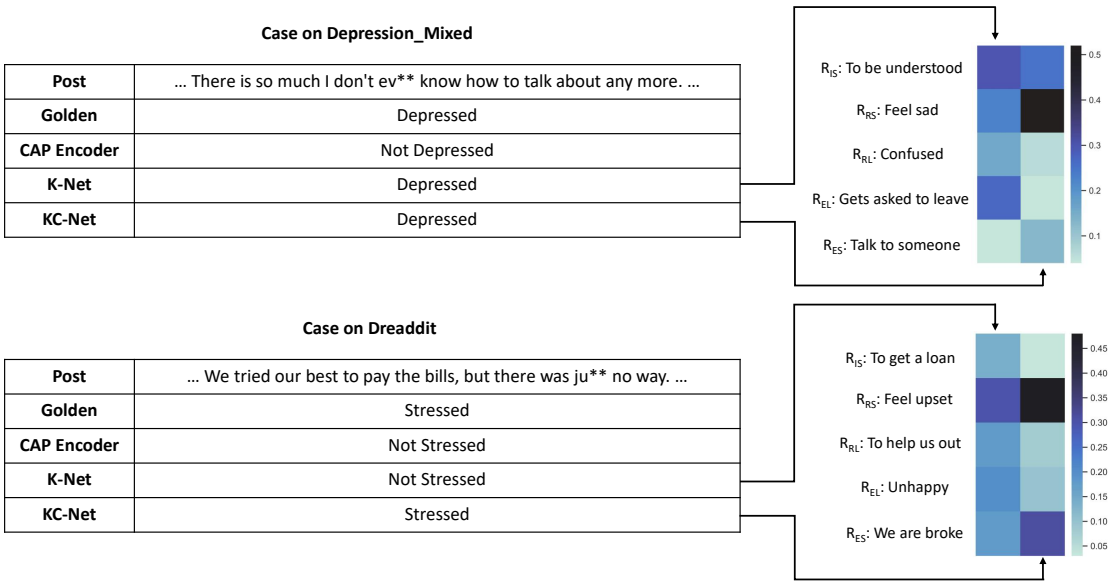


Figure 4.7. We provide two cases, each from Depression_Mixed or Dreaddit.

Our model aims to provide assistance to different stakeholders using social media as a source of information for the the early detection of stress and depression for non-clinical use (i.e. public health and policy makers, social care workers, etc. who work at the intersection of public health and social care and need to be updated on mental health issues related with specific topics from social media). The model predictions are not meant to be used as psychiatric diagnoses. One reason is that the datasets are either annotated in a weakly supervised manner or labelled by non-experts from Amazon Mechanical Turk within the predefined an-

[8] https://en.wikipedia.org/wiki/Entropy_(information_theory)

notation rules, which inevitably leads to annotation bias and can not verify an actual diagnosis [29]. The model can also make false predictions. Since most of the datasets in mental health involve sensitive privacy of the posters, we try to minimise the privacy impact when using the datasets and model. Researchers need to follow the strict protocols [37], [196], [197] by acquiring an exemption or ethical approval from their Institutional Review Board. Moreover, researchers need to obtain informed consent and protect sensitive data to avoid further psychological distress and intrusive treatment. All examples shown in our paper have been paraphrased and obfuscated according to the moderate disguise scheme suggested by Bruckman [197] to prevent misuse. In addition, we study the datasets in a purely observational capacity, with no intervention in user experience.

4.9 Summary

In this chapter, we propose KC-Net, a mental state knowledge-aware and contrastive network for early stress and depression detection on social media. KC-Net first introduces mental state knowledge from a generative knowledge base COMET, which explicitly models the mental state of speakers. Then GRU models and knowledge-aware dot-product attention are utilised for the mentalisation process, which aids the model in selecting more relevant knowledge aspects. We also use a supervised contrastive learning module to fully leverage label information for capturing class-specific features. It's also expected to better guide the knowledge selection process in mentalisation.

We test our method on three public datasets, which include a depression detection dataset, a stress detection dataset and a stress factors recognition dataset. The experiments show that our model achieves new state-of-the-art results on all three datasets. Further analysis determines the effectiveness of each module, their contributions to factor specific improvements, and the main causes of errors. We also provide visualizations and analyse cases to show the outcomes of each module intuitively. During the analysis, we notice that knowledge infusion works exceptionally well on low-resource categories, and the model shows evidence in automatically recognizing stress factors in the stress detection task. We will focus on these two observations in our following research.

Chapter 5

Conclusions

5.1 Contributions

In this thesis, we aim to contribute to the representation learning techniques for applications in emotion recognition and mental health analysis. We are especially interested in two techniques: contrastive learning and knowledge infusion. Supervised contrastive learning is employed to distinguish the representations of similar categories, and knowledge infusion is utilised to enrich the semantics and facilitate the reasoning process. We design novel architectures to adapt these techniques to the experiment tasks: emotion recognition in conversations and stress and depression detection. Comprehensive experiments and analyses are conducted for each task to prove the effectiveness of our methods and partially explain the inner mechanisms of the proposed modules.

Firstly, we design a new low-dimensional supervised cluster-level contrastive learning method for the ERC task. We reduce the high-dimensional supervised contrastive learning space to a three-dimensional space, Valance-Arousal-Dominance, and incorporate VAD prototypes from the emotion lexicon NRC-VAD by proposing the novel SCCL method. In addition, the pre-trained knowledge adapters are devised to infuse factual and linguistic knowledge into the PLM-based context-aware utterance encoder. Experimental results show that our method achieves new state-of-the-art results on three datasets IEMOCAP, MELD, and DailyDialog. The ablation study proves the effectiveness of each proposed module, and further analysis indicates that VAD space is an appropriate and interpretable space for SCCL. Emotion prototypes from NRC-VAD provide helpful quantitative information to guide SCCL, which improves model performance and stabilises the training process. The knowledge infused by pre-trained knowledge adapters also enhances the performance of the utterance encoder and SCCL.

Secondly, we propose KC-Net, a mental state knowledge-aware and contrastive network for early stress and depression detection on social media. KC-Net first introduces mental state knowledge from a generative knowledge base COMET, which explicitly models the mental state of speakers. Then GRU models and knowledge-aware dot-product attention are utilised for the mentalisation process, which aids the model in selecting more relevant knowledge aspects. We also use a supervised contrastive learning module to fully leverage label information for capturing class-specific features. It is also expected to guide the knowledge selection

process in mentalisation better. We test our method on three public datasets: a depression detection dataset, a stress detection dataset and a stress factors recognition dataset. The experiments show that our model achieves new state-of-the-art results on all three datasets. Further analysis determines the effectiveness of each module, their contributions to factor-specific improvements, and the leading causes of errors. We also provide visualisations and analyse cases to show each module's outcomes intuitively.

