

NEURAL NETWORKS FOR TEXTUAL EMOTION RECOGNITION AND ANALYSIS

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

2022

Student id: 10448591

Department of Computer Science

Contents

Abstract	10
Declaration	12
Copyright	13
Acknowledgements	15
List of Abbreviations	17
1 Introduction	18
1.1 Motivation	18
1.2 Research Questions, Hypotheses and Objectives	23
1.3 Contributions and Publications	26
1.4 Thesis Structure	28
2 Technical Background	31
2.1 Neural Networks	31
2.2 Neural Network Training	33
2.2.1 Classification	33
2.2.2 Loss Function	34
2.2.3 Learning	34
2.2.4 Regularisation	36
2.3 Neural Network Approaches	38
2.3.1 Word Representations	38
2.3.2 Convolutional Neural Networks (CNN)	39
2.3.3 Recurrent Neural Networks (RNN)	41
2.3.4 Attention Mechanisms	43

2.3.5	Pre-training of Deep Bidirectional Transformers for Language Understanding (BERT)	45
2.3.6	Fine-tuning Methods	47
2.3.7	Deep Metric Learning	48
2.4	Summary	50
3	Emotion Recognition: Background	51
3.1	Problem Definitions	52
3.1.1	Task	52
3.1.2	Word-Emotion Association	52
3.1.3	Emotion Correlation	53
3.1.4	Emotion Labels	54
3.2	Related Terms	55
3.2.1	Emotion and Subjectivity	55
3.2.2	Emotion and Sentiment	56
3.2.3	Emotion and Affect	58
3.3	How Emotion is Reflected in Text?	58
3.4	Models of Emotion	60
3.4.1	Categorical Model	60
3.4.2	Dimensional Model	62
3.5	Datasets & corpora	62
3.6	Approaches to Textual Emotion Recognition	67
3.6.1	Lexicon-based	67
3.6.2	Supervised Learning	69
3.6.3	Unsupervised Learning	73
3.6.4	Transfer Learning	75
3.7	Evaluation Metrics	77
3.8	Summary and Limitations	79
4	Emotion Correlations and Associations	81
4.1	Motivation	82
4.2	Methodology	84
4.2.1	Framework	84
4.2.2	Our Method (SpanEmo)	85
4.2.3	Label-Correlation Aware (LCA) Loss	86
4.2.4	Training Objective	86

4.3	Experiments	87
4.3.1	Implementation Details	87
4.3.2	Data Set and Task Settings	87
4.4	Results	89
4.4.1	Ablation Study	90
4.5	Analysis	91
4.5.1	Prediction of Multiple Emotions	91
4.5.2	Learning Emotion-specific Associations	92
4.5.2.1	Word-Level	92
4.5.2.2	Tweet-Level	93
4.5.3	Qualitative Analysis	94
4.5.4	Label Correlations	96
4.5.5	Influence of Parameter (α)	97
4.6	Summary	98
5	Case Studies	100
5.1	Use Case on Adverse Drug Reaction	101
5.1.1	Introduction	101
5.1.2	Related Work	103
5.1.3	Data	104
5.1.4	Experiment (1)	106
5.1.4.1	Proposed Approach	106
5.1.4.2	Results	108
5.1.4.3	Analysis	110
5.1.5	Experiment (2)	114
5.1.5.1	Proposed Approach	114
5.1.5.2	Results and Analysis	115
5.2	Use Case on Mental Health	118
5.2.1	Introduction	118
5.2.2	Related Work	119
5.2.3	Experiments	120
5.2.3.1	Proposed Approach	122
5.2.3.2	Results and Analysis	125
5.2.3.3	Negative Results	129
5.3	Summary	130

6	Intra- and Inter-Class Variations	132
6.1	Motivation	133
6.2	Methodology	135
6.2.1	Triplet Centre Loss	135
6.2.2	Variant Triplet Centre Loss	135
6.2.3	Training Objective	136
6.3	Experiments	137
6.3.1	Implementation Details	138
6.3.2	Datasets and Task Settings	138
6.4	Evaluation	139
6.4.1	Results	139
6.4.1.1	Relevant Work	140
6.4.1.2	Contextualised Embeddings	140
6.4.1.3	Our Method (CEL+VTCL)	142
6.4.2	Ablation Study	142
6.4.3	Intra- and inter-class evaluation	143
6.5	Analysis	145
6.5.1	Model Predictions	145
6.5.2	Visualisation of Learned Representations	147
6.5.3	Qualitative analysis	148
6.5.4	Selecting the number of negative centres	149
6.6	Summary	150
7	Conclusion	152
7.1	Contributions	153
7.2	Limitations and Future Work	157
7.2.1	Task Limitations	157
7.2.2	TER Models	159
7.2.3	Future Work	162
	Bibliography	163

Word Count: 42321

List of Tables

3.1	Available Emotion Recognition Datasets.	64
3.2	Confusion matrix of binary classification.	77
3.3	Confusion matrix of multi-class classification.	78
4.1	Examples Tweets for multi-label emotion classification	82
4.2	Hyper-parameter values.	87
4.3	Data Statistics	88
4.4	The results of multi-label emotion classification on SemEval-2018 test set.	89
4.5	Ablation experiment results.	91
4.6	Presenting the number of co-existing emotion classes.	92
4.7	Top 10 words associated with each corresponding emotion.	93
4.8	Prediction of emotion classes with Bert-base (BERT) versus with our method (SpanEmo).	95
5.1	Data statistics (DailyS. = DailyStrength)	104
5.2	Network architecture and hyper-parameters.	107
5.3	Comparison of our models to those reported in previous work.	109
5.4	Word coverage.	112
5.5	Comparison of our method to those reported in prior work.	115
5.6	Data Statistics.	121
5.7	Experimental results on the test set.	126
6.1	Example Tweets from IEST dataset.	133
6.2	Hyper-parameters.	138
6.3	Statistics of datasets.	139
6.4	Comparison of our method to previous approaches.	141
6.5	Ablation experiment results.	142
6.6	Illustration of how intra- and inter-class scores are computed.	143

6.7	The results (%) of intra- and inter-class values.	144
6.8	Analysis of the model predictions trained on two settings.	148

List of Figures

2.1	Perceptron.	32
2.2	Single-layer Perceptron vs two-layer Perceptrons.	33
2.3	An illustration of over-fitting vs under-fitting.	37
2.4	Dropout Illustration.	37
2.5	Early stopping criterion.	38
2.6	An illustration of a Convolutional Neural Network.	40
2.7	An illustration of an unrolled-RNN.	41
2.8	An illustration of an LSTM architecture.	42
2.9	BERT input representation.	46
2.10	Text classification based on BERT.	47
2.11	Illustration of Triplet loss.	49
3.1	Illustration of TER.	51
3.2	Taxonomy of subjectivity, sentiment and basic emotions.	56
3.3	Emotion communication in text.	58
3.4	Categorical of emotions	61
3.5	Illustration of the different learning schemes for TER.	62
3.6	Russell’s circumplex model of emotion	63
3.7	Taxonomy of Emotion Recognition approaches.	68
3.8	An instance of text containing emotion cues.	74
3.9	An overview of the different TL strategies.	76
4.1	Illustration of our proposed framework (SpanEmo).	84
4.2	SpanEmo input construction.	85
4.3	Visualisation of an example	94
4.4	An analysis of emotion correlations.	97
4.5	Sensitivity analysis of the parameter (α).	98

5.1	An illustration of the two conducted experiments for the first case study (ADR).	105
5.2	Description of our framework.	106
5.3	F-score for our model with a different set of fine-tuning methods. . . .	111
5.4	Error Analysis of ADR instances.	113
5.5	The results of Daily Strength and Twitter when evaluated against the same test set (In-distribution) vs the other test set (Out-distribution). .	116
5.6	Illustration of two questions from the questionnaire with their answers.	120
5.7	Illustration of our framework.	122
5.8	The results of each SpanEmo-Encoder layer when applied to the validation data set of the depression task.	127
5.9	The results of varying the number of posts.	128
5.10	The results of varying the number of posts.	129
6.1	Illustration of our method (VTCL).	137
6.2	Prediction scores (y-axis) across emotions (x-axis).	146
6.3	t-SNE feature visualisation of CNN and BERT.	147
6.4	Our method with a range of C negative centres (x-axis).	149

Abstract

NEURAL NETWORKS FOR TEXTUAL EMOTION RECOGNITION AND ANALYSIS

Hassan Alhuzali

A thesis submitted to the University of Manchester
for the degree of Doctor of Philosophy, 2022

Textual Emotion recognition (TER) is an important task in Natural Language Processing (NLP), due to its high impact in real-world applications from health and well-being to author profiling, consumer analysis and security. The task of TER is often formalised as the process of detecting, interpreting, and understanding users’ emotions (i.e., the experience of feeling). This process can be performed on different units of analyses like words, phrases, sentences, documents and tweets/posts. Since the majority of existing emotion corpora are collected from social media data, the focus of this thesis is specifically on tweets and posts. This thesis investigates three research questions, as discussed below.

Firstly, we recommend that emotion correlations and associations should be taken into consideration when dealing with the classification and identification of emotion expressions in texts. This aims to enable TER models to account for the ambiguity and complexity of the task by taking into account that certain emotions can be highly correlated with each other. More specifically, we want to leverage information about emotion-to-emotion correlations, as well as associations between emotions and words in the case of multi-label emotion classification. To address the first research question, we propose “a novel model SpanEmo casting multi-label emotion classification as a span-prediction problem”, which can help TER models learn associations between labels and words in an input instance. Furthermore, we introduce a training objective focused on modelling multiple co-existing emotions in the input instance. Experiments performed on a multi-label emotion corpus across multiple languages demonstrate our method’s effectiveness in terms of improving the model performance and learning meaningful correlations as well as associations.

Secondly, existing emotion corpora labelled for a single-label emotion classification problem are more than those labelled for a multi-label emotion classification problem. Through extensive experiments, we observe that certain emotions are highly associated with each other, causing TER models to select incorrect predictions. Therefore, we want to improve TER models ability to handle highly associated emotions by introducing discriminator features. To address this, we introduce an auxiliary task to emotion classification. Furthermore, we introduce a method for evaluating the impact of intra- and inter-class variations on each emotion class. Experiments performed on three emotion corpora demonstrate our method’s effectiveness in terms of improving the prediction scores and producing discriminative features against highly confused emotions.

Thirdly, emotion features can be beneficial for related tasks that share common patterns with emotion. Based on the observation that in social media, negative sentiments and emotions are frequently expressed towards certain topics, such as politics, but also adverse drug reactions and depression. We examine the benefits of emotion features to the last two topics, while at the same time modelling them without the use of hand-crafted features. To avoid the use of hand-crafted features, we decide to use transfer learning by training a neural model on sentiment/emotion corpora and then fine-tuning it on the target tasks. We also adapt our proposed model for the first research question to both Adverse Drug Reactions (ADRs) and depression. Experiments performed on different corpora for the topics of ADRs and depression demonstrate our model’s effectiveness in achieving strong performance compared to previous approaches and being easily adapted to other tasks.

Declaration

No portion of the work referred to in this 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

- 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, trade marks 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=487>), in any relevant Thesis restriction declarations deposited in the University Library, The University Library’s regulations (see <http://www.manchester.ac.uk/library/aboutus/regulations>) and in The University’s policy on presentation of Theses
- v. In reference to IEEE copyrighted material which is used with permission in this

thesis, the IEEE does not endorse any of the University of Manchester's products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink. If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

Acknowledgements

By coming to the end of this scientific journey; all praises are firstly due to ALLAH for his merciful guidance during my stay in Manchester and in completing this thesis.



I would like to express my sincere gratitude to my supervisor, Professor Sophia Ananiadou, for having me as her student, for the trust, patience and freedom she offered me and for constant enthusiasm, support and encouragement about my work.

My gratitude and appreciation also goes to my viva committee members, Dr. Riza Batista-Navarro and Professor Ioannis Korkontzelos. I would like to thank them for the efforts and time spent on reading and reviewing my thesis, as well as for their valuable comments. These comments greatly contributed to the improvement of my thesis.

I would like also to thank all my colleagues at the National Centre for Text Mining: Paul, who provided great support with my English writing; Chryssa, whom I was very happy to share the same office, as well as who gave me a lot of invaluable advice and feedback during my PhD study; Fenia, who provided useful comments/suggestions while I was finalising my “IEEE-TAC-2021” paper; Kurt, who recommended me to attend “LxMLs2019”, from which I had an excellent/enjoyable experience and my other colleagues, Nhung, Meizhi, Maolin, Tianlin, Thy, Annika, Yifei, Jiarun, Erxue, Panos, Phong, Minh, Sunil, Sam, Emrah, Laura, Jake, Boyang. Due to the sudden pandemic, we unfortunately had to work away from each other and I have really missed daily gatherings during lunch time. I could not spend both my second and third years with you, but I still hope that we will have the opportunity to meet again in the future so I can express my gratitude. Thank you all for the wonderful memories, support, lunches, dinners and cakes.

My genuine appreciation goes to Umm Al-Qura University in Saudi Arabia for funding and supporting my study at the University of Manchester. This important milestone of my life would not have been possible without their full funding and support.

Finally, my deep and sincere gratitude goes to my family for their continuous love,

help and support. I have no words to express how grateful I am to all members of my family, including parents, brothers and sisters. Words cannot describe how thankful I am deeply to my wife (Hanan ) for her tremendous patience, understanding, encouragement and sharing this entire journey with me. I am also grateful for my two lovely kids (Lama and Jaad ) who made this journey not only more challenging, but also beautiful and vivid.

List of Abbreviations

BERT Pre-training of Deep Bidirectional Transformers for Language Understanding

ADAM Adaptive Moment Estimation Algorithm

CNN Convolutional Neural Network

LCA Label Correlation Aware Loss

TER Textual Emotion Recognition

NLP Natural Language Processing

LSTM Long-Short Term Memory

VTCL Variant Triple centre Loss

BCE Binary Cross-Entropy Loss

RNN Recurrent Neural Network

ADR Adverse Drug Reaction

FFN Feed Forward Network

EC Emotion Classification

CEL Cross-Entropy Loss

DS Distant Supervision

TL Transfer Learning

GD Gradient Descent

GT Ground Truth

Chapter 1

Introduction

1.1 Motivation

Emotion is a key aspect of human life, and hence Textual Emotion Recognition (TER) research is bound to have a wide range of real-world applications from health and well-being to user profiling, security and marketing, among others ([Mohammad and Turney, 2013a](#)). Emotion research also plays a significant role in improving human-machine interaction ([Fung, 2015](#); [Picard, 2000](#)). [Picard \(2000\)](#) states that human interaction with machines could be enhanced when they are enabled with proper understanding of emotion. She pointed out that emotion could help machines adapt to their users and better understand users' expressions and reactions. This indicates, for example, that chatbots may not be able to produce pleasant and positive language when they have little understanding and knowledge of emotion.

“Tay.ai” is a good illustration of a chat-bot made by Microsoft via Twitter that suddenly began to post offensive tweets, forcing the company to shut down the system. The bot had no notion of emotion (e.g., negative vs positive content); therefore it ended up posting abusive and offensive tweets. This incident is related to Picard's point where human-machine interaction could be improved by enabling machines to understand emotion and hence adapt to their users accordingly, while encouraging them to be aware of certain behaviours and feelings. This highlights the significant role of emotion research, a sub-field of affective computing science, that can lead to a positive impact on society.

As we mentioned above, research in TER has contributed to a wide range of real-world applications, e.g. health and well-being, author profiling and human-machine interaction, and we now turn to discussing below some of these applications that have

been influenced by the research in this area.

- **Author profiling:** Understanding an individual's personality provides valuable information about their thought and attitude towards decision-making, risks and preferences. Previous research has shown that emotion features are beneficial for author profiling (Mohammad and Kiritchenko, 2013, 2015a; Farnadi et al., 2014; Volkova and Bachrach, 2016). Mohammad and Kiritchenko (2013, 2015a) evaluated the use of emotion features on personality detection from text using “hashtag emotion lexicon”, corresponding to fine-grained emotions as well as equivalent to more than 500 words. They found a strong indicator between fine-grained emotions and being able to infer an individual's personality. Volkova and Bachrach (2016) also analysed communications in social media by contrasting between groups of users, i.e., those who expressed the same emotions from those who do not. In this respect, contrasting between the different groups contributed highly to inferring users' attributes, such as, age, gender, education, ethnicity, income, children and life satisfaction. Another study examined the relation between emotions and users' age, gender and personality, observing interesting properties between emotions and those users' attributes (Farnadi et al., 2014). One can further benefit from such information in understanding users' preferences towards products and services, which are of interest to businesses as they can gain a broader knowledge of their customer.
- **Customer experience:** Emotion is the main key to customer experience, where it can play an important role in driving brand preferences, purchase decisions, and customer loyalty. Understanding customer needs can not only boost their satisfaction, but also unlock their motivation. In the world of social media today, online users share experiences about many products and services which they interact with and then provide feedback, whether it be positive or negative. The feedback attempts to inform businesses about how to improve their services and products when users' needs are not met. Being responsive to the feedback can improve satisfaction and make customers loyal to such businesses because they see that their frustration and anger have been taken care of. Clibbon (2020) claims that “understanding customers in their own words to extract emotional territories is the key to transforming brand experience”. Thus, through understanding customer needs, businesses can adapt to them, which in turn drives brand growth in the longer term. Herzig et al. (2016) analysed customer support conversations taking place in social media to improve customer satisfaction as

well as determining which conversation must be dealt with right away. The authors found out that incorporating emotion features into their model led to better prediction of customer satisfaction and handling of prioritised conversations.

- Health and well-being:** Emotion analysis can play an important role in monitoring users' health and well-being, especially on social media because they share opinions on various aspects of life. Doctors and health professionals can also benefit from this monitoring system to track users' emotional and well-being changes, from which they can intervene immediately before something is escalated. For example, they could use the system to send alerts to care givers or request a visit when something is abnormal. This is even useful for mitigating major psychological disorders or preventing suicidal behaviours/thoughts. Some works utilised emotion-based features to specifically detect adverse drug reactions reported by users on social media, which can guide health professionals and pharmaceutical companies in making medications safer and advocating patient safety (Alhuzali and Ananiadou, 2019; Aragón et al., 2019; Chen et al., 2018; Khanpour and Caragea, 2018; Korkontzelos et al., 2016). Moreover, the idea of emotional contagion can further play a crucial role in either improving the overall well-being of users or preventing them from developing mental health problems. Kramer et al. (2014) states that emotions can be transferred to others through emotional contagion. This makes people experience the same emotions, even if they are not aware of their emotional changes. On the one hand, other works found a strong link between people's mental health problems (i.e., depression and anxiety) and the outbreak of Covid-19, due to the intense exposure to negative content on social media (Gao et al., 2020; Van Bavel et al., 2020). On the other hand, one can also expose people to positive or desired emotions (e.g., calm, joy, optimism and rest) to improve their overall well-being (Kramer et al., 2014).
- Security:** Potential hazards and dangerous behaviours of online users can be modelled by having access to knowledge about their emotional stability or sudden-change. In this respect, tracking online users' emotion can be vital for different reasons, including natural disasters (Desai et al., 2020) and flagging those who are threatening, abusive or risky (Karlgren et al., 2012). For example, O'Toole (2009) stated that individuals, who planned to cause some sort of harm, disclose

their intention in advance or during the planning stage. This is in particular relevant to social media, which people use to disseminate both negative and positive contents about various aspects of their life and intended behaviours.

Examples (Discussed ↓):

- S1. “school-name” 11 kids in a Year-10 class have tested positive for Covid in the last 10 days. The school was NOT ALLOWED to advise us following a change to the rules. Guessing 7 unnecessary infections, holidays ruined plus all the fear, anxiety and God knows how many people isolating #COVID19.
- S2. I expected the media to report this the same way they did regarding the floods in Germany and Belgium...but am totally disappointed 😡 ... Or because it is in Africa 🇳🇮 . This is how heavy rain left the city of Lagos in Nigeria.

The growing interest in TER has also been motivated by the proliferation of social media and online data, which have made it possible for people to communicate and share opinions on a variety of topics. Social media have become a key source, where people express their opinions, feelings and emotions. This is often caused by events and activities happening around the world, encouraging those individuals to express their attitude towards those events. Examples *S1* and *S2* highlight typical emotion expressions, where the first example reports anger and disappointment against “hiding information about infected kids at school” and the second example reports the same emotions against media outlets. We can observe that the examples indeed convey some emotional reactions. Analysing these emotions can help researchers study how people react to different situations and issues happening in real-time. This allows various companies to know what their users complain about exactly and in return work out ways to improve their services/products in the case of businesses. In a similar vein, authorities can benefit from such research by flagging risky, abusive or threatening behaviours.

In addition, as the volume of data increases on a daily-basis, it becomes almost impossible to process information manually. This challenge has given rise to new NLP methods focusing on TER identification and classification from social media data. One popular platform of social media, which has been used extensively by the community of NLP, is Twitter. Twitter makes data collection easy for large scale purposes, and it also allows users to utilise different symbols (e.g., hashtags, emoticons and emojis), some of which are rich in emotional expression. These symbols have helped the community to gather large data using hashtags (e.g., *#joy*, *#anger*, and *#fear*, among others)

as well as to take advantage of emoticons and emojis in analysing and understanding emotions (Shoeb and de Melo, 2020; Felbo et al., 2017).

Despite the increase in the number of publications for TER, there are challenges that hinder existing approaches in handling emotion expression effectively. This is because emotion is a complex phenomenon in that it is context sensitive as well as subjective. The sensitivity and subjectivity emerge from different personal and specific circumstances, where an emotive expression may be perceived differently by different people whose understanding and interpretation of the emotion depends on some factors (e.g., direct influence, previous experience, cultural differences etc.). Because of the multiple interpretations issue, an emotive expression is more likely to be associated with different emotions, yet similar to some extent to their valence space. For example, “anger and disgust” have been found to be highly correlated together in the case of multi-label emotion classification or even confused with each other in the case of single-label emotion classification (Agrawal et al., 2018; Mohammad and Bravo-Marquez, 2017a; Strapparava and Mihalcea, 2007).

The majority of existing approaches tackle TER without considering the above mentioned challenges by simply focusing on classification of emotion expressions. This limits the ability of TER models to better understand the complexity of the task, and more specifically, highly correlated and confused emotions. In the context of highly correlated emotions, it would be beneficial for TER models to learn to group together highly correlated emotions, while disentangling less correlated emotions. In the case of highly confused emotions, it would be useful for TER models to incorporate this information to improve their capability of introducing discriminated features between those highly confused emotions and hence, enhance their performance.

Examples (Discussed ↓):

S3. I get so [trigger_word] when parents smoke right next to their little kids. (Ground-Truth “GT”: disgust)

S4. I’m doing all this to make sure you’re smiling down on me bro. (GT: joy, love and optimism)

To give an example, consider *S3*, whose correct label is “disgust”, but a TER model may choose “anger”. Although labelling this example with “anger” is acceptable, it is incorrect from the single-label point of view. The model confusion can be attributed to the two emotion similarities in linguistic expressions as well as to the lack of explicit verbal cues. On the other hand, *S4* is labelled with three emotions that are often

correlated with each other. This example illustrates multi-label emotions, where positive emotions are highly correlated with each other, while they are less correlated with negative emotions. In this thesis, we address these issues and develop new computational approaches for TER that take emotion correlations into account in downstream applications, such as multi-label emotion classification. In addition, we demonstrate that our methods improve the discriminator capability of TER models to disambiguate between emotions.

1.2 Research Questions, Hypotheses and Objectives

This section describes our research questions, hypotheses and objectives, and discusses each one of them separately. In this thesis, negative labels or classes refer to wrong predictions made by a TER model, whereas negative emotions refer to negative emotional polarity (e.g., *anger*, *disgust*, *fear* and *sadness*)¹. We make this distinction clearer from the beginning in order to prevent any potential confusion or misunderstanding.

Research Question (RQ#1)

The first research question aims to extend the scientific understanding of how emotion is reflected in text in terms of tackling expressions evoking multiple emotions. TER research should focus on the objective that emotion expressions can have not only a single interpretation, but potentially multiple ones depending on the context and situation in which they occur. Limited research has been done to tackle the problem of detecting multiple emotions, while at the same time learning emotion-specific associations (i.e., associations between emotion labels and words in an input instance), as well as emotion correlations (i.e., label-label correlations). Our research question is if we can learn emotion correlations and associations with the purpose of improving TER performance as well as without using any external resources (lexicons or theories of emotion).