With the above outcomes, we can answer the research questions raised in Sec. 1.2. For research question #1, supervised contrastive learning can enhance the representations for ERC and stress and depression detection tasks since SCL pushes apart the representations with different labels, which forces the model to be aware of the fine-grained features indicating the differences and facilitates the discovery of the decision boundary. In addition, VAD information from sentiment lexicons enables stable clustering in low-dimensional contrast space, further improving SCL's performance.

For research question #2, task-related knowledge can enrich the representations and benefit the reasoning process for ERC and stress and depression detection task. In stress and depression detection, model performance and case studies show that the infusion of mental state knowledge enables the model to focus on critical parts of the speaker's mental state and find a clue to make a correct hypothesis. In ERC, factual knowledge incorporated from the knowledge adapter provides emotion-related relations and enriches the semantics of the utterance representation. Linguistic knowledge provides clear sentence structures on the utterances and helps to model the dialogue. Both knowledge types improve model performance in ERC, but linguistic knowledge benefits more significantly.

5.2 Limitations

Though our representation learning methods achieve impressive results in ERC and stress and depression detection tasks, several limitations remain. For ERC, the fuzzy emotions that vary in VAD levels under different scenarios are not handled with the unified emotion prototypes. This limitation directly affects the performance of our model on the dataset EmoryNLP labelled with fuzzy emotions. In addition, label imbalance problems still affect the model performance on low-resource emotion categories. The model lacks training samples to mine functional patterns on these emotions, which leads to low accuracy. Finally, the model performance improves less significantly on the short-context multi-party dataset MELD, which shows the importance of context information in multi-party ERC tasks. Though knowledge infusion enriches the semantics of each utterance, how to leverage external knowledge to enhance short-context scenarios remains unsolved.

For stress and depression detection, the research topic is more sensitive. The most crucial concern of the real-world applications is the ethics and the diagnosis standard. The training data can leak personal information and possess potential bias. Therefore, a fundamental solution is to increase the interpretability of the model, which facilitates human supervision

and provides more information for human diagnosis. However, a fundamental limitation of the deep learning-based method is the lack of interpretability, as the learnt high-dimensional distributed representations are hard to understand. Though we provide a case study and visualisation in our thesis, more efforts are required to develop more interpretable models. In addition, the error analysis shows that the proposed KC-Net also suffers from label imbalance in stress factors detection, which leads to bad performance in low-resource categories.

5.3 Future Work

Based on the above limitations, we propose several directions for future work. We will leverage more fine-grained supervision signals for ERC to handle fuzzy emotions. In our work, we only utilise the NRC-VAD emotion prototypes of the emotion labels, while the emotion prototypes of many phrases in the utterances are also provided. Incorporating these emotion prototypes is expected to provide fine-grained information to ERC. We will also develop more efficient methods to alleviate the label imbalance problem. For example, a more appropriate loss function can enforce the model to focus on the low-resource categories during training. In addition, we will explore more knowledge infusion methods to solve the lack of context problems. For example, new pre-training methods can infuse the contextualised commonsense knowledge in the CICERO [89] dataset.

For stress and depression detection, according to the case study, the model shows evidence of automatically recognising stress factors in the stress detection task. Therefore, one direction of future work in improving interpretability is to jointly make a diagnosis and mine the decision factors, such as the stress factors. The joint training not only enhances model performance in each task but also provides more evidence for the diagnosis. During the analysis, we also notice that the mental state knowledge infusion works exceptionally well on low-resource categories, which provides a future direction for alleviating the label imbalance problem. For example, we can design new knowledge infusion methods to focus more on the low-resource categories.

References

[1] P. Ekman, "Facial expression and emotion.," *American psychologist*, vol. 48, no. 4, p. 384, 1993.

[2] R. Plutchik, *A psychoevolutionary theory of emotions*, 1982.

[3] J. A. Russell and A. Mehrabian, "Evidence for a three-factor theory of emotions," *Journal of research in Personality*, vol. 11, no. 3, pp. 273–294, 1977.

[4] A. Mehrabian, "Framework for a comprehensive description and measurement of emotional states.," *Genetic, social, and general psychology monographs*, 1995.

[5] A. B. Warriner, V. Kuperman, and M. Brysbaert, "Norms of valence, arousal, and dominance for 13,915 english lemmas," *Behavior research methods*, vol. 45, no. 4, pp. 1191–1207, 2013.

[6] A. Saxena, A. Khanna, and D. Gupta, "Emotion recognition and detection methods: A comprehensive survey," *Journal of Artificial Intelligence and Systems*, vol. 2, no. 1, pp. 53–79, 2020.

[7] S. Poria, N. Majumder, R. Mihalcea, and E. Hovy, "Emotion recognition in conversation: Research challenges, datasets, and recent advances," *IEEE Access*, vol. 7, pp. 100 943–100 953, 2019. DOI: 10.1109/ACCESS.2019.2929050.

[8] Y. Ma, K. L. Nguyen, F. Z. Xing, and E. Cambria, "A survey on empathetic dialogue systems," *Information Fusion*, vol. 64, pp. 50–70, 2020.

[9] T. Saha and S. Ananiadou, "Emotion-aware and intent-controlled empathetic response generation using hierarchical transformer network," in *IJCNN*, IEEE, 2022, pp. 1–8.

[10] S. Zad, M. Heidari, J. H. J. Jones, and O. Uzuner, "Emotion detection of textual data: An interdisciplinary survey," in *AIIoT*, 2021, pp. 0255–0261. DOI: 10.1109/AIIoT52608.2021.9454192.

[11] J. Wang, J. Wang, C. Sun, S. Li, X. Liu, L. Si, M. Zhang, and G. Zhou, "Sentiment classification in customer service dialogue with topic-aware multi-task learning," in *AAAI*, vol. 34, AAAI Press, 2020, pp. 9177–9184.

[12] P. Nandwani and R. Verma, "A review on sentiment analysis and emotion detection from text," *Social Network Analysis and Mining*, vol. 11, no. 1, pp. 1–19, 2021.

[13] A. Chatterjee, U. Gupta, M. K. Chinnakotla, R. Srikanth, M. Galley, and P. Agrawal, "Understanding emotions in text using deep learning and big data," *Computers in Human Behavior*, vol. 93, pp. 309–317, 2019.

[14] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, "Iemocap: Interactive emotional dyadic motion capture database," *Language resources and evaluation*, vol. 42, no. 4, pp. 335–359, 2008.

[15] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, and R. Mihalcea, "MELD: A multimodal multi-party dataset for emotion recognition in conversations," in *ACL*, Association for Computational Linguistics, 2019, pp. 527–536. DOI: 10.18653/v1/p19-1050.

[16] S. M. Zahiri and J. D. Choi, "Emotion detection on TV show transcripts with sequence-based convolutional neural networks," in *AAAI Workshops*, ser. AAAI Technical Report, vol. WS-18, AAAI Press, 2018, pp. 44–52.

[17] Y. Li, H. Su, X. Shen, W. Li, Z. Cao, and S. Niu, "Dailydialog: A manually labelled multi-turn dialogue dataset," in *IJCNLP*, Asian Federation of Natural Language Processing, 2017, pp. 986–995.

[18] G. McKeown, M. Valstar, R. Cowie, M. Pantic, and M. Schroder, "The semaine database: Annotated multimodal records of emotionally colored conversations between a person and a limited agent," *IEEE transactions on affective computing*, vol. 3, no. 1, pp. 5–17, 2011.

[19] S.-Y. Chen, C.-C. Hsu, C.-C. Kuo, L.-W. Ku, *et al.*, "Emotionlines: An emotion corpus of multi-party conversations," *arXiv preprint arXiv:1802.08379*, 2018.

[20] A. Chatterjee, U. Gupta, M. K. Chinnakotla, R. Srikanth, M. Galley, and P. Agrawal, "Understanding emotions in text using deep learning and big data," *Computers in Human Behavior*, vol. 93, pp. 309–317, 2019.

[21] M. W. Morris and D. Keltner, "How emotions work: The social functions of emotional expression in negotiations," *Research in organizational behavior*, vol. 22, pp. 1–50, 2000.

[22] S. Evans-Lacko, S. Aguilar-Gaxiola, A. Al-Hamzawi, *et al.*, "Socio-economic variations in the mental health treatment gap for people with anxiety, mood, and substance use disorders: Results from the who world mental health (wmh) surveys," *Psychological medicine*, vol. 48, no. 9, pp. 1560–1571, 2018.

[23] I. Pirina and Ç. Çöltekin, "Identifying depression on reddit: The effect of training data," in *EMNLP Workshops*, Association for Computational Linguistics, 2018, pp. 9–12.

[24] A. T. Beck, *Cognitive therapy of depression*. Guilford press, 1979.

[25] T. Pyszczynski, K. Holt, and J. Greenberg, "Depression, self-focused attention, and expectancies for positive and negative future life events for self and others.," *Journal of personality and social psychology*, vol. 52, no. 5, p. 994, 1987.

[26] S. Rude, E.-M. Gortner, and J. Pennebaker, "Language use of depressed and depression-vulnerable college students," *Cognition & Emotion*, vol. 18, no. 8, pp. 1121–1133, 2004.

[27] N. Ramirez-Esparza, C. Chung, E. Kacewic, and J. Pennebaker, "The psychology of word use in depression forums in english and in spanish: Testing two text analytic approaches," in *AAAI Conference on Web and Social Media*, vol. 2, AAAI Press, 2008, pp. 102–108.

[28] G. Gkotsis, A. Oellrich, T. J. P. Hubbard, R. J. B. Dobson, M. Liakata, S. Velupillai, and R. Dutta, "The language of mental health problems in social media," in *NAACL Workshops*, K. Hollingshead and L. H. Ungar, Eds., The Association for Computational Linguistics, 2016, pp. 63–73. DOI: 10.18653/v1/w16-0307.

[29] T. Zhang, A. M. Schoene, S. Ji, and S. Ananiadou, "Natural language processing applied to mental illness detection: A narrative review," *npj Digital Medicine*, vol. 5, no. 1, pp. 1–13, 2022.

[30] A. Tsakalidis, M. Liakata, T. Damoulas, and A. I. Cristea, "Can we assess mental health through social media and smart devices? addressing bias in methodology and evaluation," in *ECML-PKDD*, Springer, 2018, pp. 407–423. DOI: 10.1007/978-3-030-10997-4_25.

[31] S. J. Lupien, B. S. McEwen, M. R. Gunnar, and C. Heim, "Effects of stress throughout the lifespan on the brain, behaviour and cognition," *Nature reviews neuroscience*, vol. 10, no. 6, pp. 434–445, 2009.

[32] M. A. Calcia, D. R. Bonsall, P. S. Bloomfield, S. Selvaraj, T. Barichello, and O. D. Howes, "Stress and neuroinflammation: A systematic review of the effects of stress on microglia and the implications for mental illness," *Psychopharmacology*, vol. 233, no. 9, pp. 1637–1650, 2016.

[33] D. V. Sheehan, "Depression: Underdiagnosed, undertreated, underappreciated.," *Managed care (Langhorne, Pa.)*, vol. 13, no. 6 Suppl Depression, pp. 6–8, 2004.

[34] C. S. Richards and M. W. O'Hara, *The Oxford handbook of depression and comorbidity*. Oxford University Press, 2014.

[35] A. G. Reece, A. J. Reagan, K. L. Lix, P. S. Dodds, C. M. Danforth, and E. J. Langer, "Forecasting the onset and course of mental illness with twitter data," *Scientific reports*, vol. 7, no. 1, pp. 1–11, 2017.

[36] T. W. Bickmore, H. Trinh, S. Olafsson, T. K. O'Leary, R. Asadi, N. M. Rickles, and R. Cruz, "Patient and consumer safety risks when using conversational assistants for medical information: An observational study of siri, alexa, and google assistant," *Journal of medical Internet research*, vol. 20, no. 9, e11510, 2018.

[37] A. Benton, G. Coppersmith, and M. Dredze, "Ethical research protocols for social media health research," in *ACL Workshops*, Association for Computational Linguistics, 2017, pp. 94–102.

[38] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *ICML*, PMLR, 2020, pp. 1597–1607.

[39] Y. Li, P. Hu, J. Z. Liu, D. Peng, J. T. Zhou, and X. Peng, "Contrastive clustering," in *AAAI*, AAAI Press, 2021, pp. 8547–8555.

[40] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 18 661–18 673, 2020.

[41] T. Gao, X. Yao, and D. Chen, "Simcse: Simple contrastive learning of sentence embeddings," in *EMNLP*, Association for Computational Linguistics, 2021, pp. 6894–6910.