Research Hypotheses #1

- ★ Potential ambiguities, in which multiple emotions overlap, can be addressed by taking correlations between emotions into account such that highly correlated emotions are grouped together, while less correlated emotions are distant from each other.
- ★ Embedding descriptive label information with an input instance can help TER

¹More discussions and definitions will be given in Chapter 3.

models learn associations between emotions and words, which in turn reduce the effect of highly correlated emotions and enhance their performance considerably.

Research Objectives #1

- ★ Develop an instance-level emotion recognition model that learns correlations and associations in an end-to-end fashion (i.e., training a single model to learn both correlations and associations, as well as performing emotion recognition), and is independent of theories of emotion and emotion lexicons, to ease adoption to other languages and tasks.
- ★ Incorporate multiple co-existing emotions in the input instance into the training objective to improve the model capability in handling highly correlated emotions.
- ★ Validate the proposed approach on a widely used multi-label emotion corpus labelled over multiple languages and against state-of-the-art approaches.
- ★ Analyse the benefits of the learned associations and correlations for multi-label emotion classification.

Research Question (RQ#2)

The second research question aims to extend scientific understanding of the benefits of emotion in downstream applications. This is motivated by previous findings, demonstrating that emotion features can improve the performance of other tasks, especially when they share similarities. Nevertheless, prior research has relied on hand-crafted features engineering; we aim to address this by investigating the applicability of emotion features to two case studies or applications: adverse drug reactions (ADR) and early signs of depression detection.

Can we improve the results of both ADR and depression by taking into account knowledge of sentiment and emotion? Can we apply the developed model for **RQ#1** to improve identification of adverse drug reactions (ADRs) reported in social networks as well as the detection of early signs of depression?

Research Hypotheses #2

- ★ Emotion knowledge can be beneficial for downstream tasks. This is because negative sentiments/emotions are frequently expressed towards different topics, i.e., such as ADRs and depression.

Research Objectives #2

- ★ Develop a neural model with knowledge of sentiment/emotion that is trained and then transferred into both ADRs and depression.
- ★ Validate the claim that emotion features are beneficial for other tasks.
- ★ Apply the proposed model in **RQ#1** to the two case studies, i.e., the detection of adverse drug reactions and depression.

Research Question (RQ#3)

The third research question aims to extend the scientific understanding of how emotion is expressed in text in terms of studying highly associated emotions, especially in the case of single-label emotion classification. As mentioned in the above discussion, certain emotions (e.g., anger, disgust and sadness) can be confused with each other, due to the high associations between those emotions. It is worth mentioning that we use the term “highly confused or associated” emotions interchangeably in this thesis. This is because highly associated emotions can make TER models confused in terms of selecting the correct label.

Previous research has mainly focused on emotion classification, while ignoring the problem of highly confused emotions. In this research question, we aim to address this problem, by first adapting the concept of correlations from **RQ#1**. We define the concept of correlation for single-label emotion classification as input instances that are labelled with the same emotion. In other words, input instances labelled with the same emotion are more likely to share similar features rather than instances labelled with dissimilar emotions. Our research question is if we can improve the model ability to disentangle positive emotive expressions from negative ones. This notion is inspired by our proposed method in **RQ#1**, in which we aim to disentangle highly correlated emotions from negative ones. However, we focus in **RQ#3** on a single-label emotion classification problem instead of multi-label emotion classification. Specifically, our proposed method does not rely on label co-occurrences, which makes it more suitable for the single-label emotion classification case as such information can be found only in multi-label emotion classification datasets.

Research Hypotheses #3

- ★ Input instances labelled with specific emotions (e.g., *anger*, *disgust* or *sadness*) can be confused with each other, which can be addressed by improving the discriminator power of an emotion classifier. This can be achieved by incorporating intra- and inter-class variations.

- ★ Enabling TER models to incorporate variations within and between different classes of emotion can improve their capability in learning better discriminator features, and also enhance their performance considerably.

Research Objectives #3

- ★ Adapt the concept of correlation to single-label emotion corpora and then incorporate variations within and between different classes of emotion into the model.
- ★ Propose a training objective that enforces variations within and between different classes of emotion into the model.
- ★ Validate the proposed approach on widely used single-label emotion corpora, against state-of-the-art approaches and on each emotion class.
- ★ Analyse and evaluate the advantages of the proposed approach for TER.

1.3 Contributions and Publications

In this thesis, our novel contributions related to the above-mentioned research questions, hypotheses and objectives are:

The contributions corresponding to RQ#1.

- We proposed a novel model SpanEmo casting multi-label emotion classification as span-prediction. The main attributes of our SpanEmo model, in contrast to previous work, are summarised as follows: 1) the inclusion of emotion classes to the input instance, 2) the selection of predictions from the label segment directly, 3) the modelling of multiple co-existing emotions and 4) the independence from emotion lexicons and theories of emotion in learning both emotion correlations and associations, which makes it easily adaptable to other tasks, emotion corpora and languages.
- We incorporated emotion correlations into the training objective, enabling SpanEmo to model highly correlated emotions together, while distancing less correlated emotions from each other.
- We tested the applicability of SpanEmo on the SemEval-2018 multi-label emotion corpus (Mohammad et al., 2018) based on tweets labelled in English, Arabic and Spanish. This helps to show that our approach is language agnostic in the

sense that it can be easily adapted to emotion corpora in other languages without requiring any change in the model architecture.

The contributions corresponding to RQ#2.

- We investigated the benefits of emotion features in enhancing the detection of ADRs and depression.
- We adapted the architecture of SpanEmo to the detection of ADRs and depression by firstly training the model with knowledge of emotion and subsequently fine-tuning it on the chosen target tasks. We demonstrated that our SpanEmo model can be easily adoptable to other tasks and applications.
- We presented an in-depth analysis that illustrates the advantages and utility of using emotion data and SpanEmo model in enhancing the detection of ADRs and depression.

The contributions corresponding to RQ#3.

- We introduced a loss function as an auxiliary task to emotion classes by taking variations within and between different classes of emotion into account. This was inspired by our training objective introduced for RQ1, which attempted to separate a set of negative labels from the set of positive ones. The main attributes of our method, in comparison to prior methods, are summarised as follows: 1) the introduction and incorporation of the concept of intra- and inter-class variations into TER models, 2) the introduction of an evaluation method to quantify the benefits of intra- and inter-class variations on each emotion class, 3) the improvement of the discriminative ability of TER models and 4) the independence from emotion lexicons as well as theories of emotion in incorporating both intra- and inter-class variations, which makes it easily adaptable to other networks and corpora.
- We further proposed an evaluation metric that is able to compute intra- and inter-class variations for each emotion class.
- We evaluated our introduced loss function on three single-label emotion corpora, demonstrating that our approach helps TER models obtain high prediction scores, and is also a better discriminator against highly confused emotions.

It is worth mentioning that the majority of the work proposed in this thesis has been already published. This thesis includes existing, improved or additional details with respect to the following publications, as described in the corresponding chapters.

- **Chapter #4.** SpanEmo: Casting Multi-label Emotion Classification as Span-Prediction
 - ★ Alhuzali, H., and Ananiadou, S. (2021, April). SpanEmo: Casting Multi-label Emotion Classification as Span-prediction. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume (pp. 1573-1584).
- **Chapter #5.** Improving classification of Adverse Drug Reactions by Using Sentiment Analysis and Transfer Learning
 - ★ Alhuzali, H., and Ananiadou, S. (2019, August). Improving classification of adverse drug reactions through using sentiment analysis and transfer learning. In Proceedings of the 18th BioNLP Workshop and Shared Task (pp. 339-347).
- **Chapter #5.** Predicting Signs of Depression by Using Frozen Pre-trained Models and Standard Classifiers.
 - ★ Alhuzali, H., Zhang, T., and Ananiadou, S. (2021). Predicting Sign of Depression via Using Frozen Pre-trained Models and Random Forest Classifier. In Conference and Labs of the Evaluation Forum (CLEF-2021), (pp. 888-896).
- **Chapter #6.** Improving Textual Emotion Recognition Based on Intra- and Inter-class Variations
 - ★ Alhuzali, H., and Ananiadou, S. (2021). Improving Textual Emotion Recognition Based on Intra-and Inter-Class Variation. IEEE Transactions on Affective Computing (In Press).

1.4 Thesis Structure

The thesis consists of seven chapters. The first chapter (i.e., current one) is the introduction, in which we discuss our motivation, research questions, hypotheses and objectives. The contributions presented in this thesis are then discussed, along with those that are already published. Next, the thesis structure is given, with an outline

of the other chapters in the following order, i.e., an overview of both technical background and textual emotion recognition, three main chapters and finally the conclusion chapter.

In Chapter 2, we offer some technical background on neural network components and some of the advanced techniques that inspire our works presented in this thesis. This aims to help readers transition easily from chapter to chapter, as well as helping them when reaching future chapters. Chapter 3 provides an extensive overview of the task of emotion recognition, in which we start with the problem definitions and then move on to discuss terms related to emotion. Next, we discuss models of emotion that tend to provide a taxonomy of emotion classification and categorisation. We further describe existing emotion corpora as well as common approaches to TER. Finally, we conclude with some observations and limitations of previous research and demonstrate how our works presented in this thesis build upon them.

In Chapter 4, we cover our initial research question (i.e., **RQ#1**). We first mention the motivation behind SpanEmo and then discuss it in greater detail, including its architecture and training objective. After that, we elaborate on our experiments, i.e., implementation details and dataset and task settings. A large number of experiments and analyses both at the word- and tweet-level are also conducted on the different components of SpanEmo to better understand its behaviour. It is worth noting that the contents of this chapter have been published in [Alhuzali and Ananiadou \(2021b\)](#).

In Chapter 5, we first introduce each case study and highlight some of the motivations behind the use of emotion/sentiment knowledge to improve the detection of ADR. An overview of the related work with respect to the task of interest is given. Next, we discuss some experimental details, including our proposed approach, evaluation and analysis. This chapter specifically addresses **RQ#2**, in which we examine the effect of emotional knowledge to the improvement of ADR. To achieve this, we demonstrate how a neural model can be trained to acquire knowledge of sentiment/emotion and then adapted to the task of ADR. We run two experiments: 1) we are interested in testing our hypothesis to determine whether adapting a pre-trained model on sentiment data can help improve the classification of ADRs. 2) If the first experiment shows improvement, we expect to obtain the same, or even better results when the model is pre-trained with emotional knowledge, due to the inclusion of fine-grained annotation of emotions rather than coarse-grained annotation (i.e., *positive vs negative*). In this experiment, we use the SpanEmo architecture and follow the initial experiment setting presented in this chapter, by making use of transfer learning. We also run the same experiments

on detecting early signs of depression, but without any fine-tuning. Finally, parts of this chapter have been published in [Alhuzali and Ananiadou \(2019\)](#) and [Alhuzali et al. \(2021\)](#).

In Chapter 6, we examine the last **RQ#3** and provide motivation about the importance of incorporating intra- and inter-class variations within and between different classes of emotion. Then, we review some related approaches and describe how our proposed approach builds upon them. Next, an overview of our implementation and the used corpora are included. Finally, we provide extensive evaluations and analyses of the proposed approach and its benefits to the task of TER. It should be mentioned that this chapter is drawn from [Alhuzali and Ananiadou \(2021a\)](#).

In the last chapter, we synthesise the findings of each individual chapter and highlight the important observations from the overall thesis. We then refer to the limitations of the works presented, including those related to the task of TER as well as those related to the proposed models in this thesis. Finally, we discuss future directions. In the reminder of this thesis, we use examples of text to motivate our novel methods and to describe our results/analyses. It should be mentioned that some of those examples contain words that might be found offensive by some readers.

Chapter 2

Technical Background

Before the revolution of neural networks in early 2010, most research in Natural Language Processing (NLP) focused on developing many types of feature templates derived from domain knowledge. This type of feature engineering was time-consuming and expensive, required domain expertise to generate features, and was not generalisable. However, with the advent of neural networks, hand-crafted features can be avoided by simply learning features from text automatically. Learning features automatically has given rise to new NLP methods that enable better representation learning, e.g., word embeddings ([Mikolov et al., 2013a](#); [Pennington et al., 2014](#); [Agrawal et al., 2018](#)) and contextual embeddings ([Peters et al., 2018b,c](#); [Devlin et al., 2019](#)). In this respect, it has become common to use such representation learning approaches to learn features that are beneficial for many NLP tasks, including Textual Emotion Recognition (TER), which is the main topic of this thesis. We now turn to describing neural networks, their training process and popular types of neural networks, from which this thesis benefits.

2.1 Neural Networks

Neural Networks (NN) are computing mechanisms that were inspired by efforts to mimic the processing of information in the brain. The basic unit of NN is the neuron, which is also called a unit or node. The neuron takes an input from an external resource (e.g., text or image) and then computes an output. Each input is associated with a weight, and both are eventually multiplied together. The computation of a single neuron is often called a Perceptron, formulated via a weighted summation of its inputs.

Figure 2.1 depicts the Perceptron and its computation is shown in equation (2.1).

$$y = f(\mathbf{w}^\top \mathbf{x} + b), \quad (2.1)$$

where y represents the output of the neuron, f is an activation function, $\mathbf{x} \in \mathbb{R}^n$ are the inputs to the network, $\mathbf{w} \in \mathbb{R}^n$ is a vector with the associated weights and b is a scalar value.

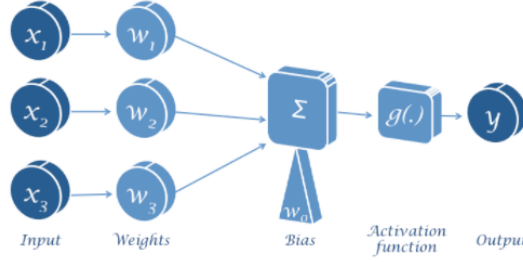


Figure 2.1: Perceptron. Adapted from the following blog: [“Intro to Perceptron”](#)

The first and simplest type of neural network is known as the Feed-Forward Network (FFN), which consists of multiple neurons combined together. Similar to the computation of the Perceptron, we can reformulate the computation of a single-layer Perceptron as follows.

$$\mathbf{y} = f(\mathbf{W}\mathbf{x} + \mathbf{b}), \quad (2.2)$$

where $\mathbf{W} \in \mathbb{R}^{d \times n}$ is a weight matrix multiplied by the input vector $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^d$ is a bias vector and $\mathbf{y} \in \mathbb{R}^d$ is the output vector of the network, with d denoting the dimensional size. Figure 2.2a illustrates the single-layer Perceptron. The single-layer Perceptron can overcome the limitation of the Perceptron, which can only solve linear decision boundaries. A famous yet intuitive example is the logic XOR operator, which obviously requires non-linear decision boundaries to be solved. The single-layer Perceptron can also be extended to multi-layer Perceptrons, where the input is connected to the output via an intermediate hidden layer. An instance of this is a two-layer network shown in Figure 2.2b.

The output of the two-layer network can be computed as in equation (2.3). With the analogy of the two-layer network, one can construct deep neural network architectures by stacking multiple layers.

$$\hat{\mathbf{y}} = f_2(\mathbf{W}_2 f_1(\mathbf{W}_1 \mathbf{x} + \mathbf{b}_1) + \mathbf{b}_2), \quad (2.3)$$

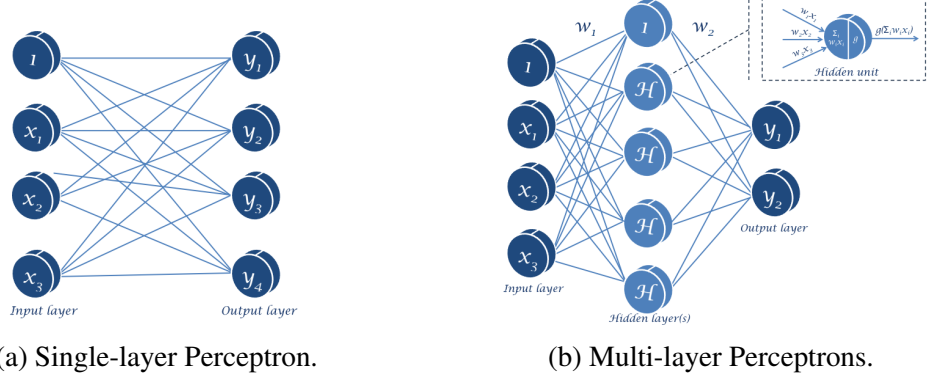


Figure 2.2: Single-layer Perceptron vs two-layer Perceptrons, Adapted from ([Intro to Perceptron](#))

where $\mathbf{y} \in \mathbb{R}^m$, $\mathbf{W}_1 \in \mathbb{R}^{d \times n}$, $\mathbf{W}_2 \in \mathbb{R}^{m \times d}$, $\mathbf{b}_1 \in \mathbb{R}^d$, $\mathbf{b}_2 \in \mathbb{R}^m$. f_1 and f_2 are two hidden layers, where the second layer takes its input from the first layer (i.e., f_1). \hat{y} represents the final output, which can be used directly for classification.

2.2 Neural Network Training

2.2.1 Classification

We have already defined the output of a multi-layer neural network as a vector \hat{y} . The last layer of the network converts the output to a vector of dimensionality equivalent to the number of classes ($\hat{y} \in \mathbb{R}^C$) in classification settings (i.e., multi-class or multi-label). Each score in the vector maps to a class in un-normalised form. To transform the un-normalised scores of the output vector, we typically use the softmax activation function in the case of multi-class classification. The softmax activation function squashes the scores between zero and one (i.e., $[0, 1]$) that add up to one. Because of this, it has a probabilistic interpretation, where one can determine how much confidence the network assigns to a class. In other words, we can compute the probability p_c of a class $c_i \in C$ given an input vector \mathbf{x} as shown in equation (2.4).

$$p_c = \text{softmax}(\hat{y}_c) \equiv \frac{\exp(\hat{y}_c)}{\sum_{j=1}^C \exp(\hat{y}_j)}, \quad (2.4)$$

where \hat{y}_c is the un-normalised score obtained from the network for class c . In the case of multi-label classification, we typically employ the sigmoid activation function, which squashes the score of each class between zero and one. Then, a threshold value

is set to select whether or not the class should be chosen. The sigmoid function can be computed as follows.

$$p_c = \text{sigmoid}(\hat{y}_c) \equiv \frac{1}{1 + e^{-\hat{y}_c}}, \quad (2.5)$$

where \hat{y}_c is the un-normalised score obtained from the network for class c that receives an independent probability score from the rest of classes. If the received score is above the threshold, the class will be selected.

2.2.2 Loss Function

A network can be defined as a function that maps between inputs and outputs. It is parameterised by a set of weights (e.g., \mathbf{W} and \mathbf{b}). It is often the case that the data (i.e., input and output) is given and fixed, and hence we do not have control over them. However, we can control the weights such that the predicted scores are compatible with the correct classes in the training. In this respect, the loss function, which is known as the cost function or the objective, measures our satisfaction with outcomes produced by the network. When the loss is high, the network performs poorly, while performing well when the loss is low.

Let $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ be a set of N examples with the corresponding class y_i , where \mathbf{x}_i denotes the i^{th} example and y_i represents the correct class associated with this example. We define $p(y_i | \mathbf{x}_i)$ being the probability of example i predicted by a network and θ representing the parameter set of the network. The loss function $\mathcal{L}(\theta)$ is computed over the entire data set by averaging its losses. Computing the loss for a classification task is based on the cross-entropy, which optimises the negative log likelihood of the correct class.

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^N -\log p(y_i | \mathbf{x}_i) \quad (2.6)$$

2.2.3 Learning

Back-propagation. The process of network optimisation follows right after the loss is computed, for which we update the parameters of the network utilising the back-propagation algorithm (Rumelhart et al., 1986). The goal of the back-propagation algorithm is to compute the partial derivative of a loss function with respect to any parameter in the network. In other words, this algorithm back-propagates errors from the output layer to the input layer. By using this algorithm, we can compute the gradient with respect to the input. In the case of a scalar input x , let us assume we have the

following expression $(x + y)z$. We can easily differentiate this expression directly, but will take an alternative approach that helps us illustrate the process of back-propagation intuitively. To simplify the process, let us first assign u to represent $(x + y)$ and let $f = uz$. Here f is a multiplication of both u and z . We can now compute the partial derivative of the expression with respect to each term as shown in Equation (2.7).

$$\frac{\partial f}{\partial u} = z, \frac{\partial f}{\partial z} = u, \frac{\partial u}{\partial x} = 1, \frac{\partial u}{\partial y} = 1 \quad (2.7)$$

In the case of NNs, we are always interested in computing the partial derivative of the output f with respect to the input variables (i.e., x, y, z) as follows:

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial u} \frac{\partial u}{\partial x} = z \cdot 1 = z, \quad (2.8)$$

$$\frac{\partial f}{\partial y} = \frac{\partial f}{\partial u} \frac{\partial u}{\partial y} = z \cdot 1 = z, \quad (2.9)$$

$$\frac{\partial f}{\partial z} = u = (x + y), \quad (2.10)$$

We have computed so far the gradient for scalar inputs, but this analogy can easily be extended to other forms like vectors. To give an example of the process, let us assume that a neural network has the following expression, in which we assign the inner function to u and the outer function to g for simplicity.

$$\mathbf{f}(\mathbf{x}, \mathbf{W}) = (\mathbf{W}\mathbf{x})^2, \mathbf{u} = \mathbf{W}\mathbf{x}, \mathbf{g} = \mathbf{u}^2 \quad (2.11)$$

As shown in Equation (2.11), we are interested in computing the gradients of the two parameters (i.e., $\frac{\partial g}{\partial \mathbf{W}}$ and $\frac{\partial g}{\partial \mathbf{x}}$). This can be done by computing the gradients of these two parameters beginning from the output of the network and moving backwards up to the input layer. It is worth mentioning that the output dimension of each gradient equals the dimension of the respective parameter. The process discussed above can be extended to networks with multiple layers.

$$\begin{aligned} \frac{\partial g}{\partial \mathbf{W}} &= \frac{\partial g}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \mathbf{W}} = 2\mathbf{u}\mathbf{x}^\top \\ \frac{\partial g}{\partial \mathbf{x}} &= \frac{\partial g}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \mathbf{x}} = 2\mathbf{W}^\top \mathbf{u} \end{aligned} \quad (2.12)$$

Parameter update. After computing the gradients, we can update the network parameters (θ), where θ_t represents the parameter set at iteration t . One of the most

widely used optimisation algorithms is Gradient descent (GD) (Cauchy et al., 1847). Its main objective is to minimise a loss function $\mathcal{L}(\theta)$ with respect to the parameters of the network, which are updated globally over the entire training set. The update is often set towards the opposite direction of the gradient of the loss.

$$\theta_{t+1} = \theta_t - \eta \frac{\partial \mathcal{L}}{\partial \theta_t}, \quad (2.13)$$

where η is known as the learning rate that is responsible for scaling the volume of updates. Large learning rate tends to make large changes to the network parameters, whereas small learning rate tends to make small changes. The learning rate η is a hyper-parameter that needs to be tuned depending on the chosen algorithm of optimisation. There are other variants of GD (Ruder, 2016), such as Stochastic GD and mini-batch GD, where the SGD estimates the gradient of the loss for each sample at a time, while the mini-batch GD computes the gradient of the loss for every mini-batch of n training samples. The second variant benefits from both GD and SGD. Besides GD, there are other optimisation algorithms that tend to work well for training NNs, e.g., Adam (Kingma and Ba, 2014).

2.2.4 Regularisation

A major challenge in training a neural network is how to make them perform well both on the training set and the test set (i.e., unseen inputs during training). When training the neural network, we often evaluate it against another set, known as the validation or development set. Once the training process is over, we test the network capability on a second different set called the test set. The key idea of such evaluation is to assess the network generalisation on unseen inputs during training, on which it should obtain similar or comparable results to the ones observed on the validation set. Since this is not always the case, some regularisation techniques are introduced to reduce both under-fitting and over-fitting as shown in Figure 2.3. We briefly discuss a couple of regularisation techniques (i.e., dropout and early stopping) that are relevant to the works presented in this dissertation.