[42] Y. Yan, R. Li, S. Wang, F. Zhang, W. Wu, and W. Xu, "Consert: A contrastive framework for self-supervised sentence representation transfer," in *ACL*, Association for Computational Linguistics, 2021, pp. 5065–5075.

[43] B. Gunel, J. Du, A. Conneau, *et al.*, "Supervised contrastive learning for pre-trained language model fine-tuning," in *ICLR*, OpenReview.net, 2021.

[44] Y. Xie, K. Yang, C. Sun, B. Liu, and Z. Ji, "Knowledge-interactive network with sentiment polarity intensity-aware multi-task learning for emotion recognition in conversations," in *Findings of EMNLP*, Association for Computational Linguistics, 2021, pp. 2879–2889. DOI: 10.18653/v1/2021.findings-emnlp.245.

[45] R. Speer, J. Chin, and C. Havasi, "Conceptnet 5.5: An open multilingual graph of general knowledge," in *AAAI*, AAAI Press, 2017, pp. 4444–4451.

[46] P. Singh *et al.*, "The public acquisition of commonsense knowledge," in *AAAI Spring Symposium*, AAAI Press, 2002.

[47] M. Sap, R. Le Bras, E. Allaway, *et al.*, "Atomic: An atlas of machine commonsense for if-then reasoning," in *AAAI*, vol. 33, AAAI Press, 2019, pp. 3027–3035.

[48] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI open*, vol. 1, pp. 57–81, 2020.

[49] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL*, Association for Computational Linguistics, 2019, pp. 4171–4186.

[50] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[51] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," *Advances in neural information processing systems*, vol. 32, 2019.

[52] R. Wang, D. Tang, N. Duan, Z. Wei, X. Huang, J. Ji, G. Cao, D. Jiang, and M. Zhou, "K-adapter: Infusing knowledge into pre-trained models with adapters," in *Findings of ACL*, Association for Computational Linguistics, 2021, pp. 1405–1418. DOI: 10.18653/v1/2021.findings-acl.121.

[53] S. M. Mohammad, "Obtaining reliable human ratings of valence, arousal, and dominance for 20, 000 english words," in *ACL*, Association for Computational Linguistics, 2018, pp. 174–184. DOI: 10.18653/v1/P18-1017.

[54] A. Bosselut, H. Rashkin, M. Sap, *et al.*, "COMET: Commonsense transformers for automatic knowledge graph construction," in *ACL*, Association for Computational Linguistics, 2019, pp. 4762–4779.

[55] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[56] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[57] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *EMNLP*, Association for Computational Linguistics, 2014, pp. 1532–1543.

[58] Y. Kim, "Convolutional neural networks for sentence classification," in *EMNLP*, Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1746–1751. DOI: 10.3115/v1/D14-1181.

[59] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[60] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *EMNLP*, Association for Computational Linguistics, 2014, pp. 1724–1734. DOI: 10.3115/v1/d14-1179.

[61] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[62] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[63] X. Bai, "Text classification based on lstm and attention," in *ICDIM*, IEEE, 2018, pp. 29–32.

[64] Y. Luan and S. Lin, "Research on text classification based on cnn and lstm," in *ICAICA*, IEEE, 2019, pp. 352–355.

[65] D. Pawade, A. Sakhapara, M. Jain, N. Jain, and K. Gada, "Story scrambler-automatic text generation using word level rnn-lstm," *International Journal of Information Technology and Computer Science (IJITCS)*, vol. 10, no. 6, pp. 44–53, 2018.

[66] J. Vasilakes, C. Zerva, M. Miwa, and S. Ananiadou, "Learning disentangled representations of negation and uncertainty," in *ACL*, Association for Computational Linguistics, May 2022, pp. 8380–8397. DOI: 10.18653/v1/2022.acl-long.574.

[67] C. Su, H. Huang, S. Shi, P. Jian, and X. Shi, "Neural machine translation with gumbel tree-lstm based encoder," *Journal of Visual Communication and Image Representation*, vol. 71, p. 102 811, 2020.

[68] H. Xu, Q. Liu, J. van Genabith, D. Xiong, and M. Zhang, "Multi-head highly parallelized lstm decoder for neural machine translation," in *ACL*, Association for Computational Linguistics, 2021, pp. 273–282.

[69] H.-w. An and N. Moon, "Design of recommendation system for tourist spot using sentiment analysis based on cnn-lstm," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–11, 2019.

[70] F. Kong, J. Li, and Z. Lv, "Construction of intelligent traffic information recommendation system based on long short-term memory," *Journal of computational science*, vol. 26, pp. 78–86, 2018.

[71] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017, pp. 5998–6008.

[72] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, IEEE Computer Society, 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[73] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, "Pre-trained models for natural language processing: A survey," *Science China Technological Sciences*, vol. 63, no. 10, pp. 1872–1897, 2020.

[74] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *NAACL*, Association for Computational Linguistics, 2018, pp. 2227–2237. DOI: 10.18653/v1/n18-1202.

[75] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, *et al.*, "Improving language understanding by generative pre-training," 2018.

[76] D. Hazarika, S. Poria, R. Zimmermann, and R. Mihalcea, "Conversational transfer learning for emotion recognition," *Information Fusion*, vol. 65, pp. 1–12, 2021.

[77] M. Kaya and H. Ş. Bilge, "Deep metric learning: A survey," *Symmetry*, vol. 11, no. 9, p. 1066, 2019.

[78] J. Li, P. Zhou, C. Xiong, and S. C. H. Hoi, "Prototypical contrastive learning of unsupervised representations," in *ICLR*, OpenReview.net, 2021.

[79] Y. Wang, J. Lin, Q. Cai, Y. Pan, T. Yao, H. Chao, and T. Mei, "A low rank promoting prior for unsupervised contrastive learning," *arXiv preprint arXiv:2108.02696*, 2021.

[80] J. Li, C. Xiong, and S. C. H. Hoi, "Comatch: Semi-supervised learning with contrastive graph regularization," in *ICCV*, IEEE, 2021, pp. 9455–9464. DOI: 10.1109/ICCV48922.2021.00934.

[81] R. Zhu, B. Zhao, J. Liu, Z. Sun, and C. W. Chen, "Improving contrastive learning by visualizing feature transformation," in *ICCV*, IEEE, 2021, pp. 10 286–10 295. DOI: 10.1109/ICCV48922.2021.01014.

[82] J. M. Giorgi, O. Nitski, B. Wang, and G. D. Bader, "Declutr: Deep contrastive learning for unsupervised textual representations," in *ACL*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds., Association for Computational Linguistics, 2021, pp. 879–895. DOI: 10.18653/v1/2021.acl-long.72.

[83] T. Kim, K. M. Yoo, and S.-g. Lee, "Self-guided contrastive learning for bert sentence representations," in *ACL*, Association for Computational Linguistics, 2021, pp. 2528–2540.

[84] S. Li, H. Yan, and X. Qiu, "Contrast and generation make BART a good dialogue emotion recognizer," in *AAAI*, AAAI Press, 2022, pp. 11 002–11 010.

[85] H. Alhuzali and S. Ananiadou, "Improving textual emotion recognition based on intra- and inter-class variation," *IEEE Transactions on Affective Computing*, pp. 1–1, 2021. DOI: 10.1109/TAFFC.2021.3104720.

[86] H. P. Grice, "Logic and conversation," in *Speech acts*, Brill, 1975, pp. 41–58.

[87] Y. Xie and P. Pu, "How commonsense knowledge helps with natural language tasks: A survey of recent resources and methodologies," *arXiv preprint arXiv:2108.04674*, 2021.

[88] N. Tandon, G. De Melo, and G. Weikum, "Webchild 2.0: Fine-grained commonsense knowledge distillation," in *ACL Demo*, Association for Computational Linguistics, 2017, pp. 115–120.

[89] D. Ghosal, S. Shen, N. Majumder, R. Mihalcea, and S. Poria, "CICERO: A dataset for contextualized commonsense inference in dialogues," in *ACL*, Association for Computational Linguistics, May 2022, pp. 5010–5028. DOI: 10.18653/v1/2022.acl-long.344.

[90] F. Ilievski, A. Oltramari, K. Ma, B. Zhang, D. L. McGuinness, and P. Szekely, "Dimensions of commonsense knowledge," *Knowledge-Based Systems*, vol. 229, p. 107 347, 2021.

[91] F. Bond and R. Foster, "Linking and extending an open multilingual wordnet," in *ACL*, Association for Computational Linguistics, 2013, pp. 1352–1362.

[92] J. Breen, "Jmdict: A japanese-multilingual dictionary," in *Workshop on multilingual linguistic resources*, 2004, pp. 65–72.

[93] D. Lenat and R. Guha, "Building large knowledge-based systems: Representation and inference in the cyc project," *Artificial Intelligence*, vol. 61, no. 1, p. 4152, 1993.

[94] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "Dbpedia: A nucleus for a web of open data," in *The semantic web*, Springer, 2007, pp. 722–735.

[95] E. Cambria, R. Speer, C. Havasi, and A. Hussain, "Senticnet: A publicly available semantic resource for opinion mining," in *AAAI Fall Symposium*, ser. AAAI Technical Report, vol. FS-10-02, AAAI Press, 2010.

[96] A. Esuli and F. Sebastiani, "SENTIWORDNET: A publicly available lexical resource for opinion mining," in *LREC*, European Language Resources Association (ELRA), May 2006.

[97] R. Plutchik, "The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice," *American scientist*, vol. 89, no. 4, pp. 344–350, 2001.

[98] D. Chen and C. Manning, "A fast and accurate dependency parser using neural networks," in *EMNLP*, Association for Computational Linguistics, Oct. 2014, pp. 740–750. DOI: 10.3115/v1/D14-1082.

[99] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 12, pp. 2724–2743, 2017.

[100] S. Zhang, H. Tong, J. Xu, and R. Maciejewski, "Graph convolutional networks: A comprehensive review," *Computational Social Networks*, vol. 6, no. 1, pp. 1–23, 2019.

[101] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *ICLR*, OpenReview.net, 2018.

[102] Z. Zhang, P. Cui, and W. Zhu, "Deep learning on graphs: A survey," *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[103] D. Ghosal, N. Majumder, A. F. Gelbukh, R. Mihalcea, and S. Poria, "COSMIC: commonsense knowledge for emotion identification in conversations," in *Findings of EMNLP*, Association for Computational Linguistics, 2020, pp. 2470–2481. DOI: 10.18653/v1/2020.findings-emnlp.224.

[104] K. Yang, T. Zhang, and S. Ananiadou, "A mental state knowledge–aware and contrastive network for early stress and depression detection on social media," *Information Processing & Management*, vol. 59, no. 4, p. 102 961, 2022.

[105] J. Li, Z. Lin, P. Fu, and W. Wang, "Past, present, and future: Conversational emotion recognition through structural modeling of psychological knowledge," in *Findings of EMNLP*, Association for Computational Linguistics, 2021, pp. 1204–1214.

[106] Y. Sun, S. Wang, Y. Li, S. Feng, X. Chen, H. Zhang, X. Tian, D. Zhu, H. Tian, and H. Wu, "Ernie: Enhanced representation through knowledge integration," *arXiv preprint arXiv:1904.09223*, 2019.

[107] P. Ke, H. Ji, S. Liu, X. Zhu, and M. Huang, "Sentilare: Sentiment-aware language representation learning with linguistic knowledge," in *EMNLP*, Association for Computational Linguistics, 2020, pp. 6975–6988.

[108] A. Lauscher, I. Vulic, E. M. Ponti, A. Korhonen, and G. Glavas, "Specializing unsupervised pretraining models for word-level semantic similarity," in *COLING*, International Committee on Computational Linguistics, 2020, pp. 1371–1383. DOI: 10.18653/v1/2020.coling-main.118.

[109] X. Chen, N. Zhang, X. Xie, S. Deng, Y. Yao, C. Tan, F. Huang, L. Si, and H. Chen, "Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction," in *WWW*, ACM, 2022, pp. 2778–2788. DOI: 10.1145/3485447.3511998.

[110] D. Hazarika, S. Poria, R. Mihalcea, E. Cambria, and R. Zimmermann, "Icon: Interactive conversational memory network for multimodal emotion detection," in *EMNLP*, Association for Computational Linguistics, 2018, pp. 2594–2604.

[111] D. Hazarika, S. Poria, A. Zadeh, E. Cambria, L.-P. Morency, and R. Zimmermann, "Conversational memory network for emotion recognition in dyadic dialogue videos," in *NAACL*, Association for Computational Linguistics, 2018, pp. 2122–2132.

[112] N. Majumder, S. Poria, D. Hazarika, R. Mihalcea, A. F. Gelbukh, and E. Cambria, "Dialoguernn: An attentive RNN for emotion detection in conversations," in *AAAI*, AAAI Press, 2019, pp. 6818–6825. DOI: 10.1609/aaai.v33i01.33016818.

[113] E. Chapuis, P. Colombo, M. Manica, M. Labeau, and C. Clavel, "Hierarchical pretraining for sequence labelling in spoken dialog," in *Findings of EMNLP*, Association for Computational Linguistics, 2020, pp. 2636–2648.