Dropout is an effective way to regularise deep NNs during training. This means that, unlike in the past, when the network parameter weights were learned together, some parts of the network weights are learned in the iteration steps instead. It was proposed by Srivastava et al. (2014) to prevent NNs from co-adaptation. This means that

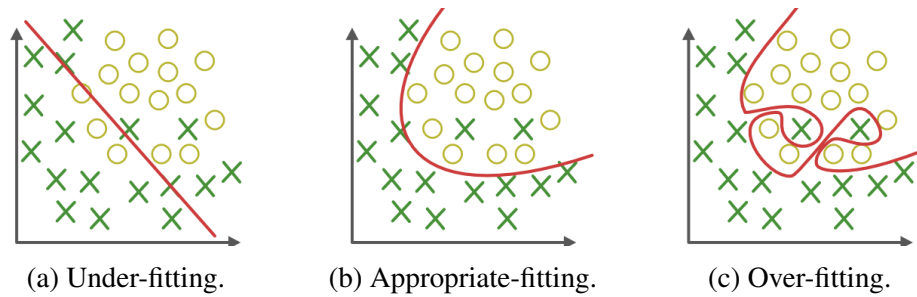


Figure 2.3: An illustration of over-fitting vs under-fitting. Taken from ([geeks.org](https://www.geeks.org)).

some neurons have stronger predictive capability than others, which can lead the network to over-fitting by only concentrating on the strong signals. The main advantage of dropout is to address this problem by randomly replacing weights in the networks with zeros and the amount is determined by a probability $p \in [0, 1]$, set in advance before the training process begins. Figure 2.4 presents an illustration of dropout. The process of dropout is applied to different units at each iteration step, and it produces a different model in each iteration, which eventually results in an ensemble. Another benefit of dropout is the incorporation of noise into the network via the missing units. Finally, the probability p is a hyper-parameter that can be tuned for different parts of the network.

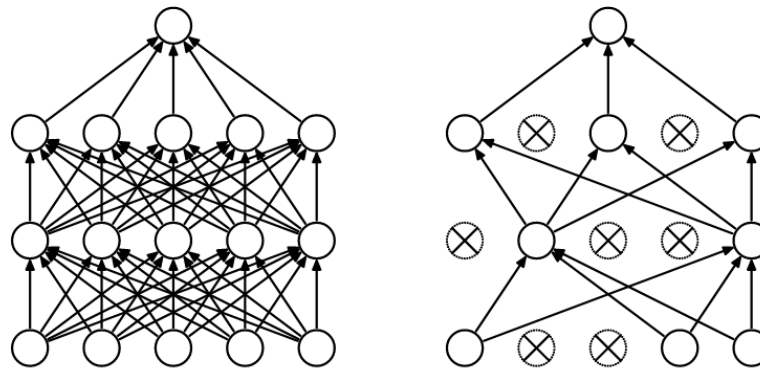


Figure 2.4: Dropout Illustration. Standard Neural Network (left) and after dropout is applied (right). Taken from ([Srivastava et al., 2014](#)).

Early Stopping is another regularisation technique that indicates when the network should stop the training process ([Caruana et al., 2001](#)). The criteria for when to stop the training process are determined based on the validation loss when it reaches a plateau or starts to increase. Figure 2.5 depicts the early stopping criterion. It can be clearly observed that once the validation loss deteriorates drastically from the training loss, it becomes obvious to stop the network from training as it is more likely to overfit the

training data. Typically, a hyper-parameter, known as “patience”, indicates how many epochs are required before the training should stop. After the training is terminated, we select the model parameters that achieve the lowest validation loss.

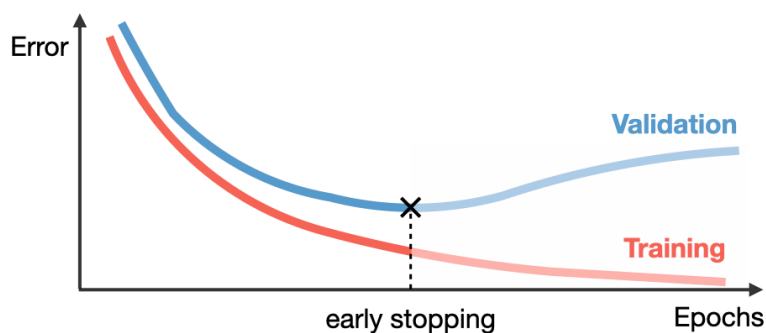


Figure 2.5: Early stopping criterion. Taken from ([Stanford-cs-230](#)).

2.3 Neural Network Approaches

We discussed in the previous sections that NNs take an input, and transform it via a series of hidden layers. Each hidden layer is composed of a set of neurons, where each one is connected to all neurons in the previous layer. The last output layer is called a fully-connected layer, which represents the classes in classification problems. There are several NNs that have been developed in the literature for different purposes (e.g., Convolutional NNs, Long-Short Term Memory, Attention Mechanisms and Pre-training of Deep Bidirectional Transformers for Language Understanding). We first describe the initial step of transforming words into real-valued vectors and then discuss each one of these four architectures. Next, we discuss the concept of “fine-tuning”, describing how NNs trained on specific domains and corpora can be adapted to other domains and corpora as well. Finally, we conclude with deep metric learning.

2.3.1 Word Representations

The initial step of any NLP model is to represent words in real-valued vectors, where each vector maps to a single unique word. The process of converting the input of a network is called embedding. The first step in performing this process is by defining a dictionary D that consists of all the words in a corpus. Next, a mapping between each word and its associated real-valued vector is created. The embedding layer (EL)

is then built to form a look-up table (Collobert and Weston, 2008), where each word $i \in D$ is embedded into a d -dimensional feature vector.

$$\text{EL}_W(i) = \mathbf{W}_i, \quad (2.14)$$

where $W_i \in \mathbb{R}^d$ is the i^{th} column of a matrix W and d denotes the dimensionality of the column vector. The concept of word embeddings/representations has found its popularity in NLP right after the work of Mikolov et al. (2013b) who proposed the “Word2Vec” algorithm. This algorithm is based on a neural network to learn word associations from a large data set via the use of a context window as a hyper-parameter. Then, any type of NNs can use word embeddings to initialise or represent their inputs. Subsequently, another word embedding is introduced by Pennington et al. (2014) known as “Glove”. The Glove embedding is also an unsupervised learning algorithm for obtaining word representations. This is based on the concept of word-word co-occurrence statistics that are learned from a large data set. The learned co-occurrences are then aggregated and mapped into a meaningful space, where common co-occurred words are distantly similar, whereas less co-occurred words are distantly dissimilar. The above discussed word embeddings are called static word representations because each word is represented by the same embedding/vector even if the contexts of words have changed.

2.3.2 Convolutional Neural Networks (CNN)

The architecture of CNN was firstly adapted to images, and it was proposed for automatic training of the Convolutional map/kernel (LeCun et al., 1998). The Convolutional layer consists of a set of learnable parameters called filters. It was firstly introduced to text by Kim (2014). Figure 2.6 illustrates how it works on a sequence of words. The first layer is basically the embedding layer, which transforms each word into a real-valued vector. The embedding layer is a matrix of words and their associated real-valued vectors that is then fed into a convolutional layer with a non-linear activation. As shown in Figure 2.6, we can utilise multiple filters, where each one learns a different set of features, similar to how n-grams are used to extract features from text. After that, a pooling operation is used, which can be one of the following, e.g., a max, min or mean operation. The purpose of the pooling operation is to select the most salient and important features. Let us now distil each component of the CNN network. Given a sequence of words $s = [\mathbf{w}_1; \mathbf{w}_2; \dots; \mathbf{w}_n]$, where $w_i \in \mathbb{R}^k$ represents

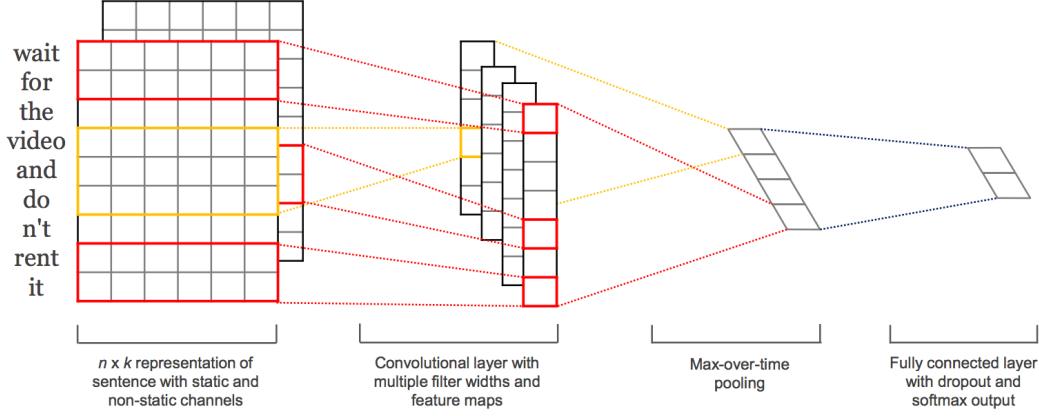


Figure 2.6: An illustration of a Convolutional Neural Network (CNN). Adapted from (Kim, 2014)

the i^{th} word in the sequence, k denotes the dimensional size and “;” represents the concatenation of n words. Later, we use a filter $f \in \mathbb{R}^{h \cdot k}$, with dimensionality equivalent to the number of a sliced window of words h and their dimension k . Again, the sliced window in the convolutional layer mimics the process of n-grams. The computation of one filter on a window size of h words results in a single feature c_i , as depicted in Equation (2.15).

$$c_i = g(\mathbf{f}^\top \mathbf{w}_{i:i+h-1} + b) \quad (2.15)$$

where $\mathbf{w}_{i:i+h-1} \in \mathbb{R}^{h \cdot k}$ is the concatenation of h words, b is a scalar and g is a non-layer activation function. This process is performed on all slices of the sequence, producing a set of features called the feature map, Equation (2.16). The feature map output is then passed through the same pooling operation discussed above, such as the max as shown in Equation (2.17). When multiple filters are utilised, one can concatenate all m filters. This produces a single vector that is then fed into the output layer (i.e., a fully-connected layer).

$$\mathbf{c} = [c_1, c_2, \dots, c_{n-h+1}] \quad (2.16)$$

$$\hat{c} = \max(\mathbf{c}) \quad (2.17)$$

2.3.3 Recurrent Neural Networks (RNN)

When learning a new skill, humans do not begin from scratch. In other words, they make use of previous experiences which have been accumulated over time. This sequential process, which cannot be achieved by CNN and Perceptron, leads to the emergence of RNN that process information in loops. Every loop performs some computation on an input and then passes it to the next step. Figure 2.7 presents an unrolled RNN, depicting the process of an unrolled loop.

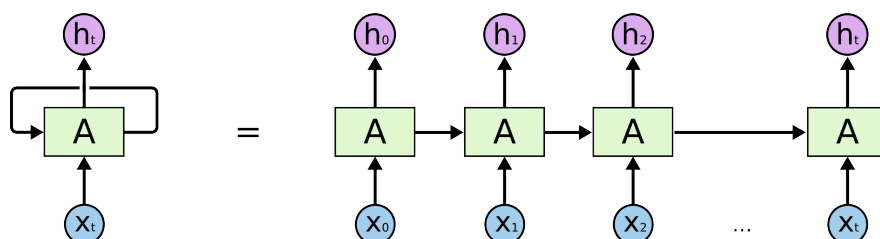


Figure 2.7: An illustration of an unrolled-RNN. Taken from ([colah.github](https://colah.github.io/)).

From Figure 2.7, we can see that such networks can be effective for text due to the nature of how words are grouped to form a sequence. There are different types of RNN that have been introduced (e.g., Long-Short Term Memory (LSTM) and Gated Recurrent Unit (Choi et al., 2014)). Gated Recurrent Unit (GRU) is another variant of RNN and is quite similar to LSTM. However, GRU differs from LSTM in the sense that it consists of only two gates (i.e., reset and update gates). In this respect, GRU contains less training parameters, which makes it fast to execute. In this section, we only describe LSTM since it is widely used in NLP.

Long-Short Term Memory (LSTM). One of the positive properties of RNNs is the notion that they can connect history to the future (i.e., previous information to present information). However, how much history is relevant to the current task is negligible. For one task, it might be useful to look far back in history, while it may not be the case for another task. If the former is the case, RNN fails to model such long dependency, which is known as the problem of long-term dependencies. LSTM was proposed by Hochreiter and Schmidhuber (1997) to overcome this challenge, which performs well in comparison to its counterpart RNN and has become widely used for many tasks (e.g., language modelling (Howard and Ruder, 2018), machine translation (Bahdanau et al., 2014), summarisation (See et al., 2017), text classification (Alhuzali and Ananiadou, 2019), etc.). Figure 2.8 presents the LSTM architecture, which still has a chain-like structure, similar to that of RNN.

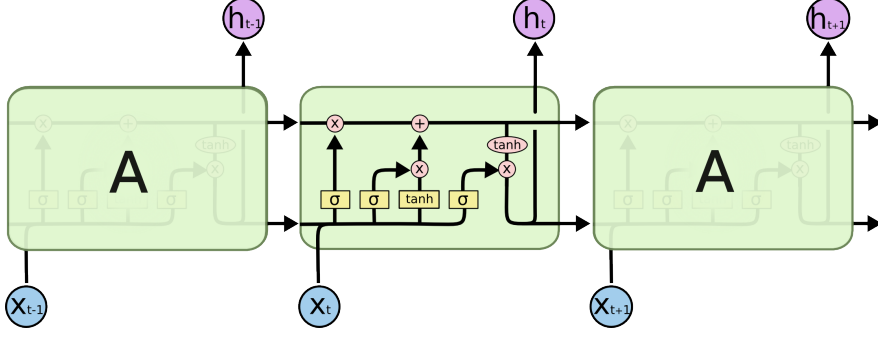


Figure 2.8: An illustration of an LSTM architecture. Each block shows an abstract representation of an LSTM cell. Yellow boxes correspond to hidden layers, while pink circles correspond to pointwise operation. Taken from ([colah.github](https://colah.github.io/)).

The main idea of LSTM lies in its ability to retain useful information, as well as removing redundant information. This is achieved by introducing three gates, each one of them is responsible for different objectives. The first gate (left) is called the forget gate layer that receives a previous hidden state h_{t-1} and a current input x_t (e.g., a word), Equation (2.18). Both are then fused together through a sigmoid activation function to determine which information to exclude from the cell state (top black line).

$$f_t = \sigma(\mathbf{W}_f \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f) \quad (2.18)$$

where \mathbf{W} and \mathbf{b} are trainable parameters and σ denotes the sigmoid activation function. Once the information removal process is completed, the next step is to determine which new information should be stored in the cell. This gate is called the input gate layer, which indicates which values that need to be updated. Next, another layer with a tanh activation function creates a vector of candidate values that are then combined with the input gate and added to the cell state, Equation (2.19).

$$\begin{aligned} i_t &= \sigma(\mathbf{W}_i \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i) \\ \tilde{c}_t &= \tanh(\mathbf{W}_c \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_c) \end{aligned} \quad (2.19)$$

At this point, we update the previous cell state by taking into account the forget and input gates, Equation (2.20). The forget gate is multiplied by the previous cell state and it is then added to the input gate.

$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t \quad (2.20)$$

Finally, the computation of the above steps (c_t) is passed through a \tanh activation function to round values in the range $\in [-1, 1]$. Another gate is created, known as the output gate, which is fused with $\tanh(c_t)$ via a multiplication operation, and which decides what aspects of the cell state to output, Equation (2.21).

$$\begin{aligned} o_t &= \sigma(W_o [\mathbf{h}_{t-1}, \mathbf{x}_t] + b_o) \\ h_t &= o_t * \tanh(c_t) \end{aligned} \quad (2.21)$$

The LSTM architecture can be applied to sequences from left-to-right and from right-to-left. This concept is called “Bi-directionality”, which was firstly proposed by [Schuster and Paliwal \(1997\)](#) for RNNs. A bi-directional LSTM processes both past and future information all at the same time. The future states may include additional contextual information that aids the network to take advantage of it. Consider a simple example, *I am blank in / to...*, where the future choice of either “in” or “to” is a strong indicator of which word to use in the blank (i.e., interested or interesting). Allowing the network to have access to such information can help in the disambiguation of both options and eventually the selection of the correct one. The output of bi-directionality is typically concatenated together to form a final representation for the input sequence as follows:

$$\hat{\mathbf{y}}_t = [\vec{\mathbf{o}}_t; \overleftarrow{\mathbf{o}}_t], \quad (2.22)$$

where $\vec{\mathbf{o}}_t$ and $\overleftarrow{\mathbf{o}}_t$ represents the output from left-to-right and from right-to-left, respectively. “;” denotes a concatenation operation and $\hat{\mathbf{y}}_t$ is the final output of the bi-directional LSTM.

2.3.4 Attention Mechanisms

The concept of attention is derived from humans’ visual attention, which enables them to focus on a specific region of an image. The same notion can also be seen in text, especially in reading. It is often the case that we do not read a whole sentence from left-to-right, but we skip certain words of the sentence and can still understand the meaning of the whole sentence. This is a powerful mechanism that could help NNs for effective understanding of a piece of text.

In NLP research, attention mechanisms were firstly applied to machine translation by [Bahdanau et al. \(2014\)](#) who used the attention to jointly align and translate at the same time. After this work, attention mechanisms have become popular in NLP, and utilised for different tasks beyond machine translation, such as classification ([Alhuzali](#)

and Ananiadou, 2019; Felbo et al., 2017), summarisation (See et al., 2017; Cheng and Lapata, 2016), etc. The popularity of attention in NLP led to enhanced mechanisms that serve different objectives but still share the same function in that they assess the importance of different elements in a sequence with respect to other elements in the same sequence. Given an input sequence $x = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_n\}$, the attention weight a_i is then estimated by normalising the weights via the softmax activation function across the entire sequence as follows.

$$\mathbf{a}_i = f(\mathbf{W}_i, \mathbf{h}_i), \quad (2.23)$$

$$\alpha_i = \text{softmax}(\mathbf{a}_i) = \frac{\exp(\mathbf{a}_i)}{\sum_{j=1}^n \exp(\mathbf{a}_j)}, \quad (2.24)$$

$$r = \sum_{i=1}^n \alpha_i \cdot h_i \quad (2.25)$$

where $\mathbf{W} \in \mathbb{R}^{m \times d}$ is a trainable attention parameter, $\mathbf{h}_i \in \mathbb{R}^d$ is the hidden representation corresponding to an element in the sequence, α_i is the normalised attention score corresponding to the element i and n is the size of the sequence. f is an activation function, and r holds the attention weights associated with each element in the sequence.

Self-attention, also known as intra-attention, is an attention mechanism allowing sequences to interact with each other and find out which part of the sequence should pay more attention to what with respect to an element. The interaction is then aggregated to form the attention scores. Self-attention was introduced by Lin et al. (2017) and applied to different NLP tasks, including machine reading comprehension (Zhuang and Wang, 2019), abstractive summarisation (Xu et al., 2020b), image description generation (Li et al., 2020), emotion recognition (Chronopoulou et al., 2018), aspect sentiment analysis (Cheng et al., 2017). The self-attention module takes n inputs and returns n outputs. In other words, the attention vector \mathbf{a}_i for an element in a sequence (e.g., a word i) is computed as follows.

$$\mathbf{a}_i = \mathbf{v}_a^\top \tanh(\mathbf{W}_a \mathbf{h}_i) \quad (2.26)$$

where $\mathbf{v} \in \mathbb{R}^m$ and $\mathbf{W} \in \mathbb{R}^{m \times d}$ are trainable attention parameters and \mathbf{h}_i is the representation of an element in the input sequence. This process is performed on the whole input sequence to extract important spans. The vector \mathbf{v} becomes a matrix $\mathbf{V} \in \mathbb{R}^{m \times r}$, which contains r attention weights for element i of the input sequence.

Multi-head Attention is an extension of the self-attention mechanism proposed by Vaswani et al. (2017). Based on this type of attention, a network, known as the Transformer, was built that demonstrates strong performance and even surpasses other NNs (e.g., RNNs and CNNs) on various NLP tasks. Previous studies often utilised the attention mechanism as an assistant block in a network to enhance an input representation, while Transformer-based networks used multi-head attention as their main source of computation. The use of multi-head attention divides the input into multiple heads, each of which is made of self-attention that ultimately learns different views of the input.

Several approaches were inspired by the Transformer architecture, including BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019), GPT (Radford et al., 2018; Brown et al., 2020), etc. These networks have been pre-trained on a huge volume of data in unsupervised fashion showing that they can gain some useful knowledge about the use of language in text (Qiu et al., 2020). After pre-training, we can apply such models to various NLP tasks through what is known as “fine-tuning”. Both pre-training and fine-tuning have become the de facto standard in NLP research nowadays. We turn now to describing the BERT model before covering the concept of fine-tuning.

2.3.5 Pre-training of Deep Bidirectional Transformers for Language Understanding (BERT)

This model is designed to perform two steps, i.e., pre-training and fine-tuning. The first step involves pre-training BERT on a large amount of unlabelled text by benefiting from both left and right contexts, corresponding to the idea of bi-directionality in the title. Once this step is done, the pre-trained model is then fine-tuned on any target task that has shown to achieve state-of-the-art results across a variety of NLP tasks. As the title includes the word “Transformer”, BERT itself is based on the Transformer’s architecture.

Let us first understand how BERT processes text. Figure 2.9 depicts an example of the input text, which contains three embeddings: 1) position embeddings, 2) segment embeddings and 3) token embeddings. The first embedding is responsible for encoding positional information since the BERT model follows the Transformer architecture that processes tokens in parallel instead of in order. This way, the model learns the position

embedding of a word in the text input. The second embedding is the segment embedding, which helps the model to distinguish segment A from segment B in the case of tasks requiring sentence pairs, e.g., question answering, entailment, paraphrasing, etc. Finally, the token embedding represents the vector representation for each word in the sentence, which is based on the vocabulary set used by the BERT model. An important token is the $[CLS]$, which is always used for representing the whole input text in the case of text classification.

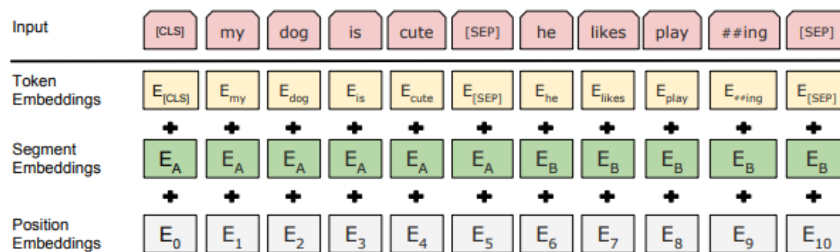


Figure 2.9: BERT input representation. Taken from (Devlin et al., 2019)

Next, the pre-training step comprises two objectives, which are Masked Language Modelling (MLM) and Next Sentence Prediction (NSP). The idea behind MLM is that some words of the input text are masked with a special token “ $[MASK]$ ” and the model has to recover the masked tokens. Consider the example, “I love to read books on the topic of affective computing”, in which we can replace the word “affective” with the mask token and the model is then trained to predict the missing word. The authors of BERT experimented with some strategies to prevent the model from concentrating on specific positions that are masked (e.g., masking randomly 15% of the words). The other objective is NSP, which aims to learn the relationship between two inputs. As the name suggests, the task is to predict whether sentence A is followed by sentence B or not. By training the BERT model on these two objectives, it can handle many tasks due to the similarity of both objectives to many tasks in NLP. The fine-tuning step comes next after the model is pre-trained, which is straightforward and fast to execute. Figure 2.10 presents how BERT can be utilised for text classification. The output of the $[CLS]$ is fed into a classifier layer with softmax for multi-class classification or sigmoid for multi-label classification.

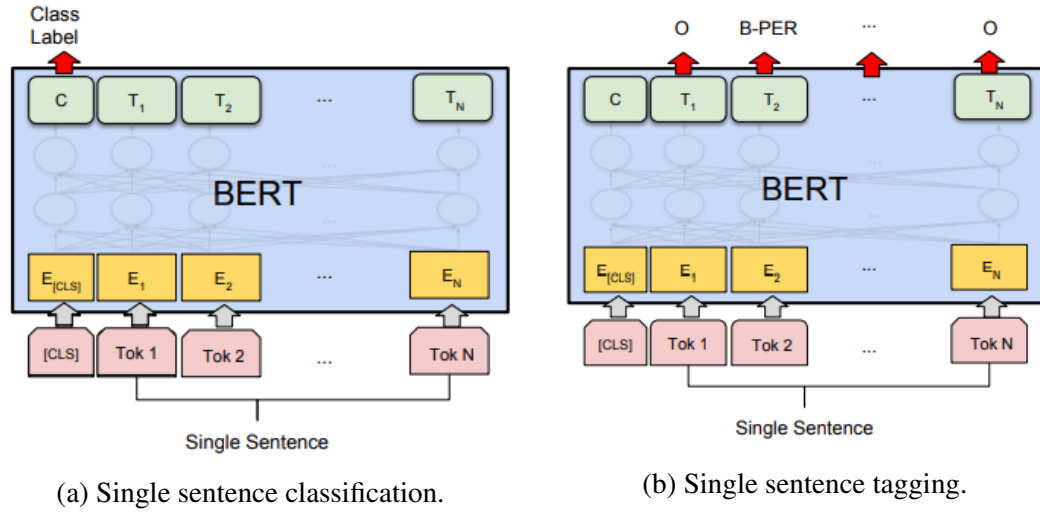


Figure 2.10: Text classification based on BERT. Taken from (Devlin et al., 2019)

2.3.6 Fine-tuning Methods

There are two strategies for adaptation in NLP research, i.e., fine-tuning and feature extraction. In the former strategy, the network's weights are unfrozen and then fine-tuned to a target task. In the latter strategy, they are frozen and then utilised directly for feature extraction without any further training on the target task. Several works have been focused on the fine-tuning strategy (Howard and Ruder, 2018; Devlin et al., 2019) as well as on the feature extraction (Peters et al., 2018c,a). Peters et al. (2019) compared the two strategies and provided some guidelines for NLP practitioners.

The concept of fine-tuning has become popular in NLP after the work of Howard and Ruder (2018) who proposed Universal Language Model Fine-tuning (ULMFiT). The core idea of ULMFiT involves three phases: i) pre-training a language model (LM) on a large corpus (i.e., Wikitext-103), ii) pre-training LM on target data, iii) fine-tuning LM on a target task. Some effective fine-tuning strategies are further introduced in the same work: i) “gradual unfreezing”, which focuses on fine-tuning each layer of the network independently and then fine-tuning all layers together. ii) “discriminative fine-tuning”, which tunes each layer with different learning rates. Another popular and widely-used model is BERT that also follows the same two-stage fine-tuning. More specifically, the BERT model is firstly trained on both BooksCorpus (Zhu et al., 2015) and English Wikipedia, which can then be fine-tuned directly on any target task. These two models have produced impressive results on a variety of NLP tasks, as well as contributing to many new studies.

The two-stage fine-tuning, also known as sequential fine-tuning, has shown to perform well on different tasks in NLP, including text classification (Felbo et al., 2017), story ending prediction (Li et al., 2019b), adverse drug reaction (Alhuzali and Ananiadou, 2019). Felbo et al. (2017) constructed a bi-directional LSTM with a self-attention mechanism using millions of emoji data, and then fine-tuned it to emotion, sentiment and sarcasm classification. In a similar vein, Alhuzali and Ananiadou (2019) developed a model based on LSTM with the self-attention mechanism that was then pre-trained on sentiment data. After that, the developed model was fine-tuned on Adverse Drug Reaction (ADR) corpora. The rationale for using sentiment data for ADR is that negative sentiment is frequently expressed towards ADR.

Additional studies further demonstrated the important role of pre-training on a related-domain corpus to the task under investigation, which can further boost the ability of the model and its performance (Garg et al., 2020; Sun et al., 2019; Phang et al., 2018). Gururangan et al. (2020) introduced two approaches (i.e., domain-adaptive pre-training and task-adaptive pre-training) that are similar to the ones discussed in the above studies. The Domain-Adaptive Pre-Training (DAPT) continued pre-training RoBERTa on a large corpus of unlabelled data from a related-domain, whereas the Task-Adaptive Pre-Training (TAPT) performed the same process, with the exception this time of pre-training on the task of interest. Both approaches demonstrated strong performance across four domains (i.e., biomedical and computer science publications, news and reviews) and eight classification tasks.

2.3.7 Deep Metric Learning

Deep Metric Learning (DML) is inspired by humans' visual system that enables them to identify similar objects and images effectively. In this respect, DML is a group of methods that attempt to measure the similarity between data samples, including objects and images. The main goal of DML methods are to group similar objects together, while separating dissimilar objects from each other. Popular DML methods include contrastive loss (Chopra et al., 2005), triplet loss (Schroff et al., 2015) and centre loss (Wen et al., 2016). Let us first define a set of samples \mathcal{X} and their respective labels \mathcal{Y} , which will be used to describe each of these methods briefly. Here, we aim to provide an overview of DML¹ methods that can help readers navigate easily through chapter 6.

¹The discussion in this section is based on the following [deep metric learning survey](#).

To explain contrastive loss, we sample two data points x_1, x_2 and their corresponding labels y_1, y_2 . Next, we need to use a distance (D) metric (e.g., Euclidean distance or Square Euclidean distance), which helps us measure whether the two data points are similar or dissimilar. A neural network is used to learn an embedding for each data point (known as feature extractor $f_\theta \in \mathbb{R}^d$), where θ represents the neural network weights, and d denotes the dimensional size. The neural network can be built based on any type of architecture discussed in this chapter (e.g., CNN, LSTM, BERT, etc.). The two points are then passed to the feature extractor $f_\theta(x_1, x_2)$, resulting in the embedding vector corresponding to each data point. Now, we can plug everything into the contrastive loss to estimate whether the two data points are similar or not as follows.

$$\mathcal{L}_{\text{contrast}} = \mathbb{1}_{(y_1=y_2)} \mathcal{D}_{f_\theta}(x_1, x_2) + \mathbb{1}_{(y_1 \neq y_2)} \max(0, m - \mathcal{D}_{f_\theta}(x_1, x_2)), \quad (2.27)$$

where the indicator function $\mathbb{1}\{condition\} = 1$ if the *condition* is satisfied, or 0 otherwise. \max is the hinge style loss and m is the margin. The margin here indicates that dissimilar data points beyond m do not contribute to the loss.

We can also extend our discussion to triplet loss. The triplet loss takes as input three samples instead of only pairs of samples, that are called “anchor (a), positive (p) and negative (n)”. Both anchor and positive samples share the same label, whereas the negative samples belong to a different label. Let us define x_a, x_p, x_n corresponding to some samples and y_a, y_p, y_n corresponding to the samples’ labels. Figure 2.11 illustrates the main idea of triplet loss, from which we can observe its objective in pulling the positive sample as close as possible to the anchor sample, while simultaneously pushing the negative sample as far away as possible from the anchor sample.

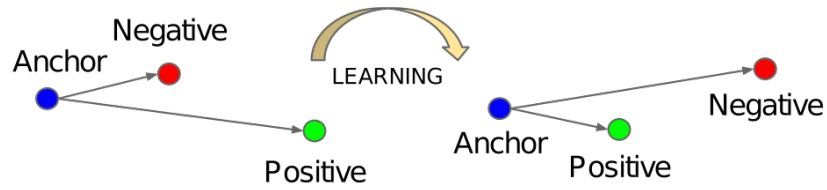


Figure 2.11: Illustration of Triplet loss. Taken from (Schroff et al., 2015).

In this respect, it learns an embedding space that minimises the distance between an anchor sample and a positive sample, while increasing the distance to a negative sample by at least a margin m . The triplet loss can be computed as in Equation (2.28). Again, we can still use any type of feature extractor network as well as distance metric. In order to make triplet loss work effectively, we need to leverage what is known as

“triplet mining” aimed at finding a triplet of x_a, x_p, x_n that satisfies $\mathcal{D}_{f_\theta}(x_a, x_n) < \mathcal{D}_{f_\theta}(x_a, x_p) + m$, which is often quite hard and task-dependent.

$$\mathcal{L}_{\text{triplet}} = \max(0, \mathcal{D}_{f_\theta}(x_a, x_p) - \mathcal{D}_{f_\theta}(x_a, x_n) + m), \quad (2.28)$$

To address the challenges of triplet loss in identifying such triplets, [Wen et al. \(2016\)](#) introduced the “centre loss”. This loss does not require comparisons between any samples, but instead it defines a centroid per label. The objective of this loss function is then to pull samples as close as possible to their corresponding centroid. The authors opt to train it with cross-entropy loss to achieve accurate centroids, especially at the beginning of the training phase. Equation 2.29 shows the computation of centre loss, which is much easier to compute than triplet loss.

$$\mathcal{L}_{\text{centre}} = \mathcal{D}_{f_\theta}(x_i, c_{y_i}), \quad (2.29)$$

Finally, some previous works have managed to improve the objective of triplet loss and centre loss, by building on the same objective of triplet loss, but at the same time avoiding its complexity in finding the triplets ([Cai et al., 2021](#); [Li et al., 2019a](#); [He et al., 2018](#); [Wang et al., 2017](#)).

2.4 Summary

In this chapter, we intended to provide an overview of NNs that are related with our methodology. More specifically, We began the description with some basic background of the building block of NNs, and then elaborated on how they can be trained. Next, we discussed common NNs, e.g., CNN, LSTM, Attention Mechanisms and BERT. We concluded the description with fine-tuning and deep metric learning methods.

We briefly highlight which types of NNs were used in this thesis. In chapter 4, we used BERT and a FFN to develop “SpanEmo”. In chapter 5, we further utilised some of the introduced networks in this chapter (i.e., BERT, LSTM, Attention Mechanism and FFN) to develop a neural model. In chapter 6, we finally used deep metric learning, BERT and CNN to develop our proposed Variant of Triple Centre Loss (VTCL). We will elaborate on each developed neural network in the corresponding chapter.

Chapter 3

Emotion Recognition: Background

What people think and how they feel serve as important information in understanding their decision making. “What do you think of the Spirit Untamed” movie?, “What is your opinion of the new Huawei p40”? or “Do you have any preferences for a specific type of food/restaurant?”, are just a few questions one would encounter in everyday discussion and communication. Answering these questions on a large scale can improve decision making, as well as helping people decide which movies to watch, phones to purchase, restaurants to visit, etc. In this thesis, we investigate the area of Textual Emotion Recognition (TER), which is concerned with the analysis and classification of a piece of text into discrete emotions (e.g., anger, joy and sadness, among others). Figure 3.1 provides an illustration of a TER model that takes an input text, processes it and then predicts some potential emotions.

We now turn to describing the task in greater detail, including terms related to

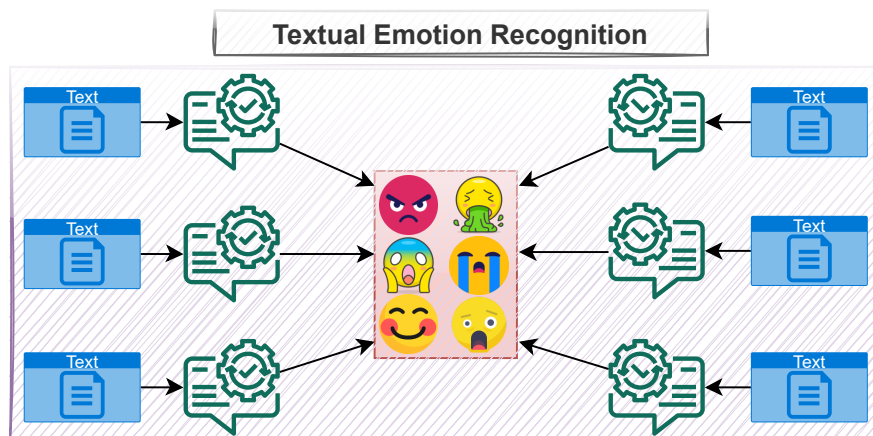


Figure 3.1: Illustration of TER.

emotion, how emotion is expressed in text, models of emotion (also known as theories of emotion), existing emotion corpora, common approaches to TER and evaluation metrics. Finally, we conclude the discussion with some observations and limitations of prior research.

3.1 Problem Definitions

3.1.1 Task

Emotion recognition selects the most appropriate emotion class $e_i \in E$ for an input instance x (i.e., a tweet or an example) in the case of multi-class classification or chooses more than one emotion in the case of multi-label emotion classification. $E = \{e_1, e_2, \dots, e_k\}$ corresponds to a set of emotion classes, where k denotes the number of emotions and $x = \{w_1, w_2, \dots, w_n\}$, with n representing the number of words in x . We make use of the categorical model¹ and the number/type of emotions may vary from one corpus to another. More specifically, the categorical model consists of discrete emotion classes, such as *joy*, *anger*, *disgust*, among others.

3.1.2 Word-Emotion Association

Mohammad and Turney (2013b, 2010) created the NRC Word-Emotion Association Lexicon (also called EmoLex), deriving associations between a set of emotions and more than $14k$ words. The NRC lexicon associates words with multiple emotions by using a binary value, where one means there is association between the given word and emotion class, and zero means there is no association. For example, the word “reject” is associated with *anger*, *fear*, *sadness* and *negative*, showing that this word can be used to express any one of these four emotions or a combination of two emotions when expressed in text. Consider the example, “well my day started off great the mocha machine wasn’t working @ mcdonalds.”, which contains clue words like “great” which is more likely to be associated with “joy”, whereas “wasn’t working” is more likely to be associated with negative emotions. This can be observed clearly via the ground truth labels (i.e., *anger*, *disgust*, *joy*, *sadness*) assigned to this example, where the first part of this tweet only expresses positive emotions (e.g., *joy* and *love*), while the other part expresses negative emotions. In this thesis, the term association refers to learning a mapping function between each word in the input and the set of emotions, where the

¹We will discuss what the categorical model is in Section 3.4.1.

mapping function determines the value of association between the two. The overall goal is to find words that are more associated with one emotion over another.

3.1.3 Emotion Correlation

We define the concept of correlation based on the type of emotion corpus, whether it is a single-label or multi-label corpus. For the multi-label case, the concept of correlation is obvious because it aims to learn correlations between emotion classes that share a similar emotion space (Wang and Zong, 2021), such as anger, disgust and sadness. However, the concept of correlation has not been studied for the single-label emotion case because each input is only labelled with one emotion class. We describe the concept of correlation for each case below and how we specifically define it for the single-label case.

Single-label emotion corpora allow each input instance to be labelled with one emotion, which limits textual emotion recognition models from taking advantage of multiple emotion annotations. However, correlations can be modelled between labels, as well as between instances. In order to capture correlations for the single-label case, we model correlations at the instance-level. The reason for this is that instances labelled with the same emotion class are more likely to have similar patterns and hence should be considered more highly correlated than those labelled with different emotions. Thus, we define examples sharing the same emotion class as “intra-class”, while examples belonging to different emotion classes are defined as “inter-class”. This definition is sensible for the following reasons: i) The single-label case holds the assumption that all labels are independent from each other. ii) Disentangling between negative emotions in the single-label case can also lead to better performance since they may be confused with each other in certain expressions. Consider the example, “I love you so much and *i am* [trigger_word] because you do not know that i exist.”, which contains both positive and negative emotion keywords although it is more negative oriented. The use of the word “love” can mislead the model to select the “joy” class over “sadness”. iii) TER models are more likely to benefit from such information, which in turn improves their prediction capability.

For the **multi-label** case, correlations are often modelled at the class-level since the data allow each input to have multiple labels. In this respect, emotion correlations are indispensable for multi-label emotion classification. This can be attributed to the fact that emotion classes are not semantically independent; a particular emotive expression can be associated with one or multiple emotions (Zhang et al., 2018; Deyu et al.,

2016). [Mohammad and Bravo-Marquez \(2017a\)](#) also observed the notion of negative emotions being highly correlated with each other, while less correlated with positive emotions. The high correlations issue between certain emotions can also be attributed to the lack of explicit emotion-based keywords as well as their inter-connection in linguistic expressions. Consider the example, “I’m doing all this to make sure you are smiling down on me bro.”, which is labelled with three highly correlated emotion classes (i.e., *joy*, *love*, *optimism*). The same idea has also been mentioned in psychological theories of emotion (e.g., Plutchik’s Wheel of Emotion). According to the wheel of emotion, “joy, love and optimism” are close to each other.

3.1.4 Emotion Labels

This section defines emotion classes used in this thesis to study textual emotion recognition. For each term, we draw inspiration from [Mohammad et al. \(2018\)](#) who proposed the SemEval-2018 multi-label emotion corpus that was labelled with eleven emotions, as discussed below. We also use emojis just as an approximation to represent the respective emotion, but they are not exhaustive. The emojis are derived from the work of [Shoeb and de Melo \(2020\)](#) who examined the association between 1200 emojis and emotions. It should be mentioned that we used the eleven emotions in Chapter 4, whereas a subset of them was utilised in Chapter 6, based on Ekman’s six basic emotions. This is because existing emotion corpora do not often follow the same taxonomy in terms of classifying and categorising emotions in text.

- **Anger** 🤔 : A strong feeling of displeasure, annoyance or rage.
- **Disgust** 🤢 : A strong feeling of disinterest, dislike, loathing and something unpleasant.
- **Fear** 🙏 : A feeling of apprehension, anxiety, concern, terror, fear or worry.
- **Joy** 😊 : A feeling of pleasure and happiness, including serenity and ecstasy.
- **Love** ❤️ : A strong positive emotion of regard and affection.
- **Optimism** 🙌 : Hopefulness and confidence about the future or the success of something.
- **Sadness** 💔 : Emotional feeling of pain, sorrow, pensiveness and grief.
- **Anticipation** 👁️ : Emotional feeling of interest and vigilance.

- **Pessimism** 🙄 : A strong feeling of cynicism and lack of confidence.
- **Surprise** 😲 : Feeling astonished, startled, distracted or amazed by something unexpected.
- **Trust** ¹⁰⁰ : Emotional feeling of acceptance, liking, and admiration.

3.2 Related Terms

This section aims to discuss some of the viewpoints regarding the concept of emotion and other related terms, such as affect and sentiment, which are used interchangeably. It is worth noting that there is still no consensus on what emotion is and which classification scheme should/should not be used. However, this section attempts to better distinguish emotion from other related terms and then report on how emotion is conceptualised in text. As [Izard \(2010\)](#) pointed out in his interviews with emotion scientists and theorists, there is still no consensus on the meaning of emotion. Having no clear meaning for the term emotion is not the problem, but having many different meanings, is.

3.2.1 Emotion and Subjectivity

The term subjectivity is often contrasted with the term objectivity, where the former corresponds to the expression of feelings, views, opinions, allegations, desires, suspicions, speculations or beliefs, while the latter corresponds to the expression of factual information ([Liu, 2012](#); [Riloff et al., 1993](#)).

Examples (Discussed ↓):

S1. Huawei phones are amazing because of their cheap price.

S2. Huawei is a technology company.

To give an example of the two terms, *S1* represents the subjectivity term because it expresses some opinions about Huawei phones, whereas *S2* is a fact in that it does not express anything specific about the company. In sentiment analysis, there is a first-step task called subjectivity classification which aims to classify an input as either subjective or objective. It is worth noting that a subjective expression may not express sentiment (i.e., positive or negative sentiment). Consider the example, “I think she is busy with something”, which is a subjective expression, but does not reveal any specific

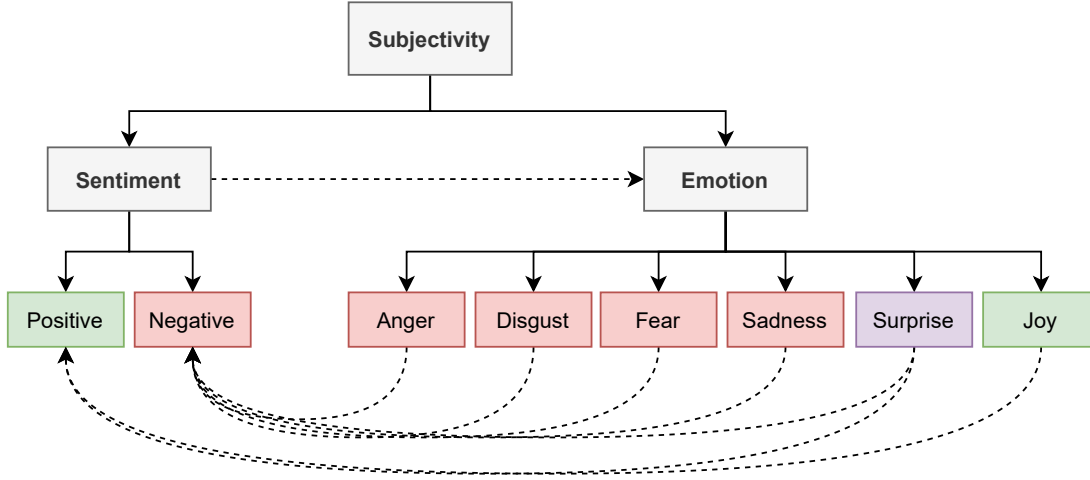


Figure 3.2: Taxonomy of subjectivity, sentiment and basic emotions. Dashed lines indicate that those emotion classes can be possibly part of the same valence space (i.e., positive or negative).

sentiment. The task of subjectivity classification is also relevant to the task of emotion classification in the sense that some previous works started by classifying the input as it expresses either emotion or no-emotion (Alhuzali et al., 2018a; Ghazi et al., 2010; Aman and Szpakowicz, 2007). After the input expressing emotion is determined, it is then classified into fine-grained emotions. We can conclude that the subjectivity term is an umbrella term for both sentiment and emotion. Figure 3.2 presents a taxonomy of subjectivity, emotion and sentiment, where both sentiment and emotion include subjectivity expressions. Sentiment analysis is more of a coarse-grained task, while emotion recognition is a fine-grained task. In this respect, emotion as a task can be connected to sentiment. The same applies to emotion labels, where negative emotions are linked to the negative sentiment class, while positive emotions are linked to the positive emotion class. It is worth mentioning that “surprise” can express both negative and positive feelings. Because of this, some prior research considered this class as an ambiguous emotion that is neither positive nor negative (Demszky et al., 2020).

3.2.2 Emotion and Sentiment

We discuss the differences between subjectivity and the emotion above and now turn to describing the relationship between emotion and sentiment. The task of sentiment analysis tends to differentiate positive content from negative and in some cases, the neutral class is added to determine subjective expressions that are neither positive nor negative. In this respect, it is coarse-grained in comparison to emotion classification,

which can involve fine-grained emotions. Some previous studies attempted to build emotion corpora with fine-grained set of emotions. For example, both [Liew et al. \(2016a\)](#) and [Demszky et al. \(2020\)](#) created emotion corpora labelled with 28 emotions, but each followed a different scheme and categorisation for emotions.

In addition, [Munezero et al. \(2014\)](#) examined the differences between sentiment and emotion. The authors reported that sentiment is defined as an evaluation, attitude or opinion towards a certain object or situation. [Cambria et al. \(2017\)](#) and [Liu \(2012\)](#) also described sentiment as the “underlying feeling, attitude, evaluation, or emotion associated with an opinion”. The above definitions highlight that sentiment is often expressed towards an object, entity or a topic. Another aspect, which makes a clear distinction between emotion and sentiment, is that emotion can last a short period of time, whereas sentiment can last a long period of time. A further line of work claimed that sentiment is expressed by individuals based on previous experience, beliefs or external influences, while emotion is triggered by a reason/cause/stimulus ([Cambria et al., 2017](#)).

Examples (Discussed ↓):

S3. I don’t have a Lexus car, but I think it looks amazing and cool.

S4. I purchased a Lexus car, but I do not feel happy anymore because its maintenance is super expensive.

The first sentence *S3* illustrates an instance of sentiment expression, where the author of this sentence expressed her/his opinion about Lexus cars being cool and amazing. Suppose that she/he purchased a Lexus car, the likelihood of being positive/happy would be high, but this is still an opinion and may change after owning the car. This can be clearly observed in the second sentence *S4*, which expressed a negative emotion (i.e., feeling not happy) about the Lexus car being very expensive to maintain. On the one hand, we can observe from the above examples that emotion is often caused by a stimulus representing why someone felt that way. On the other hand, sentiment does not require that kind of stimulus and could just be attributed to some external influences, like in the case of *S3*, where the author might have a relative/friend/colleague owning a Lexus car and based on that she/he came to the opinion that Lexus cars are amazing and cool.

3.2.3 Emotion and Affect

Although emotion and affect are used interchangeably in the literature, there are specific properties that can distinguish emotion from affect. Affect is an umbrella term that encompasses all topics related to emotion, feeling and mood (Cambria et al., 2017; Russell, 2003). In this respect, it is a more of an abstract concept, something that the more complex emotion builds upon. According to the Merriam Webster dictionary, the term affect is defined as “a set of observable manifestations of an experienced emotion”. This regards affect as what drives the resulting emotion. Russell (2003) also defined affect as a feeling evident in moods and emotions. The distinction here is that such a feeling is not directed, but primitive. For instance, suppose you are walking in the forest and suddenly see a bear, you are more likely to feel afraid. This may lead to screaming, crying, and running, which are the possible potential emotions in the given situation. Therefore, emotion is the display of affect, as well as the label given to the affect of the experienced emotion. Liu (2012) describes emotion as a “compound feeling concerned with a specific object, such as a person, an event, a thing or a topic”, which tends to be strong and to last a short period of time. This definition distinguishes emotion from affect as it is directed at something and is not primitive. Emotion is also accompanied by some reactions (e.g., physiological/physical changes).