[114] P. Zhong, D. Wang, and C. Miao, "Knowledge-enriched transformer for emotion detection in textual conversations," in *EMNLP*, Association for Computational Linguistics, 2019, pp. 165–176. DOI: 10.18653/v1/D19-1016.

[115] D. Zhang, X. Chen, S. Xu, and B. Xu, "Knowledge aware emotion recognition in textual conversations via multi-task incremental transformer," in *COLING*, International Committee on Computational Linguistics, 2020, pp. 4429–4440. DOI: 10.18653/v1/2020.coling-main.392.

[116] L. Zhu, G. Pergola, L. Gui, D. Zhou, and Y. He, "Topic-driven and knowledge-aware transformer for dialogue emotion detection," in *ACL*, Association for Computational Linguistics, 2021, pp. 1571–1582.

[117] T. Ishiwatari, Y. Yasuda, T. Miyazaki, and J. Goto, "Relation-aware graph attention networks with relational position encodings for emotion recognition in conversations," in *EMNLP*, Association for Computational Linguistics, 2020, pp. 7360–7370.

[118] W. Shen, S. Wu, Y. Yang, and X. Quan, "Directed acyclic graph network for conversational emotion recognition," in *ACL*, Association for Computational Linguistics, 2021, pp. 1551–1560.

[119] T. Kim and P. Vossen, "Emoberta: Speaker-aware emotion recognition in conversation with roberta," *arXiv preprint arXiv:2108.12009*, 2021.

[120] W. Shen, J. Chen, X. Quan, and Z. Xie, "Dialogxl: All-in-one xlnet for multi-party conversation emotion recognition," in *AAAI*, vol. 35, AAAI Press, 2021, pp. 13 789–13 797.

[121] W. Shen, S. Wu, Y. Yang, and X. Quan, "Directed acyclic graph network for conversational emotion recognition," in *ACL*, Association for Computational Linguistics, 2021, pp. 1551–1560.

[122] J. Li, D. Ji, F. Li, M. Zhang, and Y. Liu, "Hitrans: A transformer-based context- and speaker-sensitive model for emotion detection in conversations," in *COLING*, International Committee on Computational Linguistics, 2020, pp. 4190–4200. DOI: 10.18653/v1/2020.coling-main.370.

[123] D. Ghosal, N. Majumder, S. Poria, N. Chhaya, and A. Gelbukh, "Dialoguegcn: A graph convolutional neural network for emotion recognition in conversation," in *EMNLP*, Association for Computational Linguistics, 2019, pp. 154–164.

[124] P. Zhong, D. Wang, and C. Miao, "Knowledge-enriched transformer for emotion detection in textual conversations," in *EMNLP*, Association for Computational Linguistics, 2019, pp. 165–176.

[125] J. Wang, J. Wang, C. Sun, S. Li, X. Liu, L. Si, M. Zhang, and G. Zhou, "Sentiment classification in customer service dialogue with topic-aware multi-task learning," in *AAAI*, vol. 34, AAAI Press, 2020, pp. 9177–9184.

[126] S. Buechel and U. Hahn, "Emotion analysis as a regression problem - dimensional models and their implications on emotion representation and metrical evaluation," in *ECAI*, IOS Press, 2016, pp. 1114–1122. DOI: 10.3233/978-1-61499-672-9-1114.

[127] S. Park, J. Kim, S. Ye, J. Jeon, H. Park, and A. Oh, "Dimensional emotion detection from categorical emotion," in *EMNLP*, Association for Computational Linguistics, 2021, pp. 4367–4380. DOI: 10.18653/v1/2021.emnlp-main.358.

[128] R. Mukherjee, A. Naik, S. Poddar, S. Dasgupta, and N. Ganguly, "Understanding the role of affect dimensions in detecting emotions from tweets: A multi-task approach," in *SIGIR*, ACM, 2021, pp. 2303–2307. DOI: 10.1145/3404835.3463080.

[129] E. A. Ríssola, D. E. Losada, and F. Crestani, "A survey of computational methods for online mental state assessment on social media," *ACM Transactions on Computing for Healthcare*, vol. 2, no. 2, pp. 1–31, 2021.

[130] S. Chancellor and M. De Choudhury, "Methods in predictive techniques for mental health status on social media: A critical review," *NPJ digital medicine*, vol. 3, no. 1, pp. 1–11, 2020.

[131] G. Castillo-Sánchez, G. Marques, E. Dorronzoro, *et al.*, "Suicide risk assessment using machine learning and social networks: A scoping review," *Journal of medical systems*, vol. 44, no. 12, pp. 1–15, 2020.

[132] A. Trifan and J. L. Oliveira, "Bioinfo@ uavr at erisk 2019: Delving into social media texts for the early detection of mental and food disorders.," in *CLEF (Working Notes)*, 2019.

[133] M. E. Aragón, A. P. López-Monroy, L. C. González-Gurrola, and M. Montes-y-Gómez, "Detecting depression in social media using fine-grained emotions," in *NAACL*, Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 1481–1486. DOI: 10.18653/v1/N19-1151.

[134] M. Yoo, S. Lee, and T. Ha, "Semantic network analysis for understanding user experiences of bipolar and depressive disorders on reddit," *Information Processing & Management*, vol. 56, no. 4, pp. 1565–1575, 2019.

[135] S. Saleem, M. Pacula, R. Chasin, R. Kumar, R. Prasad, M. Crystal, B. Marx, D. Sloan, J. Vasterling, and T. Speroff, "Automatic detection of psychological distress indicators in online forum posts," in *APSIPA*, IEEE, 2012, pp. 1–4.

[136] R. Sawhney, P. Manchanda, R. Singh, and S. Aggarwal, "A computational approach to feature extraction for identification of suicidal ideation in tweets," in *ACL Workshops*, Association for Computational Linguistics, 2018, pp. 91–98. DOI: 10.18653/v1/P18-3013.

[137] M. Hiraga, "Predicting depression for japanese blog text," in *ACL Workshops*, Association for Computational Linguistics, 2017, pp. 107–113. DOI: 10.18653/v1/P17-3018.

[138] A. Fine, P. Crutchley, J. Blase, J. Carroll, and G. Coppersmith, "Assessing population-level symptoms of anxiety, depression, and suicide risk in real time using NLP applied to social media data," in *EMNLP Workshops*, Online: Association for Computational Linguistics, Nov. 2020, pp. 50–54. DOI: 10.18653/v1/2020.nlpcss-1.6.