3.3 How Emotion is Reflected in Text?

In this section, we focus on describing how emotion is expressed in text. Figure 3.3 illustrates the process that started with some emotional experience up until the understanding of the expressed emotion by the perceiver, which is often called “emotion perception”. Based on this illustration, one can define emotion as an expression that contains descriptions of emotion reactions formed in written text. This enables us to distinguish the concept of emotion from other general concepts and it also provides us with a mechanism to link it with the works presented in this thesis.

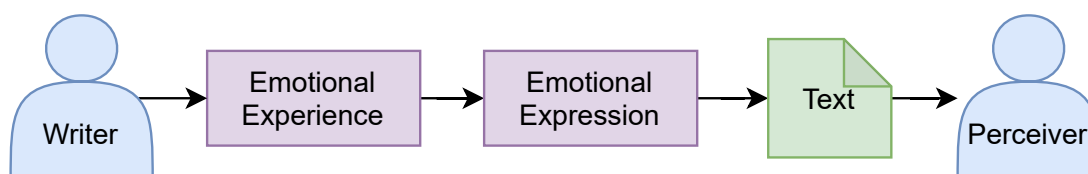


Figure 3.3: Emotion communication in text.

Writer is an individual who feels an emotion. Some individuals understand their feelings and can verbalise them in some way or another, while others may find it difficult to do so. This stage is also known as an emotional state, referring to the individual who feels an emotion. The emotional state can only be observed by the same person, but not by other people since it has not yet been shared with others.

Emotional experience corresponds to the situations, events or activities that are experienced by the person. In this respect, the emotional experience is the outcome of the outside world on the person who experiences the situation and how she/he understands it. This is relevant to the above point which highlights that emotion is often the result of stimulus. At this stage, the person, who feels the emotion, becomes aware and conscious of her/his emotion.

Emotional expression is a behaviour that describes the emotional state or feeling to the outside world, including friends, family etc. Emotional expression can be communicated via verbal and non-verbal cues. When verbal cues are used to express emotion in text, it is often the case that they describe a person's emotional state or someone else's. The person decides how much of her/his emotional state is to be shared with others. At this stage, the user can transmit her/his emotional state with some verbal cues (e.g., a word, phrase, sentence or document).

Text includes emotional cues utilised by the person who writes the text. Emotional cues depend on the choice of words, phrases or clauses that occur in the written text. This can be extended to the new channels of today's communications like social media, in which people can use emoticons and emojis to express their opinions and emotions. Consider the example, "I love the topic of affective computing 😊", which contains the word "love" as well as the smiley emoji. Both the word and emoji can be used as an emotion class or can be utilised to describe emotion classes, e.g., joy and excitement. On the other hand, there are cue words that trigger an emotional state like *accident* and *failure*, which can cause the "sadness" emotion. Emotional expression in text is influenced by different factors, including negation, syntax, intensifiers, models, and more importantly context, among others.

Emotional perceiver is the perceiver of the emotional expression. He looks for cues that aid the identification of the emotional state reported in text. Humans through previous experiences and knowledge of emotion can recognise emotion in text to some extent, depending on how much the text contains clear signals. When enough emotional cues exist in text, it becomes easier for the perceiver to understand and interpret the emotional expression. For example, if someone writes, "I am so happy because

my paper has been accepted by a major venue”, the perceiver of such expression can conclude that the person certainly expresses joy.

3.4 Models of Emotion

Understanding models of emotion is crucial for TER as they are the main source for determining how emotions should be categorised/classified. The classification of emotion is based on two widely used models of emotion: i) the categorical model, and ii) the dimensional model. These two models are derived from theories of psychology, which classify emotions into taxonomies. We now turn to describing each one of the two models in greater detail.

3.4.1 Categorical Model

The categorical model defines emotions based on discrete categories, and each category represents a distinctive emotion concept. An emotion concept is used to represent a set of similar/associated terms. For example, the emotion label ”joy” is associated with different terms to describe someone feeling positive/happy (e.g. glad, pleased and joyful). The basic emotion taxonomy is part of the categorical model, which organises emotions based on discrete classes/categories (Ekman, 1992; Plutchik, 1980, 1984). Plutchik (1980) proposed the wheel of emotion theory consisting of eight primary emotions, including (joy, anger, disgust, fear, sadness, surprise, trust and anticipation). The wheel is arranged in three circles, where each one represents different degrees of emotion intensity; emotions intensify as they move from the outer to the centre of the wheel that is also represented by colours. In other words, the darker the colour, the more intense the emotion is, whereas the lighter the colour, the less intense the emotion is. The middle circle corresponds to what is known as the primary eight emotions. The wheel shows that each primary emotion has a polar opposite emotion, e.g., the polar opposite of joy is sadness. There are also complex emotions that are identified based on a combination of two close emotions. For example, ”love” is composed of two emotions (i.e., joy and trust). Another popular emotion taxonomy is based on Ekman’s basic emotions, which are also a subset of Plutchik’s list, excluding trust and anticipation. Both theories have been widely used to study emotion expression in the field of natural language processing (Klinger et al., 2018c; Alhuzali et al., 2018a; Mohammad and Turney, 2013b; Mohammad, 2012b).

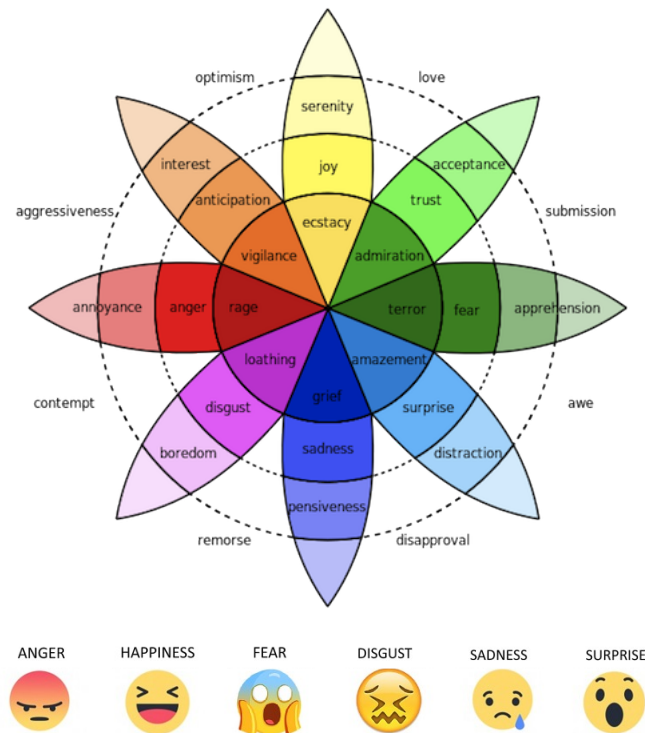


Figure 3.4: Categorical of emotions: Plutchik's (top) and Ekman's (bottom).

Figure 3.4 presents the two mentioned categorical models of emotions (i.e., wheel of emotion and Ekman's basic emotion, respectively). The categorical model is straightforward and easy to understand as well as flexible in that terms corresponding to each emotion category can be selected. However, such flexibility may cause some confusion due to the closeness of certain emotions, which require further assessment of the criteria for emotion selection. There² are a few approaches to TER that use the categorical model. Figure 3.5 shows the three common approaches to TER. The simplest type of approach aims to assign a single emotion category to an example (Islam et al., 2019a; Xia and Ding, 2019; Alhuzali et al., 2018c,b; Agrawal et al., 2018; Saravia et al., 2018; Felbo et al., 2017; Abdul-Mageed and Ungar, 2017), whereas the more challenging type of approach allows multiple emotion categories to be assigned to the same example (Fei et al., 2020; Xu et al., 2020a; Ying et al., 2019; Zhou et al., 2018; Baziotis et al., 2018; Yu et al., 2018; He and Xia, 2018). However, a more interesting and even more complex type enables multiple emotions, with the addition of assigning each emotion category with an intensity value that determines how much a given emotion category is associated with this example (Zhang et al., 2018; Zhao and Ma,

²We will discuss the citations used in this paragraph in greater detail in Section 3.6.

2019). The third type is known in NLP literature as “emotion distribution learning” firstly proposed by Zhou et al. (2016) and it has not received as much attention as the above two types because of its difficulty as well as the lack of emotion corpora labelled with distribution information.

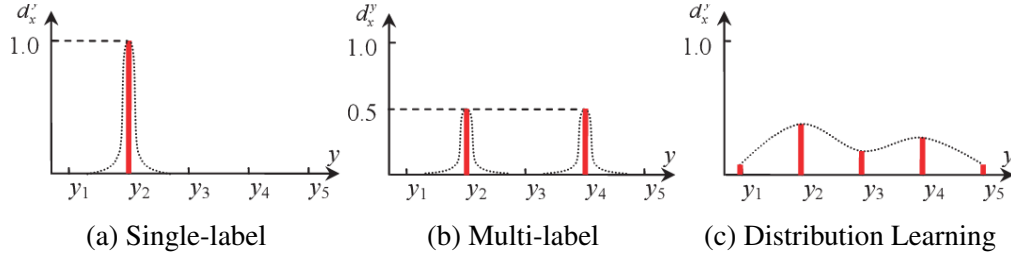


Figure 3.5: Illustration of the different learning schemes for TER. Source (Geng, 2013)

3.4.2 Dimensional Model

The dimensional model identifies emotions in a two or three-dimensional space, with the most commonly used dimensions being valence, arousal and dominance. Russell and Mehrabian (1977) constructed “Valence-Arousal-Dominance”, also known as PAD/VAD, which is a semantic model that rates emotion on three-dimensional spaces. Russell (1980) further introduced the Circumplex model of affect that maps emotions into two dimensional space of valence (positiveness-negativeness) and arousal (active-passive), and it is shown in Figure 3.6. The circumplex model establishes that emotions are not independent but interconnected, i.e., they can be recognised by a composition of both the valence and arousal dimensions. Several studies in NLP research have utilised the dimensional approach to develop emotion recognition models (Mohammad, 2018; Warriner et al., 2013), to create emotion lexicons (Zhu et al., 2019; Park et al., 2019; Akhtar et al., 2019) or to build emotion corpora (Buechel and Hahn, 2017; Preotiuc-Pietro et al., 2016). The former identifies emotion in text based on the valence, arousal and dominance dimensions, whereas the latter curates annotation of VAD annotation to words (e.g., anger, joy, fear, etc).

3.5 Datasets & corpora

Textual emotion recognition has received a great deal of attention in the last two decades, for which several emotion corpora have been created. Some of them focus

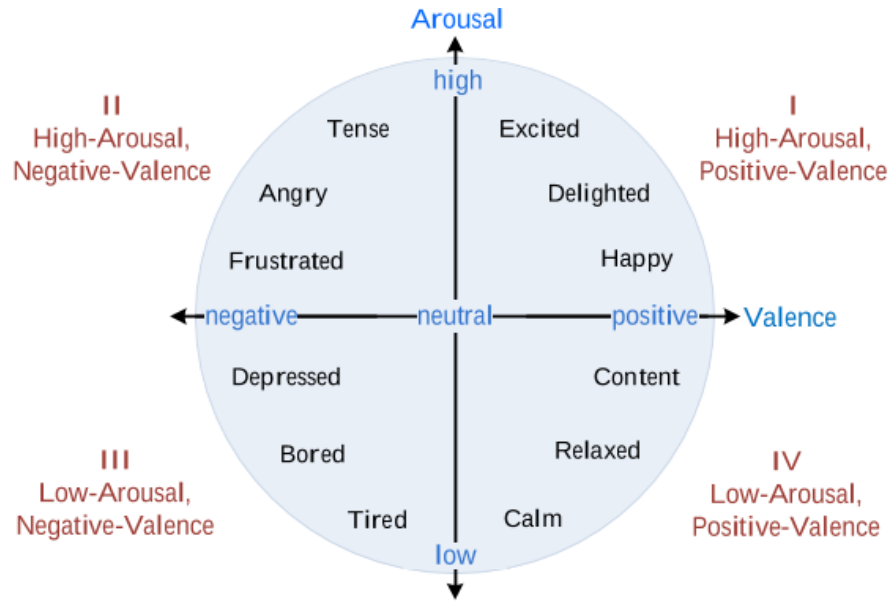


Figure 3.6: Russell's circumplex model of emotion

on general domains (i.e., news, events, blogs and stories), while the majority of recent published corpora focus on social media (i.e., Twitter, Facebook and Reddit). Some corpora have been created manually by humans, known as gold annotated corpora. In the last decade, distant supervision (DS) has been extensively utilised for emotion resource creation due to its advantages in building large emotion corpora automatically without requiring any human effort with regard to data annotation. In addition, emotion corpora are mostly collected from social media because of the availability of large volumes of data, which is user-generated content. Social media data also possess specific characteristics, such as informal language, misspellings, symbols, and abbreviations, which make the task of TER challenging as well as interesting. It is worth noting that emotion corpora collected from general domains other than social media do not pose such challenges.

Table 3.1 presents the timeline of emotion corpora collected from different sources (i.e., news, blogs, events' description and social media). The number of publications of emotion corpora after the year 2010 has clearly increased as shown in Table 3.1. Another interesting observation is that distant supervision has become quite popular during this period and has allowed the creation of large emotion corpora (e.g., Bipolar-Emo., HarnessingT., EmoNet and CARER). The number of corpora collected from social media has also increased dramatically. This can be attributed to the nature of language used on social media platforms, which enables the study of TER at scale as

well as being an easy-way for gathering data based on the so-called “self-labelling”. This term refers to the use of emotion-cue keywords that are utilised by users, such as hashtags, emojis and emoticons. An instance of a hashtag, an emoticon and emoji is *#happy*, :) and 😊, respectively. These symbols have been used extensively and continue to be used on social media. In the past five years, we have started to observe more corpora containing sets of emotions, the labels of which are both greater in number and more diverse.

Dataset	Pub.Y	Gran.	#Exp.	#Lab.	Task	Lang.	Avail.	Anno.
ISEAR	1994	Des.	7,665	7	MC	En	Y	Gold
Tales	2005	Sent.	15,302	6	MC	En	Y	Gold
Affective	2007	Head.	1,250	6	MC/Dis	En	Y	Gold
TEC	2012	Tweets	21,051	6	MC	En	Y	DS
HarnessingT.	2012	Tweets	2,488,982	7	MC	En	N	DS
Bipolar-Emo.	2013	Tweets	3,041,952	4	Bin	En	N	DS
EmoTweet	2016	Tweets	15553	28	MC/ML	En	N	Gold
EmoNet	2017	Tweets	1,608,233	24	MC	En	N	DS
CBET	2017	Tweets	76,860	9	MC	En	Y	DS
EmoInt	2017	Tweets	7,097	4	Reg	En	Y	Gold
EmoBank	2017	Sent.	10,548	3D	Reg	En	Y	Gold
SemEval-Ec	2018	Tweets	10,983	11	ML	En, Ar, Es	Y	Gold
IEST	2018	Tweets	191,731	6	MC	En	Y	DS
CARER	2018	Tweets	562,002	8	MC	En	N	DS
GoEmotion	2020	Posts	58,009	27	MC/ML	En	Y	Gold

Table 3.1: Available Emotion Recognition Datasets. *Pub.Y* refers to the year of publication, *Gran.* refers to granularity, *#Exp.* refers to the number of examples, *#Lab.* refers to the number of labels, *Lang.* refers to language, *Avail.* refers to the availability of the corpus and *Anno.* refers to the type of annotation (i.e., Gold or Distant Supervision “DS”). *Des.* stands for description, *Sent.* stands for sentence and *Headl.* stands for headline. *MC* stands for multi-class classification, *ML* stands for multi-label classification, *Bin* stands for Binary classification (i.e., one-vs-one), *Reg* stands for regression and *Dis* stands for distribution (i.e., assigning intensity values $\in [0, 1]$ to all emotion classes). *Y* means the data is available for download, while *N* means it is not available. *3D* refers to the valence, arousal and dominance dimensions.

Table 3.1 also discusses different emotion corpora in terms of text granularity, classification, language, availability and annotation. The number of samples and emotion classes are also included in Table 3.1. One of the first created emotion datasets was ISEAR, which stands for “International Survey on Emotion Antecedents and Reactions” (Scherer and Wallbott, 1994). This corpus consists of 7,665 sentences, where each sentence is annotated with a single category of emotions based on five categories of Ekman (i.e., joy, anger, sadness, fear and disgust) and two additional categories

(i.e., shame and guilt). The dataset is acquired from questionnaires based on descriptions of people’s experiences who have different cultural backgrounds. The goal was to determine if such emotional experiences are universal or specific across cultures. Subsequently, [Altman \(1991\)](#) built an emotion corpus of 15,302 sentences from 185 children stories annotated with one of Ekman’s six basic emotions, with the exception of dividing the surprise class into positive and negative *surprise*. In addition, [Strappavara and Mihalcea \(2007\)](#) created an emotion dataset for the SemEval affective-Text shared task, annotated 1,250 news headlines with one of Ekman’s basic emotions. This dataset includes distribution information across all the six emotion classes, where each class is assigned with an intensity value to indicate its association with the headline.

Unlike the above-mentioned corpora, social media have become increasingly popular since 2012 as shown in Table 3.1. A couple of emotion datasets were created in 2012 (i.e., TEC and HarnessingT.). Both [Mohammad \(2012b\)](#) and [Wang et al. \(2012\)](#) gathered emotion data from Twitter using the same method of data annotation and creation, which is based on a list of predefined hashtags corresponding to each class of emotion (e.g., #joy, #glad, #sad, and #anger, among others). On the one hand, the former study collected a corpus of 21,048 tweets self-labelled by the users of such tweets via “hashtags”. The objective was to determine whether or not this method can be used as a surrogate for gathering emotion data automatically. On the other hand, the latter study followed the same method and collected a large emotion corpus to exploit the effectiveness of the size of training data on emotion classification. In order to increase the quality of data, the authors randomly sampled 400 tweets and labelled them manually with a tag from the set relevant and irrelevant, which are then used to filter out noisy data. [Suttlés and Ide \(2013a\)](#) also utilised distant supervision, but focused on creating an emotion corpus according to a set of eight basic bipolar emotions defined by the wheels of emotion. In addition to hashtags, the authors used emoticons and emojis for data creation. Next, [Abdul-Mageed and Ungar \(2017\)](#) followed-up the self-labelled method of hashtags and collected a large dataset (EmoNet) of emotion via distant supervision labelled with a diverse set of emotion classes equivalent to 24. In a similar vein, [Saravia et al. \(2018\)](#) built upon this work and created a large emotion datasets (CARER), but it was only labelled with 8 emotions, corresponding to the primary set of emotions defined in the wheel of emotion. Another emotion corpus called “Cleaned Balanced Emotion Tweets” (CBET) was developed by [Shahraki and Zaiane \(2017\)](#) utilising the same hashtag approach for collecting the data. The “Implicit Emotions Shared” Task (IEST) is an additional dataset created via distant supervision, but

the tweets were gathered using an expression of an emotion-keyword plus either “that, because or when” (Klinger et al., 2018a). The authors claim that this method can help capture the cause of the emotion from tweets.

Furthermore, a few emotion corpora have also been created through gold annotation and involved multi-label classification. Liew et al. (2016b) built an emotion corpus called “EmoTweet-28”, which contains 15,553 tweets labelled with one of 28 emotions as well as multiple emotions. The corpus also includes annotations for valence, arousal and emotion cues. Another widely-used multi-label emotion corpus was created by Mohammad et al. (2018) known as “SemEval-Ec” based on labelled data from tweets in three languages (i.e., English, Arabic and Spanish). The corpus is labelled with 11 emotions plus neutral, some of which are corresponding to Plutchik’s eight primary emotions. Recently, Demszyk et al. (2020) proposed an emotion corpus, known as “GoEmotions”, composed of 58K Reddit comments and manually labelled with one or more of 27 emotions plus neutral. The authors also include emotion grouping at two levels, where the first grouping involves mapping all the 27 emotions into a sentiment category (i.e., positive, negative, ambiguous and neutral), while the second one involves mapping them into Ekman’s six basic emotions. The rationale behind such grouping is based on their popularity in the NLP community. Although this corpus contains multi-label annotations, the majority of Reddit posts are labelled with a single emotion class (i.e., roughly around 83%), while the remaining posts are labelled with more than one emotion.

Moreover, a regression corpus was built for the task of emotion based on four classes of Ekman’s basic emotions, i.e., anger, joy, sadness and fear (Mohammad and Bravo-Marquez, 2017b). This corpus contains 7097 tweets with intensities corresponding to four emotions, and it aims to link each tweet with various intensities of emotion. Another corpus was introduced by Buechel and Hahn (2017) based on multiple genres and domains. It includes 10,548 sentences, each of which was manually annotated from the perspectives of both writer and reader. The annotation follows the dimensional model of emotion, from which the authors annotated sentences using valence (the polarity of emotion), arousal (the intensity of emotion) and dominance (the degree of control). It is worth mentioning that Klinger et al. (2018c) conducted an extensive analysis of some of the above-mentioned corpora as well as other existing emotion corpora. We now turn to discussing common approaches to TER.

3.6 Approaches to Textual Emotion Recognition

This section describes common approaches to TER based on our taxonomy as shown in Figure 3.7. The taxonomy consists of four categories, i.e., computational tools, features, learning and tasks, some of which are also discussed in [Deng and Ren \(2021\)](#), [Alswaidan and Menai \(2020b\)](#) and [Seyeditabari et al. \(2018\)](#). The first category is computational tools, which is divided into two sub-categories, i.e., traditional machine learning (ML) and neural networks. The first sub-category was popular before the revolution of neural networks and it was the main tool used for TER. However, neural networks have become the *da facto* tool for TER because they enable TER models to learn and extract features automatically. We described some of the popular neural networks in Chapter 2, whereas we illustrate in this chapter how they have been used for the detection and identification of emotion expressions. The second category is focused on features, which is divided into three sub-categories, i.e., corpus-based, knowledge bases and languages models. Each of which has its own advantages and disadvantages, but language models have been widely-used for TER. This can be attributed to the fact that they can learn complex linguistic phenomena and can produce high quality word representations ([Clark et al., 2019](#)). The third category is learning, which is divided into four sub-categories, i.e., lexicon-based learning, supervised learning, unsupervised learning and transfer learning. The majority of TER research is focused on supervised and transfer learning approaches due to their ability to achieve high performance on existing emotion corpora. Both supervised and transfer learning approaches can be divided into further sub-categories that have been extensively explored for TER. The fourth category is the tasks of TER, which are divided into three main tasks, i.e., multi-class classification, multi-label classification and distribution learning. We introduced each of these three tasks in Section 3.4. In the next few sections, we base our discussion on the third category (i.e., learning), in which we also describe the other categories.

3.6.1 Lexicon-based

Earlier studies focused on recognising emotion using lexicon-based approaches, which rely on seed words and their corresponding labels to identify emotions in text. The rationale is that emotion-bearing keywords convey emotional meaning, and hence they can describe a text in which they occur. A simple matching algorithm is commonly adapted to search for emotion keywords in text from an emotion lexicon. In the case

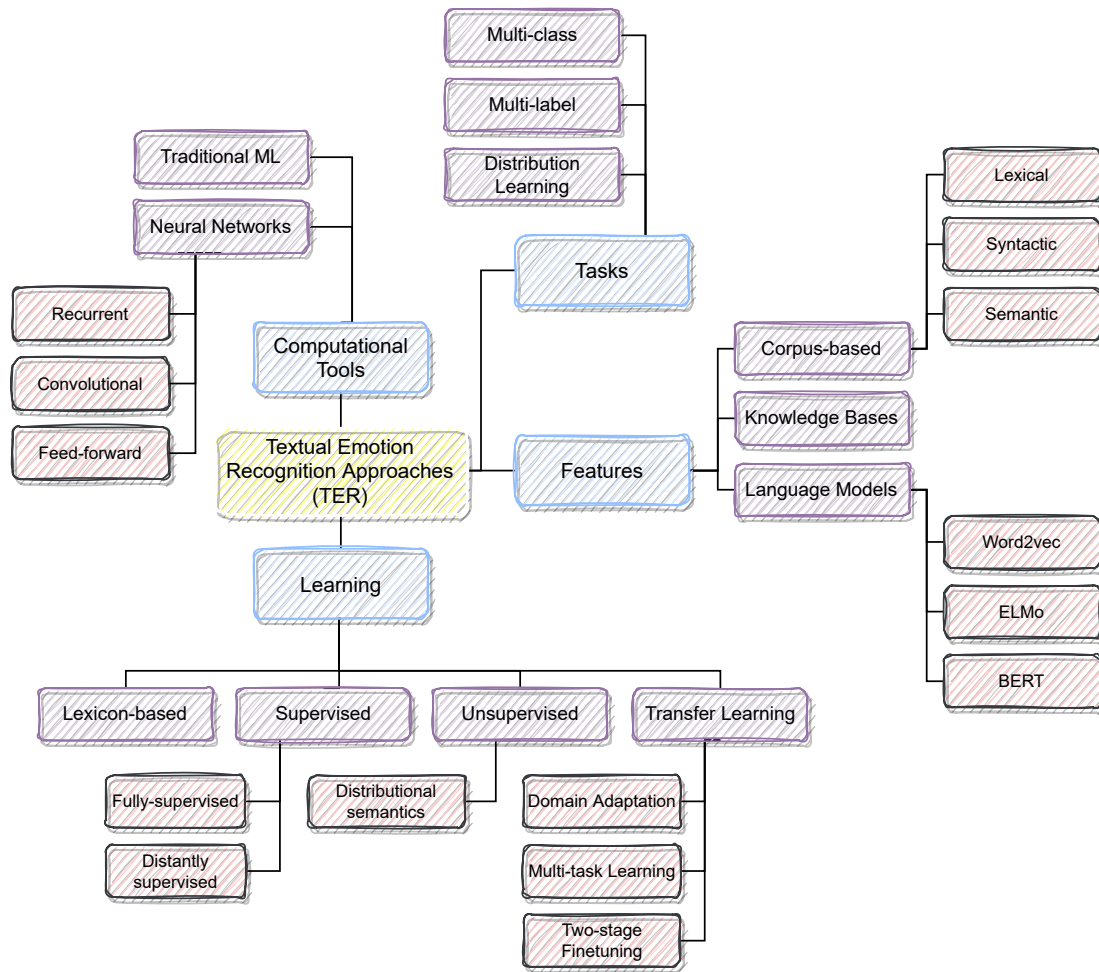


Figure 3.7: Taxonomy of Emotion Recognition approaches.

of multiple emotion keywords found in the text, a counts-based method is adopted, for which the goal is to select an emotion label with the highest count.

There are several emotion lexicons that are developed in the literature, such as WordNet-Affect (Strapparava et al., 2004), NRC (Mohammad and Turney, 2013b), LIWC (Pennebaker et al., 2015), NRC-VAD (Mohammad, 2018), among others. The WordNet-Affect lexicon is an extension of WordNet and contains affective concepts associated with affective words, covering 4,787 words. The NRC lexicon, on the other hand, consisting of 14,182 words, focused on the realisation that Words are associated with multiple emotions. For this reason, each word receives a binary score of zero and one. The former means that there is no association between the given word and emotion class, whereas the latter means an association exists between the two. The selected classes are based on Plutchik’s eight primary emotions plus negative and positive sentiment. Linguistic Inquiry and Word Count, known as LIWC, has 1393

words in the affective category. Finally, the NRC-VAD contains more than 20k English words that were manually annotated for valence, arousal and dominance. The method used to annotate VAD words is based on the so called “Best-Worst” scaling, which aimed at addressing issues related to annotation consistency as well as achieving fine-grained annotation scores. The authors claimed that this method produces more reliable annotations than in the past. Although the lexicon-based approach is easier to be interpreted, implemented and evaluated, it is time-consuming, context-free and domain-dependent, and has poor-coverage. More importantly, detecting emotion from text is not effectively achieved by simply adding up the emotional associations and their words. The above reasons make lexicon-based approaches ineffective for TER.

3.6.2 Supervised Learning

Supervised learning is focused on the development of computational methods that can be learned from labelled data. The labelled data are obtained via either human annotators or some heuristics based on predefined emotion classes (e.g., *joy*, *anger*, *sadness*, etc.). Although distant supervised approaches do not require human effort in terms of annotation, they are still created based on predefined noisy labels. That is why we also discuss distant supervision approaches in this section. The goal of developed supervised learning methods is to learn patterns associated with the predefined emotion classes from the given annotated or distantly created emotion corpus.

Feature-based Learning. Some works demonstrate the contributions of features derived from corpus-based texts (e.g., bag of words, n-grams, syntax, emoticons and punctuation features) and knowledge-based methods (e.g., WordNet-Affect, Roget’s Thesaurus and General Inquirer) by running binary classification experiments (Alm et al., 2005; Aman and Szpakowicz, 2007, 2008; Gupta et al., 2010). Alm et al. (2005) explored several corpus-based features with the Winnow linear classifier to identify emotion in Fairy Tales, while Aman and Szpakowicz (2007, 2008) exploited corpus-based and knowledge-based features with Support Vector Machine and Naive Bayes to classify emotion in blogs. Gupta et al. (2010) used n-grams and presence of words/phrases from specific dictionaries that are then used by Boostexter to recognise emotion in email.

Other works use multi-class classification experiments, in which a learner (e.g. linear classifier based methods) is trained on the features of labelled data to classify inputs into one label (Klinger et al., 2018c; Liew et al., 2016a; Mohammad and Kiritchenko, 2015b; Mohammad, 2012c,b; Wang et al., 2012). Klinger et al. (2018c) experimented

with a Maximum Entropy classifier (MaxEnt) with a bag of words features as a simple baseline for emotion classification across various emotion corpora, whereas [Mohammad \(2012c,b\)](#) used Support Vector Machine (SVM) and Logistic Regression to classify emotion in text based on n-gram and emotion lexicon features. [Wang et al. \(2012\)](#) applied two machine learning algorithms (i.e., LIBLINEAR and Multinomial Naive Bayes) to a large Twitter dataset collected via distant-supervision by using a list of hashtags to exploit the effectiveness of the size of training data on emotion classification. In a similar vein, [Mohammad and Kiritchenko \(2013\)](#) utilised hashtags to capture emotions from Tweets and demonstrated that hashtags were not only a strong indicator of emotion in tweets, but also produced consistent annotations to that obtained from trained annotators. [Purver and Battersby \(2012\)](#) constructed a distant supervision emotion corpus by using hashtags and emoticons, and they experimented with SVM by utilising a linear kernel and unigram features.

In contrast to the above studies, additional works conducted emotion classification experiments based on theories of emotion or some learning schemes (i.e., flat vs hierarchical). [Ghazi et al. \(2010\)](#) introduced a learning scheme which classified emotion expression as: 1) emotion vs no-emotion and 2) six basic emotions. The intent is to group these emotions and their relations into a hierarchy and then leverage it in the classification process. The developed method outperformed the often utilised flat classification scheme. Furthermore, [Suttles and Ide \(2013b\)](#) classified emotion in text according to a set of eight basic bipolar emotions by following Plutchik's wheels of emotion. The setup is quite similar to binary classification experiments, but the focus is on the opposite emotion pair (e.g., *joy vs sadness*).

Moreover, some studies used feature-based learning for multi-label emotion classification. [Badaro et al. \(2018\)](#) was one of the teams who participated in the SemEval-2018 shared task. The team proposed "EMA" using various pre-processing steps (e.g. diacritics removal, normalisation, emojis transcription and stemming), as well as different classification algorithms. Another team "Tw-StAR" applied similar pre-processing steps and then used TF-IDF to learn features of a Support Vector Machine for the same shared task ([Mulki et al., 2018](#)).

Even though standard machine learning is utilised extensively in the area of recognising emotion, their quality and coverage relies heavily on hand-crafted features, which are time-consuming and expensive. They also suffer from poor-coverage and cannot identify patterns beyond those features engineered. In general, the quality of

generated features depends on the expertise and understanding of the area under investigation. Due to these challenges, the focus of today's research in TER is on deep learning methods due to their ability to learn the features automatically.

Deep Learning. More recently, several neural network models have been developed for TER, obtaining competitive results on different emotion corpora. Some of these models generally focus on a single label emotion classification, in which only a single-label is assigned to each input (Islam et al., 2019a; Alhuzali et al., 2018a,c; Saravia et al., 2018; Zhang et al., 2018). Other models have also been proposed for multi-label emotion classification, in which one or more labels are assigned to each input (Huang et al., 2021; Fei et al., 2020; Xu et al., 2020a; Alswaidan and Menai, 2020a; Zhou et al., 2020; Gaonkar et al., 2020a; Ying et al., 2019; Fei et al., 2019; He and Xia, 2018; Baziotis et al., 2018). In addition, multi-label emotion classification was extended into what is known in NLP research as “emotion distribution learning”. This task focuses not only on selecting multiple emotions, but also on associating each emotion with an intensity value (Zhao and Ma, 2019; Zhang et al., 2018; Zhou et al., 2016). We now turn to describing each of the three approaches in greater detail.

The first approach is focused on multi-class emotion classification. Abdul-Mageed and Ungar (2017) proposed an emotion classification model developed using Gated Recurrent Unit (GRU) based on a large data collected via distant supervision from Twitter. Saravia et al. (2018) also followed the same principles of gathering data from Twitter, upon which contextualised affect representations were built and used as features for training various neural networks (e.g., GRU). Islam et al. (2019a) created a Multi-Channel-Cnn (MCC), where each channel was built to learn specific embeddings for each sample as well as additional features that occur in the same sample (e.g., emojis, emoticons and hashtags). The MCC was then evaluated against four emotion datasets (i.e., HarnessingT., TEC, CBET and EmoInt). Zhang et al. (2018) proposed a Multi-Task-Loss approach (MTL) that involved learning both emotion distribution and classification, based on a CNN network (Kim, 2014) trained with cross-entropy and Kullback-Leibler loss functions jointly. Next, this approach was tested on four emotion datasets (i.e., ISEAR, TEC, CBET and Tales). Fei et al. (2019) developed a neural network for implicit emotion detection. The proposed model involved firstly capturing the implicit sentiment objective as a latent variable using Variational Autoencoder and then incorporated it into an emotion classifier as prior information. This addition enables the network to make better predictions as well as to achieve strong performance over two benchmark datasets (i.e., ISEAR and IEST). All the above-mentioned models

are only applied to English. [Alhuzali et al. \(2018a\)](#) described an automatic data collection method for emotion classification in both Modern Standard and Dialectal Arabic based on Robert Plutchik's 8 basic emotions. The method exploited first person emotion seeds, where each consists of a phrase composed of the first person pronoun plus a seed word expressing an emotion (e.g., "I'm" + "happy"). This method is then evaluated on standard machine learning algorithms and Gated Recurrent Unit network.

The second approach is focused on multi-label emotion classification. With this approach, there are studies that aim to learn correlations between emotions to improve the task of multi-label emotion classification, and other studies that do not consider that. Both [He and Xia \(2018\)](#) and [Zhou et al. \(2018\)](#) considered integrating emotion correlations into the loss function and were evaluated on Chinese emotion datasets. More specifically, [He and Xia \(2018\)](#) introduced a Joint Binary Neural Network (JBNN) based on Long Short-Term Memory (LSTM) with self-attention mechanism, which focused on learning the correlations between emotions based on the theory of Plutchik's wheel of emotions ([Plutchik, 1980](#)). [Zhou et al. \(2018\)](#) also defined a ranking emotion relevant loss focused on incorporating emotion correlations into the loss function to improve both emotion prediction and rankings of relevant emotions. [Gaonkar et al. \(2020b\)](#) proposed an approach for multi-label emotion classification that takes advantage of the semantics of emotions. The approach used label embeddings that can track label-label correlations, which was also enhanced by a semi-supervised method that can regularise for the correlations on unlabelled data. The first two studies were evaluated on a Chinese emotion dataset, whereas the third study was tested on an English emotion dataset.

There are additional studies that capture emotion correlations through the network apart from the training objective. [Huang et al. \(2021\)](#) proposed a Sequence-to-Emotion approach (Seq2Emo) for multi-label emotion classification to tackle emotion correlations in a bi-directional decoder. The correlations were then utilised to recover emotion classes sequentially. [Fei et al. \(2020\)](#) introduced a Latent Emotion Memory network (LEM), in which the latent emotion module learns emotion distribution via a variational autoencoder, while the memory module captures features corresponding to each emotion. Moreover, [Xu et al. \(2020a\)](#) considered capturing emotion correlations through Graph Convolutional Network (GCN), which was then incorporated into both LSTM and BERT networks.

In contrast to the above-mentioned studies, the following studies are only focused

on the task of multi-label emotion classification and do not take into account emotion correlations. [Zhou et al. \(2020\)](#) proposed an Emotional Network (EmNet), which aimed at learning sentence emotions and constructing emotion lexicons that are dynamically adapted to a given context. The dynamic emotion lexicons are useful for handling words with multiple emotions based on different context, which can effectively improve the classification accuracy. [Ying et al. \(2019\)](#) also introduced domain knowledge into BERT for multi-label emotion classification, which led to strong performance. [Alswaidan and Menai \(2020a\)](#) proposed a Hybrid Neural Network (HNN) for multi-label emotion classification based on the use of different word embeddings (e.g. *Word2Vec*, *Glove*, *FastText*) plus variations of RNN variants. [González et al. \(2018\)](#) proposed “ELiRF” for multi-label emotion classification. The ELiRF model applied some pre-processing steps (e.g., normalising hashtags and emojis), while adapting the “TweetMotif” tokeniser ([O’Connor et al., 2010](#)) for Spanish tweets. The team then built their neural network model based on LSTM and CNN, which were further combined with several emotion and sentiment lexicons. All four works were evaluated on the same SemEval-2018 dataset, except for the first two that only used the English set, the third one that utilised the Arabic set and the fourth one that used the English and Spanish sets.

Finally, the third approach is based on emotion distribution learning. This approach extends multi-label emotion classification by learning intensity values of an input among a set of emotions. Although such a task is crucial for better emotion understanding and interpretation, there is limited research done in this area, which can be attributed to the lack of existing emotion datasets labelled with both fine-grained emotions as well as their associated scores to each input. [Zhou et al. \(2016\)](#) introduced the task of emotion distribution learning, which takes into account the theory of Plutchik’s wheel of emotion in order to learn emotion distributions. Next, [Zhang et al. \(2018\)](#) proposed a multi-task-loss approach (MTL) involving learning of both emotion distribution and classification, which is based on a CNN network ([Kim, 2014](#)) trained with cross-entropy and Kullback-Leibler loss functions jointly. The former loss function optimises the classification task, while the latter optimises the distribution learning task.

3.6.3 Unsupervised Learning

Limited research has been done on unsupervised emotion recognition. This can be attributed to the complexity and ambiguity of the task that makes unsupervised learning

hard or ineffective. One of the first approaches in modelling unsupervised emotion recognition was proposed by [Mac Kim et al. \(2010\)](#) who experimented with different unsupervised techniques using emotion lexicons and a number of dimensionality reduction algorithms (e.g., Latent Semantic Analysis, Probabilistic Latent Semantic Analysis and Non-negative Matrix Factorisation). [Agrawal and An \(2012\)](#) introduced an unsupervised approach based on the semantic relatedness between word-word statistical co-occurrences. The authors first selected a small set of emotion-bearing keywords (e.g., happy, glad, joy, good and love) and computed pointwise mutual information between those emotion words and each word in a sentence. After that, a vector was created that represented how much association there is between the sentence and each one of Ekman’s six basic emotions. Some syntactic dependency features (i.e., adjectival complement, adjectival modifier and negation modifier) were also used to account for cases like emotional shift and flipping. Finally, the approach was evaluated on three benchmark emotion datasets. The authors followed-up on the same task, but used additional association measures and experimented with three different window types ([Agrawal and An, 2016](#)).

Theater critic Michael Riedel (playing himself) also shows up, uninvited. Ivy is put out by this and gets **angry** at Michael about it. We hear but don't see Ivy singing "Bittersweet Symphony" at her **party**. Derek then walks in and gives her a present and wishes her **happy** birthday.

Figure 3.8: An instance of text containing emotion cue (i.e., party) and seed words (i.e., happy and angry). Taken from ([Agrawal and An, 2016](#)).

Figure 3.8 shows how the three settings can be formed for the given instance: i) The nearest cue word to the seed word is chosen. Based on the nearest selection criteria, the word “party” is co-occurring with “happy” instead of “anger”. ii) The preceding seed word is selected as co-occurring with the cue word. Here “angry” is chosen instead of “happy”. iii) The third setting considers the seed word that follows the cue word. The cue word “party” is followed by “happy” which is selected. It is worth mentioning that the word-word statistical co-occurrences were generated in this work from a large corpus of reviews that is more related to the emotion task.

Recently, [Gollapalli et al. \(2020\)](#) introduced an Emotion Sensitive TextRank (ES-TeR) approach based on the same idea of prior approaches in terms of taking word-word co-occurrences. Nevertheless, they exploit random walks on graphs in order to associate input texts with a set of emotions, whereas [Zad and Finlayson \(2020\)](#)

discussed a framework for unsupervised emotion recognition, focused on a specific domain, i.e., narrative text.

3.6.4 Transfer Learning

Transfer learning (TL) is an approach that deals with the scarcity of labelled data. In this respect, the main idea of this approach is to leverage information/knowledge from auxiliary domains (also called “domain-adaptation”) or tasks (also called “multi-task learning”) to boost the model performance on the target domain of interest. Another widely-used TL approach is two-stage fine-tuning, where the first-stage is focused on training Language Models (LMs) on a large volume of data, while fine-tuning it on the target domain of interest in the second-stage. In this way, LMs can learn complex linguistic phenomena (Clark et al., 2019), as well as producing high quality word embeddings that can be used directly to initialise existing neural architectures.

On one hand, there are approaches that adopt multi-task learning setting for textual emotion recognition. Yu et al. (2018) proposed a Dual Attention Transfer Network (DATN) to improve multi-label emotion classification with the help of sentiment classification. The DATN approach considered training both sentiment and emotion classifiers, while simultaneously encouraging the transfer of knowledge from sentiment to emotion. To address this issue, the model explicitly minimised the similarity between the two sets of attention weights (DA) by using the cosine similarity loss function. Xu et al. (2018) introduced “Emo2Vec”, which incorporated emotional semantics into embeddings. Emo2Vec was trained on six different emotion-related tasks and demonstrated strong performance over previously developed emotion-specific embeddings. Zhang et al. (2018) proposed a Multi-Task-Loss approach (MTL) that involved the learning of both emotion distribution and classification, where the distribution task is utilised to improve the results of single-label emotion classification. Akhtar et al. (2019) proposed a multi-task learning framework that jointly models related-emotion tasks, including emotion classification and emotion intensity prediction.

On the other hand, there are other approaches that use two-stage fine-tuning. Baziotis et al. (2018) was ranked the top-1 model of the SemEval-2018 competition (NTUA) inspired by the works of Howard and Ruder (2018) and Felbo et al. (2017) in terms of training their model on a large unlabelled data from the same domain and then fine-tuning it directly on the target task (i.e., SemEval-2018). Kant et al. (2018) followed this work and developed a multi-label emotion classification model based on the Transformer network trained on 40GB of Amazon reviews (McAuley et al., 2015). This

model achieved competitive results to the NTUA model. On a different dataset, Alhuzali et al. (2018c) also used the same two-stage fine-tuning, but focused on the implicit emotion shared task (IEST-2018). The proposed model was firstly trained on top of a pre-trained language model (LM), including forward (FW) and backward (BW) LMs. Both were then fine-tuned on the data provided by the task organisers and an ensemble of the two were considered. Among the 30 participating teams, this model ranked 3rd with 70.7% F-score. Furthermore, Agrawal et al. (2018) proposed an approach for producing emotion-enriched word embeddings that can be incorporated directly to downstream tasks for emotion recognition. The approach was based on training an LSTM model on large noisy data collected automatically via distant supervision, which was then used to initialise the word embeddings for evaluation against existing emotion datasets (i.e., Tales, blogs, ISEAR and EmoTweet-top-8³). Chronopoulou et al. (2018) explored different transfer learning strategies for the Implicit Emotion Shared Task (IEST). Figure 3.9 presents an overview of those strategies. In the first strategy, the authors pre-trained the *Word2Vec* model on a large volume of tweets, but combined the already trained model in the first strategy with another one trained on a sentiment dataset. In the third strategy, they followed the work of Howard and Ruder (2018) by training a LM on the same large volume of tweets and then fine-tuning it on the IEST dataset. Finally, the three strategies are fused together to produce the final predictions on this dataset.

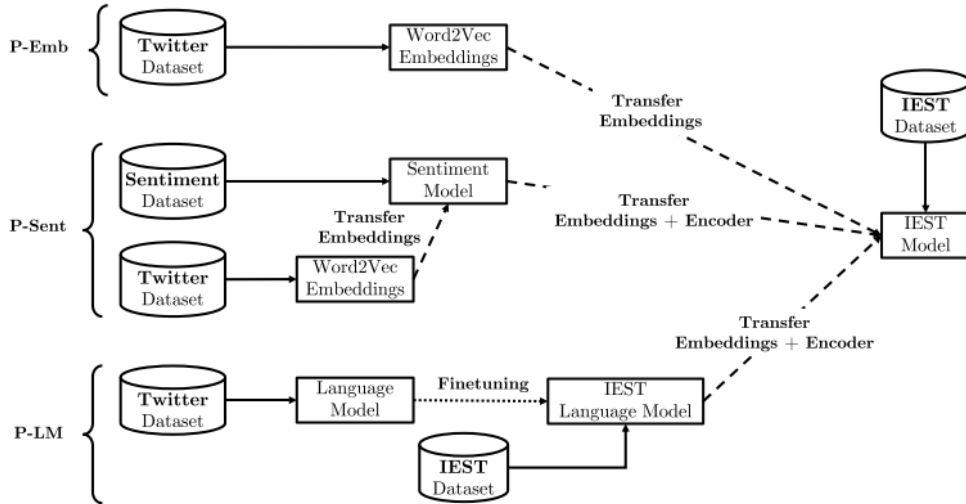


Figure 3.9: An overview of the different TL strategies employed in Chronopoulou et al. (2018).

³This corpus is a sub-set of EmoTweet that is discussed in Section 3.5.

3.7 Evaluation Metrics

TER models are evaluated with a set of metrics. Some metrics capture the overall performance of TER models, while others assess specific properties. We discuss the most widely-used metrics designed to estimate the performance of TER models. More detailed information about the evaluation metrics can be found in [Grandini et al. \(2020\)](#). Consider the case, where we have a binary classification problem with two classes. Let us first define some terminologies that can help in estimating the model performance. True Positive (TP) refers to the number of inputs correctly predicted by the model as being positive, and False Positive (FP) refers to the number of inputs incorrectly predicted by the model as being positive, but they are actually negative. True Negative (TN) refers to the number of inputs correctly predicted by the model as being negative and False Negative (FN) refers to the number of inputs incorrectly predicted by the model as being negative, but they are actually positive. Table 3.2 presents the confusion matrix for a binary classification problem, from which we can define commonly used metrics, including precision, recall, accuracy and F1-score.

		Prediction	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

Table 3.2: Confusion matrix of binary classification.

Precision and Recall (P& R). Precision is the ratio of correct positive predictions (TP) to the total predicted positives (TP + FP), whereas Recall is the ratio of correct positive predictions (TP) to the total positive examples (TP + FN).

$$P = \frac{TP}{TP + FP} \quad (3.1) \quad R = \frac{TP}{TP + FN} \quad (3.2)$$

Accuracy estimates the correct predictions made by the model, i.e., the ratio of correctly predicted inputs by the model to the total number of inputs.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.3)$$

F1-Score (F1). The above metrics can also be combined to estimate another metric, called F1-score defined as the harmonic mean of precision and recall. It is a special case of the general F_β score, in which both precision and recall are equally weighted

(i.e., $\beta = 1$).

$$F_\beta = (1 + \beta^2) \cdot \frac{P \cdot R}{\beta^2 \cdot P + R} \quad (3.4)$$

Jaccard Index (J). The Jaccard Index, also known as the Jaccard similarity coefficient, is a statistical measure used for understanding the similarities between two sets. This metric emphasises similarity between sample sets, and is formalised as the size of the intersection divided by the size of the union of the predicted (P) and correct (C) sets. The Jaccard metric is often used to evaluate multi-label classification, in which we have a true set (i.e., obtained from ground-truth) and a prediction set (i.e., produced by the model output).

$$J(C, P) = \frac{|C \cap P|}{|C \cup P|} \quad (3.5)$$

Micro- and Macro-averaged Metrics. The above-discussed measures are often compatible with binary classification experiments, where the number of classes are simply equivalent to two. However, it is common to have more than two classes and hence we can build upon Table 3.2 and extend it to multi-class classification. Table 3.3 demonstrates an example of three-class classification, where the first, second and third classes correspond to “joy”, “anger” and “neutral or no-emotion”, respectively. For instance, when the model prediction is “anger”, but the correct class is “joy”. Then, two types of errors are considered, where the first error is a FP for class (b) because of mis-classification, while the second error is a FN for class (a) because of failure to identify the correct class. The same applies to when the model prediction is “joy”, but the correct class is “anger”. In addition, when the model prediction is “neutral”, FN errors are only counted for each incorrect prediction. In the case of neutral being the correct class, FP errors are only considered for each wrong prediction.