[139] M. Trotzek, S. Koitka, and C. M. Friedrich, "Utilizing neural networks and linguistic metadata for early detection of depression indications in text sequences," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 3, pp. 588–601, 2018.

[140] S. Ghosh and T. Anwar, "Depression intensity estimation via social media: A deep learning approach," *IEEE Transactions on Computational Social Systems*, 2021.

[141] I. Sekulic and M. Strube, "Adapting deep learning methods for mental health prediction on social media," in *EMNLP Workshops*, Association for Computational Linguistics, 2019, pp. 322–327. DOI: 10.18653/v1/D19-5542.

[142] M. M. Tadesse, H. Lin, B. Xu, *et al.*, "Detection of suicide ideation in social media forums using deep learning," *Algorithms*, vol. 13, no. 1, p. 7, 2020.

[143] S. Zhou, Y. Zhao, J. Bian, *et al.*, "Exploring eating disorder topics on twitter: Machine learning approach," *JMIR Medical Informatics*, vol. 8, no. 10, e18273, 2020.

[144] X. Yao, G. Yu, J. Tang, *et al.*, "Extracting depressive symptoms and their associations from an online depression community," *Computers in human behavior*, vol. 120, p. 106 734, 2021.

[145] N. Wang, L. Fan, Y. Shivtare, V. Badal, K. Subbalakshmi, R. Chandramouli, and E. Lee, "Learning models for suicide prediction from social media posts," in *Workshop on Computational Linguistics and Clinical Psychology*, Online: Association for Computational Linguistics, 2021, pp. 87–92.

[146] Y. Wang, Z. Wang, C. Li, *et al.*, "A multitask deep learning approach for user depression detection on sina weibo," *arXiv preprint arXiv:2008.11708*, 2020.

[147] T. Yang, F. Li, D. Ji, X. Liang, T. Xie, S. Tian, B. Li, and P. Liang, "Fine-grained depression analysis based on chinese micro-blog reviews," *Information Processing & Management*, vol. 58, no. 6, p. 102 681, 2021.

[148] F. Haque, R. U. Nur, S. Al Jahan, *et al.*, "A transformer based approach to detect suicidal ideation using pre-trained language models," in *ICCIT*, IEEE, 2020, pp. 1–5.

[149] Z. Jiang, S. I. Levitan, J. Zomick, and J. Hirschberg, "Detection of mental health from reddit via deep contextualized representations," in *EMNLP Workshops*, Association for Computational Linguistics, 2020, pp. 147–156. DOI: 10.18653/v1/2020.louhi-1.16.

[150] A. Murarka, B. Radhakrishnan, and S. Ravichandran, "Detection and classification of mental illnesses on social media using roberta," *arXiv preprint arXiv:2011.11226*, 2020.

[151] P. Abed-Esfahani, D. Howard, M. Maslej, S. Patel, V. Mann, S. Goegan, and L. French, "Transfer learning for depression: Early detection and severity prediction from social media postings.," in *CLEF (Working Notes)*, 2019.

[152] E. Turcan, S. Muresan, and K. McKeown, "Emotion-infused models for explainable psychological stress detection," in *EMNLP*, Association for Computational Linguistics, 2021, pp. 2895–2909.

[153] R. Sawhney, H. Joshi, S. Gandhi, *et al.*, "A time-aware transformer based model for suicide ideation detection on social media," in *EMNLP*, Association for Computational Linguistics, 2020, pp. 7685–7697.

[154] R. Sawhney, H. Joshi, L. Flek, and R. Shah, "Phase: Learning emotional phase-aware representations for suicide ideation detection on social media," in *EACL*, Association for Computational Linguistics, 2021, pp. 2415–2428.

[155] S. Ghosh, A. Ekbal, and P. Bhattacharyya, "A multitask framework to detect depression, sentiment and multi-label emotion from suicide notes," *Cognitive Computation*, vol. 14, no. 1, pp. 110–129, 2022.

[156] K. Harrigian, C. Aguirre, and M. Dredze, "Do models of mental health based on social media data generalize?" In *Findings of EMNLP*, Online: Association for Computational Linguistics, Nov. 2020, pp. 3774–3788. DOI: 10.18653/v1/2020.findings-emnlp.337. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.337.

[157] S. Ji, T. Zhang, L. Ansari, *et al.*, "Mentalbert: Publicly available pretrained language models for mental healthcare," *arXiv preprint arXiv:2110.15621*, 2021.

[158] C. Aguirre and M. Dredze, "Qualitative analysis of depression models by demographics," in *NAACL Workshops*, Online: Association for Computational Linguistics, Jun. 2021, pp. 169–180. DOI: 10.18653/v1/2021.clpsych-1.19.

[159] B. Li, H. Zhou, J. He, M. Wang, Y. Yang, and L. Li, "On the sentence embeddings from pre-trained language models," in *EMNLP*, Association for Computational Linguistics, 2020, pp. 9119–9130.

[160] N. Alswaidan and M. E. B. Menai, "A survey of state-of-the-art approaches for emotion recognition in text," *Knowledge and Information Systems*, vol. 62, no. 8, pp. 2937–2987, 2020.

[161] G. E. Hinton and S. Roweis, "Stochastic neighbor embedding," in *Advances in Neural Information Processing Systems*, S. Becker, S. Thrun, and K. Obermayer, Eds., vol. 15, MIT Press, 2003.

[162] N. F. Liu, M. Gardner, Y. Belinkov, M. E. Peters, and N. A. Smith, "Linguistic knowledge and transferability of contextual representations," in *NAACL*, Association for Computational Linguistics, 2019, pp. 1073–1094. DOI: 10.18653/v1/n19-1112.

[163] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[164] H. ElSahar, P. Vougiouklis, A. Remaci, C. Gravier, J. S. Hare, F. Laforest, and E. Simperl, "T-rex: A large scale alignment of natural language with knowledge base triples," in *LREC*, European Language Resources Association (ELRA), 2018.

[165] Y. Bao, Q. Ma, L. Wei, W. Zhou, and S. Hu, "Speaker-guided encoder-decoder framework for emotion recognition in conversation," in *IJCAI*, L. D. Raedt, Ed., ijcai.org, 2022, pp. 4051–4057. DOI: 10.24963/ijcai.2022/562.

[166] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *ACL*, Association for Computational Linguistics, 2020, pp. 7871–7880.

[167] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *ICLR*, OpenReview.net, 2019.

[168] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He, "Accurate, large minibatch sgd: Training imagenet in 1 hour," *arXiv preprint arXiv:1706.02677*, 2017.