		Prediction		
		Joy (a)	Anger (b)	Neutral (c)
Actual	Joy (a)	TP	FP (b) & FN (a)	FN (a)
	Anger (b)	FP (a) & FN (b)	TP	FN (b)
	Neutral (c)	FP (a)	FP (b)	TN

Table 3.3: Confusion matrix of multi-class classification.

To obtain the result of F1-score in a multi-class setting, we first compute macro-averaged metric for both precision and recall, which are calculated per class and then by taking the average over all classes. The macro-averaged metric estimates all the

classes independently and does not take the problem of imbalanced classes into account. One can compute both the macro-averaged and micro-averaged for precision and recall as in Equation (3.6/3.7) and (3.8/3.9), respectively. k denotes the number of classes in the multi-class setting. The same can be applied to estimate the F1-score by substituting P and R in Equation (3.1/3.2) with the following.

$$P_{\text{macro}} = \frac{\sum_{k=1}^K P_k}{K} \quad (3.6) \quad R_{\text{macro}} = \frac{\sum_{k=1}^K R_k}{K} \quad (3.7)$$

$$P_{\text{micro}} = \frac{\sum_k TP_k}{\sum_k TP_k + \sum_k FP_k} \quad (3.8) \quad R_{\text{micro}} = \frac{\sum_k TP_k}{\sum_k TP_k + \sum_k FN_k} \quad (3.9)$$

3.8 Summary and Limitations

In this chapter, an overview of emotion recognition was presented. Firstly, we reviewed the problem of TER and its related tasks that are relevant to the work presented in this thesis. Secondly, we discussed related terms to emotion, more specifically how emotion can be distinguished from those related terms. Thirdly, we described models of emotion that are used to inform the classification and categorisation of emotion. Finally, we discussed existing emotion corpora as well as common proposed approaches to textual emotion recognition. This review has been influenced by research drawn from affective computing, natural language processing and psychology, revealing the diverse nature of research in this area.

An extensive discussion of previous work in the area of TER is provided in this chapter. More specifically, emotional expression in text was considered from both the writer's and the reader's point of view. It is often the case that TER models tend to predict the writer's emotions, but it can often be hard due to the lack of verbal emotional cues. Existing emotion corpora that undertake annotation studies often ask human annotators to assign emotion classes to a piece of text based on the writer's feeling (Alhuzali et al., 2018a; Mohammad, 2012b). Aman and Szpakowicz (2007) also requested the annotators to choose a "non-emotion" class when the text did not show any sign of the writer's feeling in order to restrict any further interpretation from the annotators. Then, we reported on models of emotion because they inform TER research on how best to classify or categorise emotion expression in text. Next, we described common approaches to TER and focused our discussion on supervised learning because it is the main learning setting on which this thesis is based. Nevertheless, we explain other learning approaches beyond supervised learning for completeness, as well as highlighting previous efforts undertaken in this area.

Based on our discussion, we observe the following: i) the categorical model is widely-adopted since it is straightforward, easy to use/understand and less complicated as opposed to the dimensional model. ii) Existing emotion datasets are mostly focused on single-labels and are created based on either gold annotations or distant supervision methods. Each approach has its own benefits and drawbacks in terms of data quality and size. iii) The majority of previous approaches on TER are also based on supervised and distant-supervised learning. This follows the trend of emotion datasets proposed in the literature as shown in Table 3.1. iv) The main methodology of prior research focused on the classification and identification of emotion expression in text, but both correlations and associations are often overlooked. Although there are some works that tackle emotion correlations, they often rely on theories of emotion or lexicons to learn emotion-specific features. These approaches are not generalisable and difficult to apply to different domains and languages.

Nevertheless, the main aim of this thesis is to build generalisable computational methods for TER, while addressing the problem of learning emotion correlations and associations without utilising any external resources or theories of emotion. The rationale for this is that we can easily adapt our approach to other domains as well as languages, which has often been an obstacle in previous research. The concept of correlation is also defined for single-label emotion corpora, many of those corpora fall under this problem. Introducing the concept of correlations to the single-label case can help TER models to be robust against highly confused emotions.

In Chapter 4, we present our method for multi-label emotion classification and how we incorporate both correlations and associations into the proposed model. In Chapter 5, we investigate the effect of emotion/sentiment knowledge within two different tasks, and demonstrate that our proposed method in Chapter 4 can be easily adapted to other relevant tasks, in which we explain in detail how that is achieved. Chapter 6 then reports on our study of highly confused emotions for the single-label case and this aims to help TER model to achieve better performance and improve its discriminator ability. Finally, we conclude with our analyses and observations based on our experiments and provide some limitations as well as future directions.

Chapter 4

SpanEmo: Casting Multi-label Emotion Classification as Span-Prediction

Current approaches to Textual Emotion Recognition (TER), mainly classify emotions independently without considering that emotions can co-exist. Such approaches overlook potential ambiguities, in which multiple emotions overlap. In this chapter, we propose a novel neural model for multi-label emotion classification to address our first research question (**RQ#1**), as described in Chapter 1, which is concerned with the incorporation of emotion correlations and emotion-specific associations without the use of any external resources (e.g., lexicons and theories of emotion). We introduce SpanEmo casting multi-label emotion classification as span-prediction, which can aid TER models to learn associations between labels and words in an input instance. Furthermore, we introduce a training objective focused on modelling multiple co-existing emotions in the input instance. Experiments performed on the SemEval-2018 multi-label emotion data over three language sets (i.e., English, Arabic and Spanish) demonstrate our method’s effectiveness. Finally, we present different analyses that illustrate the benefits of our method in terms of improving the model performance and learning meaningful associations between emotion classes and words in the input instance. It is worth noting that this chapter is drawn from [Alhuzali and Ananiadou \(2021b\)](#).

4.1 Motivation

Emotion is essential to human communication, thus TER models have a host of applications from health and well-being (Alhuzali and Ananiadou, 2019; Aragón et al., 2019; Chen et al., 2018) to consumer analysis (Alaluf and Illouz, 2019; Herzig et al., 2016) and user profiling (Volkova and Bachrach, 2016; Mohammad and Kiritchenko, 2013), among others. Interest in this area has given rise to new Natural Language Processing (NLP) approaches aimed at emotion classification, including single-label and multi-label emotion classification. Most existing approaches for multi-label emotion classification (Ying et al., 2019; Baziotis et al., 2018; Yu et al., 2018; Badaro et al., 2018; Mulki et al., 2018; Mohammad et al., 2018; Yang et al., 2018) do not effectively capture emotion-specific associations, which can be useful for prediction, as well as learning of associations between emotion labels and words in an input instance. In addition, standard approaches in emotion classification treat individual emotion independently. However, emotions are not independent; a specific emotive expression can be associated with multiple emotions. The existence of association/correlation among emotions has been well-studied in psychological theories of emotions, such as Plutchik’s wheels of emotion (Plutchik, 1984) that introduces the notion of mixed and contrastive emotions. For example, “joy” is close to “love” and “optimism”, instead of “anger” and “sadness”.

#	Sentence	GT
S1	well my day started off great the mocha machine wasn't working @ mcdonalds.	anger, disgust, joy, sadness
S2	I'm doing all this to make sure you smiling down on me bro.	joy, love, optimism

Table 4.1: Example Tweets from SemEval-18 Task 1. GT represents the ground truth labels.

Consider *S1* in Table 4.1, which contains a mix of positive and negative emotions, although it is more negative oriented. This can be observed clearly via the ground truth labels assigned to this example, where the first part of this example only expresses a positive emotion (i.e., joy), while the other part expresses negative emotions. For example, clue words like “great” are more likely to be associated with “joy”, whereas “wasn’t working” are more likely to be associated with negative emotions. Learning such associations between emotion labels and words in the input instance can help ER models to predict the correct labels. *S2* further highlights that certain emotions

are more likely to be associated with each other. Based on these observations, we seek to answer the following research questions, as discussed in Chapter 1: i) how to enable TER models to learn emotion-specific associations by taking into account label information and ii) how to benefit from the multiple co-existing emotions in a multi-label emotion data set, with the intention of learning label correlations. Our contributions, which are described in Chapter 1, are also mentioned below:

- a novel framework casting the task of multi-label emotion classification as a span-prediction problem. We introduce SpanEmo to train the model to take into consideration both the input instance and a label set (i.e., emotion classes) for selecting a span of emotion classes in the label set as the output. The objective of SpanEmo is to predict emotion classes directly from the label set and capture associations corresponding to each emotion. This also explains the naming behind our approach “SpanEmo”.
- a training objective, modelling multiple co-existing emotions for each input instance. We make use of the Label-Correlation Aware loss (LCA) (Yeh et al., 2017), originally introduced by Zhang and Zhou (2006). The objective of this loss function is to maximise the distance between positive and negative labels, which is learned directly from the multi-label emotion data set. The overall training objective contains both the LCA loss and Binary Cross-Entropy loss (BCE). Both loss functions are trained jointly in an end-to-end fashion.
- a large number of experiments and analyses both at the word- and tweet-level, demonstrating the strength of SpanEmo for multi-label emotion classification across multiple languages.

Our work is motivated by research focused on learning features corresponding to each emotion as well as incorporating the relations between emotions into a loss function (Fei et al., 2020; He and Xia, 2018). Our work differs as follows: i) our method learns features related to each corresponding emotion without relying on any external resources (e.g. lexicons). ii) We further integrated the relations between emotions into the loss function by taking advantage of the label co-occurrences in a multi-label emotion data set. In this respect, our approach does not rely on any theory of emotion. iii) We empirically evaluated our method for three languages, demonstrating its effectiveness as being language agnostic. In contrast to previous research, we focus on both learning emotion-specific associations and integrating the relations between emotions into the loss function.

The rest of the chapter is organised as follows: Section 4.2 describes our methodology, while Section 4.3 discusses experimental details. We evaluate the proposed method and compare it to related methods in Section 4.4. Section 4.5 reports on the analysis of results, while Section 4.6 provides some conclusions.

4.2 Methodology

4.2.1 Framework

Figure 4.1 presents our framework (SpanEmo). Given an input sentence and a set of classes, a base encoder was employed to learn contextualised word representations. Next, a feed forward network (FFN) was used to project the learned representations into a single score for each token. We then used the scores for the label tokens as predictions for the corresponding emotion label. The green boxes at the top of the FFN illustrate the positive label set, while the red ones illustrate the negative label set for multi-label emotion classification. We now turn to describing our framework in detail.

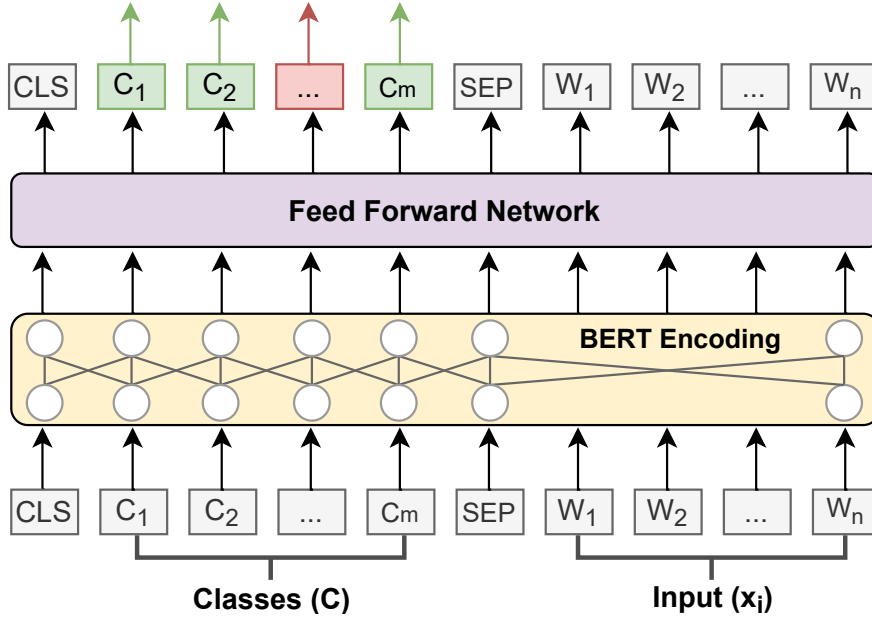


Figure 4.1: Illustration of our proposed framework (SpanEmo).

4.2.2 Our Method (SpanEmo)

Let $\{(x_i, y_i)\}_{i=1}^N$ be a set of N examples with the corresponding emotion labels of M classes (C), where x_i denotes the input sentence and $y_i \in \{0, 1\}^M$ represents the label set for x_i . As shown in Figure 4.1, both the label set and the input sentence were passed into the encoder BERT (Devlin et al., 2019). The encoder received two segments: the first corresponds to the set of emotion classes, while the second refers to the input sentence. The hidden representations ($H_i \in R^{T \times D}$)¹ for each input sentence and the label set were obtained as follows:

$$H_i = \text{Encoder}([\text{CLS}] + |C| + [\text{SEP}] + \mathbf{x}_i), \quad (4.1)$$

where $\{[\text{CLS}], [\text{SEP}]\}$ are special tokens and $|C|$ denotes the size of emotion classes. Figure 4.2 shows the SpanEmo input representation, which consists of the token, position and segment embeddings. The token embeddings represent each token with its vector representation of 768 dimensions, whereas the position embeddings encode positional information of tokens. The segment embeddings further inform the encoder to distinguish between the label segment and input segment. Feeding both segments to the encoder has a few advantages. Firstly, the encoder can interpolate between emotion classes and all words in the input sentence. Secondly, a hidden representation is generated both for words and emotion classes, which can be further used to understand whether the encoder can learn association between the emotion classes and words in the input sentence. Thirdly, SpanEmo is flexible because its predictions are directly produced from the first segment corresponding to the emotion classes.

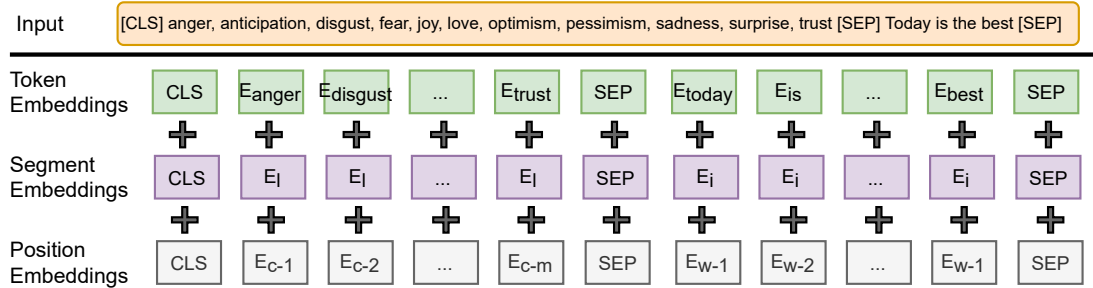


Figure 4.2: SpanEmo input construction.

We further introduced a FFN consisting of a non-linear hidden layer with a tanh activation ($f_i(H_i)$) as well as a position vector $p_i \in R^D$, which was used to compute

¹ T and D denote the input length and dimensional size, respectively.

a dot product between the output of f_i and p_i . As our task involved a multi-label emotion classification, we added a sigmoid activation to determine whether class_i was the correct emotion label or not. It should be mentioned that the use of the position vector is quite similar to how start and end vectors are defined in transformer-based models for question-answering. Finally, the span-prediction tokens were obtained from the label segment and then compared with the ground truth labels since there was a 1-to-1 correspondence between the label tokens and the original emotion labels.

$$\hat{y} = \text{sigmoid}(\text{FFN}(\mathbf{H}_i)), \quad (4.2)$$

4.2.3 Label-Correlation Aware (LCA) Loss

Following [Yeh et al. \(2017\)](#), we employed the label-correlation aware loss, which takes a vector of true-binary labels (y), as well as a vector of probabilities (\hat{y}), as input:

$$\mathcal{L}_{\text{LCA}}(y, \hat{y}) = \frac{1}{|y^0| |y^1|} \sum_{(p,q) \in y^0 \times y^1} \exp(\hat{y}_p - \hat{y}_q), \quad (4.3)$$

where y^0 denotes the set of negative labels (i.e., incorrect predictions), while y^1 denotes the set of positive labels (i.e., correct predictions). \hat{y}_p and \hat{y}_q represent the p^{th} and q^{th} elements of the \hat{y} vector. The objective of this loss function is to maximise the distance between positive and negative labels by implicitly retaining the label-dependency information. In other words, the model should be penalised when it predicts a pair of labels that should not co-exist for a given example.

4.2.4 Training Objective

To model label-correlation, we combined LCA loss with BCE and trained them jointly. This aimed to help the LCA loss to focus on maximising the distance between positive and negative label sets, while at the same time taking advantage of the BCE loss by maximising the probability of the correct labels. We experimentally observed that training our approach jointly with those two loss functions produced the best results. The overall training objective was computed as follows:

$$\mathcal{L} = (1 - \alpha) \mathcal{L}_{\text{BCE}} + \alpha \sum_{i=1}^M \mathcal{L}_{\text{LCA}}, \quad (4.4)$$

where $\alpha \in [0, 1]$ denotes the weight used to control the contribution of each part to the overall loss.

4.3 Experiments

4.3.1 Implementation Details

We used PyTorch (Paszke et al., 2017) for implementation and ran all experiments on an Nvidia GeForce GTX 1080 with 11 GB memory. We used a pre-trained BERT_{base} model and then fine-tuned it on the data of SemEval-2018, which is described below, by utilising the open-source Hugging-Face implementation (Wolf et al., 2019). For experiments related to Arabic, we chose “bert-base-arabic” developed by Safaya et al. (2020), while selecting “bert-base-spanish-uncased” developed by Cañete et al. (2020) for Spanish. All three models were trained on the same hyper-parameters with a fixed initialisation seed, including a feature dimension of 768, a batch size of 32, a dropout rate of 0.1, an early stop patience of 10 and 20 epochs. Adaptive Moment Estimation Algorithm (ADAM) was selected for optimisation (Kingma and Ba, 2014) with a learning rate of $2e-5$. It should be mentioned that we tuned our method only on the validation set and further report on the analysis of the effect of parameter α in Section 4.5.5. Table 4.2 summarises the hyper-parameters used in our experiments.

Parameter	Value
Feature dimension	768
Batch size	32
Dropout	0.1
Early stop patience	10
Number of epochs	20
Learning rate	$2e-5$
Optimiser	Adam
Alpha (α)	0.2

Table 4.2: Hyper-parameter values.

4.3.2 Data Set and Task Settings

In this work, we chose SemEval-2018 (Mohammad et al., 2018) for our multi-label emotion classification, which is based on labelled data from tweets in English, Arabic and Spanish. The data was partitioned into three sets: training set (Train), validation set (Valid) and test (Test) set. To have comparable results with prior works, we followed the metrics in Mohammad et al. (2018), and evaluated our experiments using micro

F1-score, macro F1-score and Jaccard index score². Table 4.3 presents the summary of all three sets for each language, including the number of instances in the train, valid and test sets. In addition, the number of emotion classes, its distributions, and the percentage of instances with varying numbers of classes (co-existing) are included. It is worth noting that these percentages do not include the neutral instances.

Info./Lang.	English	Arabic	Spanish
Train (#)	6,838	2,278	3,561
Valid (#)	886	585	679
Test (#)	3,259	1,518	2,854
Total (#)	10,983	4,381	7,094
Classes (#)	11	11	11
1 co.emo (%)	14.36	21.38	39.11
2 co.emo (%)	40.55	39.03	42.15
3 co.emo (%)	30.92	29.85	12.76
anger (%)	36.1	39.4	32.2
anticipation (%)	13.9	9.6	11.7
disgust (%)	36.6	19.6	14.7
fear (%)	16.8	17.8	10.5
joy (%)	39.3	26.9	30.5
love (%)	12.3	25.2	7.9
optimism (%)	31.3	24.5	10.2
pessimism (%)	11.6	22.8	16.7
sadness (%)	29.4	37.4	23.0
surprise (%)	5.2	2.2	4.6
trust (%)	5.0	5.3	4.6

Table 4.3: Data Statistics. co.emo: refers to the percentage of co-existing emotions.

To pre-process the data³, we used the “ekphrasis” tool designed for the specific characteristics of Twitter, i.e., misspellings and abbreviations (Baziotis et al., 2017). The tool offers different functionalities, such as tokenisation, normalisation, spelling correction, and segmentation. We utilised the tool to tokenise the text, convert words to lowercase, normalise user mentions, urls and repeated-characters.

Finally, we compared the performance of SpanEmo to some baseline as well as state-of-the-art models on all three languages. For experiments related to English, we selected eight models, while we chose three and two models for both Arabic and Spanish, respectively. We also include the results of BERT_{base} as an additional strong

²Jaccard score is defined as the size of the intersection divided by the size of the union of the true label set and predicted label set. This metric was also described in Chapter 3.

³It is worth mentioning that emojis are kept as they are, i.e., no modification or conversion was applied.

baseline. More detailed information about the baseline and state-of-the-art models can be found in Chapter 3, more specifically in Section 3.6.2.

4.4 Results

Table 4.4 presents the performance of our proposed approach (SpanEmo) on all three languages, in terms of micro F1-score (miF1), macro F1-score (maF1) and Jaccard index score (jacS), and compares it to the baseline and state-of-the-art models.

Language	English				
Model/Metric	LE	LC	miF1	maF1	jacS
JBNN (He and Xia, 2018)	✗	✓	0.632	0.528	-
RERc (Zhou et al., 2018)	✗	✓	0.651	0.539	-
DATN (Yu et al., 2018)	✗	✗	-	0.551	0.583
NTUA (Baziotis et al., 2018)	✗	✗	0.701	0.528	0.588
BERT _{base} (Devlin et al., 2019)	✗	✗	0.695	0.520	0.570
BERT _{base} +DK (Ying et al., 2019)	✗	✗	0.713	0.549	0.591
BERT _{base} -GCN (Xu et al., 2020a)	✓	✓	0.707	0.563	0.589
LEM (Fei et al., 2020)	✗	✓	0.675	0.567	-
Seq2Emo (Huang et al., 2021)	✓	✗	0.700	0.519	0.586
SpanEmo (ours)	✓	✓	0.713	0.578	0.601
Arabic					
	LE	LC	miF1	maF1	jacS
Tw-StAR (Mulki et al., 2018)	✗	✗	0.597	0.446	0.465
EMA (Badaro et al., 2018)	✗	✗	0.618	0.461	0.489
BERT _{base} (Devlin et al., 2019)	✗	✗	0.650	0.477	0.523
HEF (Alswaidan and Menai, 2020a)	✗	✗	0.631	0.502	0.512
SpanEmo (ours)	✓	✓	0.666	0.521	0.548
Spanish					
	LE	LC	miF1	maF1	jacS
Tw-StAR (Mulki et al., 2018)	✗	✗	0.520	0.392	0.438
ELiRF (González et al., 2018)	✗	✗	0.535	0.440	0.458
BERT _{base} (Devlin et al., 2019)	✗	✗	0.596	0.474	0.487
SpanEmo (ours)	✓	✓	0.641	0.532	0.532

Table 4.4: The results of multi-label emotion classification on SemEval-2018 test set. *LE* refers to the use of label embedding and *LC* refers to the use of label correlation.

As shown in Table 4.4, there are some approaches that incorporate label embeddings and label correlations, whereas other approaches only focus on the task of multi-label emotion classification. Table 4.4 demonstrates that our method outperformed all models on all languages, as well as on almost all metrics, with a marginal improvement of up to 1-1.3% for English, 1.9-3.6% for Arabic and 6.3-9.2% for Spanish. This demonstrates the utility and advantages of SpanEmo, as well as the label-correlation aware loss for improving the performance of multi-label emotion classification in English, Arabic and Spanish.