[169] P. Micikevicius, S. Narang, J. Alben, G. F. Diamos, E. Elsen, D. García, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh, and H. Wu, "Mixed precision training," in *ICLR*, OpenReview.net, 2018.

[170] M. E. Peters, M. Neumann, M. Iyyer, *et al.*, "Deep contextualized word representations," in *NAACL*, Association for Computational Linguistics, 2018, pp. 2227–2237.

[171] J. M. Zich, C. C. Attkisson, and T. K. Greenfield, "Screening for depression in primary care clinics: The ces-d and the bdi," *The International Journal of Psychiatry in Medicine*, vol. 20, no. 3, pp. 259–277, 1990.

[172] G. Vilagut, C. G. Forero, G. Barbaglia, and J. Alonso, "Screening for depression in the general population with the center for epidemiologic studies depression (ces-d): A systematic review with meta-analysis," *PloS one*, vol. 11, no. 5, e0155431, 2016.

[173] L. S. Radloff, "The ces-d scale: A self-report depression scale for research in the general population," *Applied psychological measurement*, vol. 1, no. 3, pp. 385–401, 1977.

[174] X. Yang, R. McEwen, L. R. Ong, and M. Zihayat, "A big data analytics framework for detecting user-level depression from social networks," *International Journal of Information Management*, vol. 54, p. 102 141, 2020.

[175] T. Charman and Y. Shmueli-Goetz, "The relationship between theory of mind, language and narrative discourse: An experimental study.," *Cahiers de Psychologie Cognitive/Current Psychology of Cognition*, 1998.

[176] M. L. Mumper and R. J. Gerrig, "Leisure reading and social cognition: A meta-analysis.," *Psychology of Aesthetics, Creativity, and the Arts*, vol. 11, no. 1, p. 109, 2017.

[177] C. Sun, X. Qiu, Y. Xu, *et al.*, "How to fine-tune bert for text classification?" In *China National Conference on Chinese Computational Linguistics*, Springer, 2019, pp. 194–206.

[178] J. Li, T. Tang, W. X. Zhao, and J.-R. Wen, "Pretrained language model for text generation: A survey," in *IJCAI*, ijcai.org, 2021, pp. 4492–4499. DOI: 10.24963/ijcai.2021/612.

[179] I. Beltagy, K. Lo, and A. Cohan, "SciBERT: A pretrained language model for scientific text," in *EMNLP*, Association for Computational Linguistics, 2019, pp. 3615–3620.

[180] K. Huang, J. Altosaar, and R. Ranganath, "Clinicalbert: Modeling clinical notes and predicting hospital readmission," *arXiv preprint arXiv:1904.05342*, 2019.

[181] K. Cho, B. van Merriënboer, C. Gulcehre, *et al.*, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *EMNLP*, Association for Computational Linguistics, 2014, pp. 1724–1734.

[182] A. J. Werntz, S. A. Steinman, J. J. Glenn, *et al.*, "Characterizing implicit mental health associations across clinical domains," *Journal of behavior therapy and experimental psychiatry*, vol. 52, pp. 17–28, 2016.

[183] E.-M. Seidel, U. Habel, A. Finkelmeyer, *et al.*, "Implicit and explicit behavioral tendencies in male and female depression," *Psychiatry research*, vol. 177, no. 1-2, pp. 124–130, 2010.

[184] F. Villarroel Ordenes, S. Ludwig, K. De Ruyter, *et al.*, "Unveiling what is written in the stars: Analyzing explicit, implicit, and discourse patterns of sentiment in social media," *Journal of Consumer Research*, vol. 43, no. 6, pp. 875–894, 2017.

[185] E. Turcan and K. McKeown, "Dreaddit: A Reddit dataset for stress analysis in social media," in *LOUHI Workshops*, Hong Kong: Association for Computational Linguistics, 2019, pp. 97–107.

[186] M. L. Mauriello, T. Lincoln, G. Hon, D. Simon, D. Jurafsky, and P. Paredes, "SAD: A stress annotated dataset for recognizing everyday stressors in sms-like conversational systems," in *CHI*, ACM, 2021, 399:1–399:7. DOI: 10.1145/3411763.3451799.

[187] P. A. Noone, "The Holmes–Rahe Stress Inventory," *Occupational Medicine*, vol. 67, no. 7, pp. 581–582, Oct. 2017, ISSN: 0962-7480. DOI: 10.1093/occmed/kqx099. eprint: https://academic.oup.com/occmed/article-pdf/67/7/581/20894844/kqx099.pdf. [Online]. Available: https://doi.org/10.1093/occmed/kqx099.

[188] A. H. Orabi, P. Buddhitha, M. H. Orabi, and D. Inkpen, "Deep learning for depression detection of twitter users," in *NAACL Workshops*, Association for Computational Linguistics, 2018, pp. 88–97. DOI: 10.18653/v1/w18-0609.

[189] F. Sadeque, D. Xu, and S. Bethard, "Uarizona at the clef erisk 2017 pilot task: Linear and recurrent models for early depression detection," in *CEUR workshops*, NIH Public Access, vol. 1866, 2017.

[190] L. Ren, H. Lin, B. Xu, S. Zhang, L. Yang, S. Sun, *et al.*, "Depression detection on reddit with an emotion-based attention network: Algorithm development and validation," *JMIR Medical Informatics*, vol. 9, no. 7, e28754, 2021.

[191] M. M. Tadesse, H. Lin, B. Xu, *et al.*, "Detection of depression-related posts in reddit social media forum," *IEEE Access*, vol. 7, pp. 44 883–44 893, 2019.

[192] D. Demszky, D. Movshovitz-Attias, J. Ko, *et al.*, "GoEmotions: A dataset of fine-grained emotions," in *ACL*, Online: Association for Computational Linguistics, 2020, pp. 4040–4054.

[193] T. Wolf, J. Chaumond, L. Debut, *et al.*, "Transformers: State-of-the-art natural language processing," in *EMNLP Demos*, Association for Computational Linguistics, 2020, pp. 38–45.

[194] I. Loshchilov and F. Hutter, "Fixing weight decay regularization in adam," *ArXiv*, vol. abs/1711.05101, 2017.

[195] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.

[196] J. Nicholas, S. Onie, and M. E. Larsen, "Ethics and privacy in social media research for mental health," *Current Psychiatry Reports*, vol. 22, no. 12, pp. 1–7, 2020.

[197] A. Bruckman, "Studying the amateur artist: A perspective on disguising data collected in human subjects research on the internet," *Ethics and Information Technology*, vol. 4, no. 3, pp. 217–231, 2002.