Based on the empirical results reported in Table 4.4, the following observations can be made. First, incorporating the relations between emotions into the models tends to lead to higher performance, especially for macro F1-score. For example, both DATN and LEM learn emotion-related features and achieve better performance than NTUA and BERT_{base}+DK. Additionally, ELiRF makes use of various sentiment/emotion features (i.e., learned from lexica) and it yielded the best performance among the three compared models. This corroborates our earlier hypothesis that learning emotion-specific associations is crucial for improving the performance. Although BERT_{base}+DK adopts the same encoder as our own and adds domain knowledge, our method still performs strongly, especially for both macro F1- and Jaccard scores with a marginal improvement of up to 2.9% and 1%, respectively. In short, capturing emotion-specific associations as well as integrating the relations between emotions into the loss function, helped SpanEmo to achieve the best results compared with all models on almost all metrics.

4.4.1 Ablation Study

To understand the effect of our framework, we undertook an ablation study of the model performance under three settings: firstly, the model was trained only with BCE loss; secondly, it was trained only with LCA loss; and thirdly it was trained without the label segment. The third setting is equivalent to training the model as a simple multi-label classification task, by only considering the input sentence. Table 4.5 presents the results. When SpanEmo was trained without the LCA loss, the results dropped by 1-2% for macro F1- and Jaccard score. In addition, the results of SpanEmo dropped by 1-2% for two metrics apart from the macro F1-score when trained without the BCE loss. However, the removal of the label segment led to a much higher drop of 3-6%. The same patterns were also observed in the Arabic and Spanish experiments.

Language	English			Arabic			Spanish		
Model/Metric	miF1	maF1	jacS	miF1	maF1	jacS	miF1	maF1	jacS
SpanEmo (joint)	0.713	0.578	0.601	0.666	0.521	0.548	0.641	0.532	0.532
- \mathcal{L} (LCA)	0.712	0.564	0.590	0.654	0.481	0.534	0.629	0.526	0.507
- \mathcal{L} (BCE)	0.698	0.583	0.582	0.660	0.526	0.532	0.606	0.544	0.499
- Label Seg.	0.695	0.520	0.570	0.650	0.477	0.523	0.596	0.474	0.487

Table 4.5: Ablation experiment results. The second and third rows correspond to the removal of the respective loss function, whereas the last row corresponds to the removal of the label segment.

This supports our earlier hypothesis that casting the task of multi-label emotion classification as span-prediction is beneficial for improving both the representation and performance of multi-label emotion classification.

In addition, we ran a significance test to further strengthen our claims, for which the results of SpanEmo (joint) were compared with the results of the last row (i.e., the removal of both the label segment and the joint training objective). Our hypothesis was that the two models have a different proportion of errors on the test set of the SemEval-2018 dataset. To perform the test, we chose the “McNemar test” following the work of [Dror et al. \(2018\)](#). More specifically, we created a 2×2 contingency table, which highlights the outcomes of the two selected models (i.e., their correct and incorrect predictions). The test demonstrated that there is statistically significant difference at $p < 0.05$ in the disagreements between the two models.

4.5 Analysis

4.5.1 Prediction of Multiple Emotions

We additionally validated the effectiveness of our method for learning the multiple co-existing emotions on English, Arabic and Spanish sets. Table 4.6 presents the results, including $BERT_{base}$. SpanEmo demonstrated a strong ability to handle multi-label emotion classification much better than $BERT_{base}$. Since $BERT_{base}$ is trained only with BCE loss, here we include the results of our method trained only with this loss function. SpanEmo still achieved consistent improvement as the number of co-existing emotions increases, showing the usefulness of our method in learning multiple emotions. Improvement on all metrics can clearly be observed for English and Arabic

experiments, but not as much for Spanish. This may be attributed to the high percentage of single-label data, which is around (40%) for Spanish, while it is lower than that for both English and Arabic. Obviously, SpanEmo can be used without LCA loss, and still obtain decent performance. Nevertheless, training our method jointly with the LCA loss leads to better results.

Model/Metric		miF1	maF1	jacS	miF1	maF1	jacS	miF1	maF1	jacS
English		≥ 1 co.emo			≥ 2 co.emo			≥ 3 co.emo		
BERT _{base}	BCE	0.703	0.515	0.587	0.712	0.521	0.596	0.692	0.509	0.554
SpanEmo	BCE	0.716	0.563	0.599	0.737	0.578	0.629	0.748	0.597	0.639
SpanEmo	Joint	0.724	0.590	0.613	0.746	0.606	0.648	0.753	0.624	0.643
Arabic		≥ 1 co.emo			≥ 2 co.emo			≥ 3 co.emo		
BERT _{base}	BCE	0.656	0.459	0.527	0.668	0.471	0.531	0.682	0.485	0.555
SpanEmo	BCE	0.689	0.518	0.565	0.709	0.536	0.586	0.745	0.567	0.629
SpanEmo	Joint	0.689	0.534	0.565	0.710	0.551	0.587	0.746	0.584	0.626
Spanish		≥ 1 co.emo			≥ 2 co.emo			≥ 3 co.emo		
BERT _{base}	BCE	0.603	0.476	0.526	0.567	0.461	0.441	0.518	0.432	0.364
SpanEmo	BCE	0.653	0.528	0.561	0.646	0.528	0.519	0.663	0.566	0.508
SpanEmo	Joint	0.662	0.565	0.581	0.655	0.568	0.530	0.644	0.570	0.490

Table 4.6: Presenting the number of co-existing emotion classes. The second row in each group corresponds to the removal of LCA loss from SpanEmo. The best results in each language group are marked in bold.

4.5.2 Learning Emotion-specific Associations

4.5.2.1 Word-Level

In this section, we present the top 10 words learned by SpanEmo for each emotion class by extracting the learned representations for each emotion class and all words in every input instance, and then computing the similarity between them via cosine similarity, Equation (4.5).

$$Assoc(h_{e^i}, h_{w^i}) = \frac{\vec{h}_{e^i} \cdot \vec{h}_{w^i}}{\|\vec{h}_{e^i}\| \|\vec{h}_{w^i}\|}, \quad (4.5)$$

where $h_{e^i} \in H_i$ is the hidden representation for an emotion class and $h_{w^i} \in H_i$ is the hidden representation for the i^{th} word from the input instance x_i (see detailed description of H_i/x_i in Section 4.2.2). Finally, we performed this operation on all words in the SemEval-2018 English validation set and then sorted them for each emotion class in ascending order. Table 4.7 presents the top-10 words per emotion class. As

Emotion	Top 10 Words
anger	anger pissed wrath idiots dammit kicking irritated thrown smashed complain
anti.	prediction planning mailsport assumptions upcoming waiting route waited frown ideas
disg.	disgusting smashed gross hate pissed wrath dirty awful vile dumb
fear	nervous fear terror frightening afraid frown panic terrifying scary dreading
joy	happy excitement joyful congratulations glad delightful excited adorable amusing smiling
love	love sweetness loved hug mate lucky carefree shine care gracious
optm.	optimism integrity salvation persevere perspective bright effort faith glad lord
pesm.	hopeless frown disappointed weary dread despair depressing chronic suicide pain
sad.	sadness frown depressing saddened hurt disappointed weary upset sorrow hate
sur.	stunned awestruck shocking awe mailsport buster genuinely curious hardly believing
trust	integrity shine respect courage sign effort confident faith easy kindness

Table 4.7: Top 10 words associated with each corresponding emotion. *anti.* stands for “anticipation”, *optm.* stands for “optimism”, *pesm.* stands for “pessimism”, *disg.* stands for “disgust”, *sad.* stands for “sadness” and *sur.* stands for “surprise”.

shown in Table 4.7, the words discovered by our framework are indicative of the corresponding emotion. This demonstrates that SpanEmo learns meaningful associations between emotion classes and words automatically, which can be beneficial for feature extraction and learning. Additionally, SpanEmo demonstrated that it can learn diverse words as well as shareable words across some emotions. For example, the words $\{pissed, wrath, smashed\}$ are associated with both anger and disgust, demonstrating the ability of SpanEmo to learn the relations between emotions.

4.5.2.2 Tweet-Level

In Figure 4.3, we visualised an example from the English validation set annotated with four emotions, i.e., *anger*, *disgust*, *pessimism* and *sadness*. Our goal was to determine whether by adding emotion classes to the example, SpanEmo could learn their associations to each other. To compute the similarity between emotion classes and words in the example, we also followed the same process discussed in Section 4.5.2.1. As shown in Figure 4.3, the learned representations capture the association between the correct emotion label set and every token in the example. Interestingly, we can also observe that the word “happy” is usually expressed as a positive emotion, but, in this context, this word becomes negative and the model learns this contextual information. Moreover, the phrase “about to join the police academy” is associated with “anticipation”, which makes sense although this class is not part of the correct label set. Although our model associates some functional words with emotion classes, this behaviour is expected since it takes the contextual information into account as is the case with the word “happy”. However, other types (e.g., commas, apostrophes and periods) have no

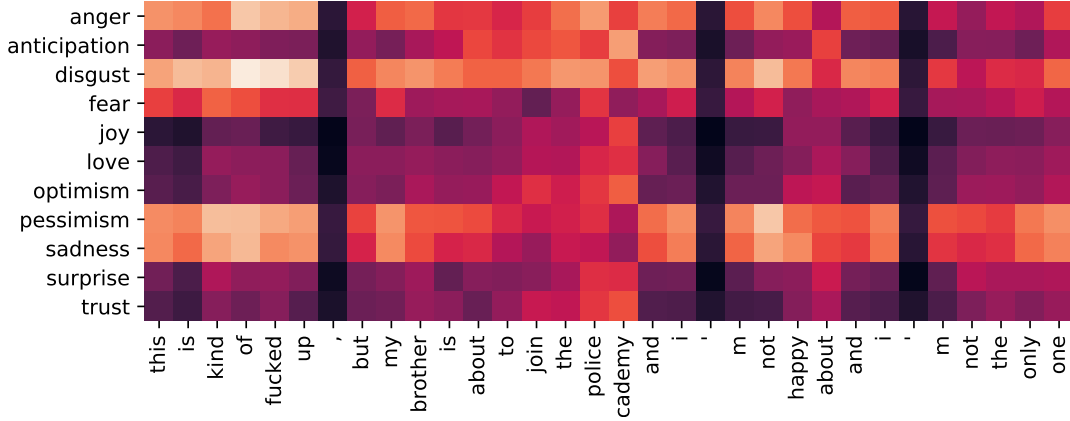


Figure 4.3: Visualisation of an example. The left presents the emotion labels, and the bottom presents the example. Each cell shows the cosine similarity value computed by using the hidden representation of each word and label. Lighter colour indicates higher similarity, while darker colour indicate lower similarity. Ground-Truth labels: anger, disgust, pessimism and sadness.

associations at all. This demonstrates the utility and advantages of our approach not only in deriving associations reported in the annotations, but also providing us with a mechanism to explore additional information beyond them.

In Section 4.5.2, both word- and Tweet-level analyses suggest that our work might be beneficial for explainable artificial intelligence. This is because we can easily investigate model behaviours w.r.t. the whole data or specific instances as is the case in both Table 4.7 and Figure 4.3. We thus anticipate that future directions can take advantage of our work to, for example, understand and interpret model predictions.

4.5.3 Qualitative Analysis

We analyse the model predictions, for which we randomly selected two examples per language and extract their ground truth and predictions. Table 4.8 shows the examples with their ground truth labels, as well as the model predictions for our approach SpanEmo and BERT-base (BERT). The first three rows represent examples from English tweets, the next three rows show examples from Arabic tweets and the last two rows present examples from Spanish tweets. Our first observation is that our approach captures more emotion labels than BERT-base. For instance, BERT-base only predicts one label for four examples out of eight, while their correct sets consist of at least 2-4 emotions. BERT-base also appears to be confused with certain part of the examples, which cause it to select incorrect emotions. This behaviour can be clearly seen

#	Tweet	GT	BERT	SpanEmo
1	well my day started off great the mocha machine wasn't working @ mcdonalds.	A,D,J,S	J	A,D,J,S
2	What do I do with my heart that is trembling by just the thought of it? I really don't know what love is.	L,S	J,O	L,S
3	Can't handle rude people. Doesn't matter what job you do, a consultant or not, treat people how you would like to be treated #disappointed.	A,D,S	A,D	A,D,S
4	عندما تظن بأن الله سيبدلك بعد الشقاء سعادة وبعد الدموع ابتسامة فقد أدت عبادة عظيمة ألا وهي حسن الظن بالله. Trans: (When you thought that God would give you after misery happiness, and after tears a smile, then you performed a great worship, which is good thinking of God.)	J,L,O,T	J,L,O	J,L,O,T
5	ياخي ما تفرض على الناس يتابعوك وثاني شي هالأيام الناس تدور الفكاهة الناس تدور الشي اللي تنبسط وتضحك ما تبي تتعلم. Trans: (Brother, don't force people to follow you, people these days look for something that make them laugh and happy, but they don't want to learn.)	A,D	J	A,D
6	وصلت للعمر هذا وللحين ما عرف اكل حبوب اقدر اقول فوييا الحبوب متلازمه فيني من الصغر ومازال. Trans: (I have reached this age, and until now I do not know how to take pills. I can say the phobia of pills has been associated with me since I was young and still is.)	F,S	F,A,S	F,S
7	harry está emocionado por la gira mundial, y yo estoy deprimida de vuelta porque no tengo entrada. Trans: (harry is excited about the world tour, and I'm depressed again because I don't have a ticket.)	P,S	J	P,S
8	Buenas noches a toda la pipul. Que tengan una excelente semana. Trans: (Good night to all the pipul. Have an excellent week.)	J,O	J	J,O

Table 4.8: Prediction of emotion classes with Bert-base (BERT) versus with our method (SpanEmo). GT: refers to the ground truth labels. Translation for Arabic and Spanish examples are included in parentheses under each example. *A* refers to “anger”, *D* refers to “disgust”, *F* refers to “fear”, *S* refers to sadness, *J* refers to “joy”, *L* refers to “love”, *O* refers to “optimism”, *P* refers to “pessimism” and *T* refers to “trust”.

in the first, fourth and fifth examples in Table 4.8. However, our approach predicts all correct labels, which can be attributed to two reasons: i) The inclusion of both labels and each example help guide the model to learn associations between the labels and each example. This in return improves the model ability to be less confused about such strong expressions in text. ii) The use of label-correlation aware loss helps to force the model to take advantage of the label co-occurrences in data, for which often co-existing emotions are more likely to co-occur together.

We further selected three examples from the English validation set of SemEval-2018 to evaluate incorrect predictions made by our model. Examples 1-3 below illustrate the three examples with their ground truth (GT) labels as well as with the

incorrect predictions. We observed that SpanEmo assigned more labels, which can be attributed to the use of correlation loss. However, it still captures the correct set as shown in examples 1-3. Although the SpanEmo predictions are not identical to the GT labels, they are still acceptable. We expect such difference arises from annotation artifacts, which can cause the selection of less GT labels. Since the SemEval-2018 data was collected by using hashtags, the presence of such cues during the annotation phase can bias the annotators to select one emotion or another without considering the whole context.

1. You could have over a hundred million followers and still not a genuine person who understands you or wants to #cantshakethis #sadness. (GT: *pessimism and sadness*; SpanEmo: *disgust, pessimism and sadness*)
2. You can have a certain arrogance, and I think that's fine, but what you should never lose is the #respect for the others. (GT: *disgust and optimism*; SpanEmo: *anger, disgust and optimism*)
3. I have actually watch drugs destroy an entire family Mother's on kid row. Oldest daughter lost her child. Father is estranged. #horrific. (GT: *anger, disgust, fear, pessimism and sadness*; SpanEmo: *anger, disgust, fear and sadness*)

4.5.4 Label Correlations

Since one of the research questions in this thesis was to learn the multiple co-existing emotions from a multi-label emotion data set, we analysed the learned emotion correlations from SpanEmo and compared them to those adopted from the ground truth labels in the SemEval-2018 validation set. Figure 4.4 presents the two emotion correlations as obtained from the ground truth labels and from the predicted labels, respectively. It can be observed that Figure 4.4(b) is almost identical to 4.4(a), demonstrating that our method in capturing the emotion correlations is in line with what the emotion annotations have revealed. 4.4(b), which was learned by SpanEmo, also highlights that negative emotions are positively correlated with each other, and negatively correlated with positive emotions. For example, “anger and disgust” share almost the same patterns, which is consistent with the studies of [Mohammad and Bravo-Marquez \(2017a\)](#) and [Agrawal et al. \(2018\)](#), both of which report the same issue with negative emotions of “anger” and “disgust”, as they are easily confused with each other. This is not surprising as their manifestation in language is quite similar in terms of the use of similar

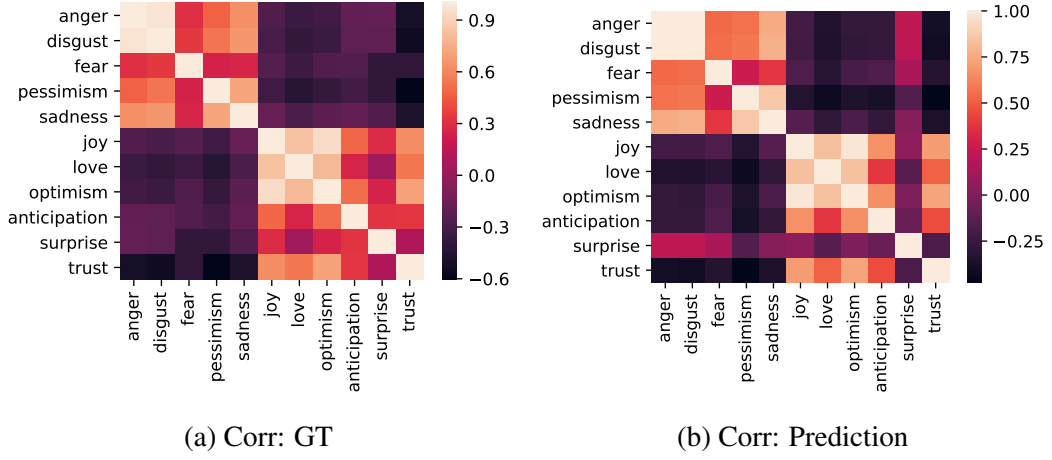


Figure 4.4: The left plot presents emotion correlations obtained from the ground truth (GT) labels, whereas the right plot presents emotion correlations obtained from the predicted labels.

words/expressions. We also noted this finding when analysing the top 10 key words learned by SpanEmo in Section 4.5.2.1. In short, taking into account emotion correlations is crucial for multi-label emotion classification in addressing the ambiguity characteristic of the task, especially for emotions that are highly correlated.

4.5.5 Influence of Parameter (α)

The model was trained with BCE loss and with LCA loss via a weight (α), whose impact on the results is presented in Figure 4.5. It should be mentioned that this analysis was performed on the validation set of SemEval-2018 data set. The lower bound (i.e., 0.0) indicates that the model was trained only with the BCE loss, whereas the upper bound (i.e., 1.0) indicates that it was trained only with the LCA loss. When the value of α increased from 0.0 to 0.5, the results first improved considerably and then gradually deteriorated apart from the results of the macro F1-score. The results of BCE loss favoured the micro F1- and Jaccard score, whereas the results of LCA loss favoured the macro F1-score. However, integrating LCA with BCE can balance the results across all three metrics, resulting in strong performance. The best results were achieved on almost all metrics when the value of α was set to 0.2. Thus, we set the value of parameter α to 0.2 for all experiments reported in our work.

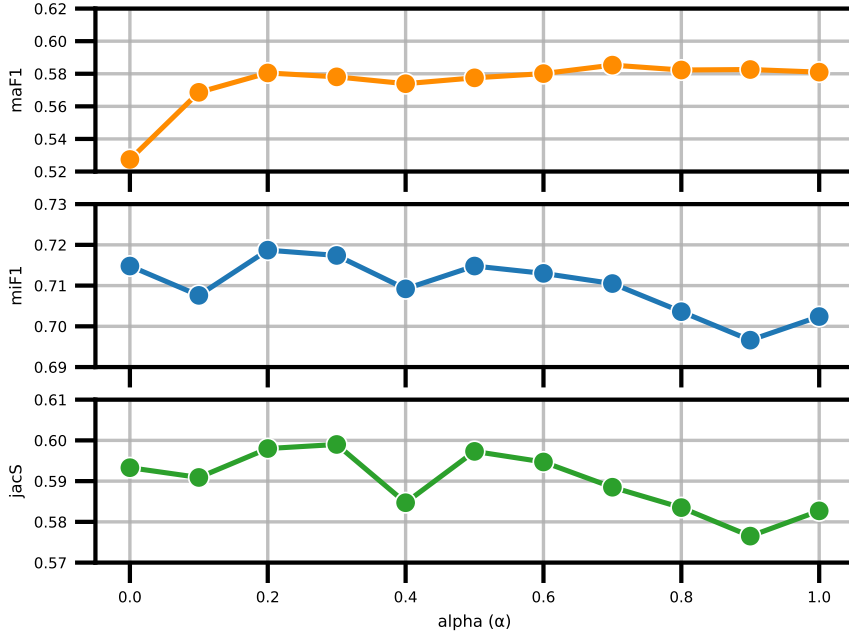


Figure 4.5: Sensitivity analysis of the parameter (α). Note that $\alpha = 0.0$ means that only BCE loss is used in training SpanEmo, whereas $\alpha = 1.0$ means that only LCA loss is utilised in training it.

4.6 Summary

In this chapter, we addressed our first research question (**RQ#1**) and developed a novel neural model aimed at casting multi-label emotion classification as a span-prediction problem. We proposed “SpanEmo” to learn emotion correlations and emotion-specific associations in an end-to-end fashion.

SpanEmo consists of four components: 1) an input encoding, 2) an input encoding network, 3) a feed-forward network and 4) an output selection layer. The first component included both an input instance (i.e., input segment) and emotion classes (i.e., label segment). In this respect, the emotion classes and input instance were concatenated together, and this was intended to help the encoding network interpolate between the two segments. The output of the encoder was then passed into the feed-forward network to transform the hidden representation for each token into a single score. The tokens belonging to the label segment were then selected for making the final predictions. We trained SpanEmo with LCA loss to enable the model to take emotion correlations into account during the training phase.

We evaluated our model on the SemEval-2018 multi-label emotion classification, which is based on labelled data from tweets in English, Arabic and Spanish. We

demonstrated that our proposed method outperforms prior approaches reported in the literature on three languages (i.e., English, Arabic and Spanish). Our empirical evaluation also demonstrated the utility and advantages of our method for multi-label emotion classification, specifically the addition of emotion classes to the input sentence, which helped the model learn emotion-specific associations and increase its performance. Finally, training our method with LCA loss jointly led to better results, showing the benefits of integrating the relations between emotions into the loss function.

We conducted a large number of analyses by focusing on four aspects: 1) prediction of multiple emotions, 2) learning emotion-specific associations both at the word-level and tweet-level, 3) qualitative study of the model predictions, 4) label correlations. Our observation regarding the first analysis demonstrated that SpanEmo achieved consistent improvement as the number of co-existing emotions increased. In addition, SpanEmo showed that it was able to learn meaningful associations between emotion classes and words automatically. Our observation regarding the third analysis revealed that SpanEmo can predict all correct labels, which can be attributed to the use of label correlation aware loss. Furthermore, SpanEmo demonstrated that it can group similar emotions, as shown in Figure 4.4, and can also learn correlations between emotions effectively.

From the above discussion, we concluded that the task of recognising emotion expressions can be modelled better by taking emotion-specific associations and emotion correlation into account. Taking these two proprieties into consideration can address the problem of ambiguity between highly correlated emotions. Learning emotion-specific associations can further help TER models to predict the correct label set as well as providing a mechanism for interpretation. Our model can be easily applied to other ER corpora and languages without requiring any modification in its architecture.

Finally, the main attributes of our work can be summarised as follows: 1) the addition of emotion classes to the input instance, 2) the selection of predictions from the label segment directly, 3) the modelling of multiple co-existing emotions and 4) the independence from emotion lexicons as well as theories of emotion in learning both emotion correlations and associations.

Chapter 5

Case Studies

In the previous chapter, we introduced our neural model (i.e., SpanEmo) for multi-label emotion classification, which can take advantage of both emotion-specific associations and emotion correlations in a tweet. The proposed method was specifically evaluated against an emotion corpus that was collected from social media data. In this chapter, we aim to investigate the benefits of emotional knowledge and the adaption of the SpanEmo architecture to two case studies, which both address our second research question (**RQ#2**). The two case studies are adverse drug reactions (ADR) and an application of mental health (i.e., depression). The rationale for choosing these two case studies is their relevance to emotion expressions. This chapter demonstrates that emotional knowledge can have a direct influence on downstream applications and that SpanEmo can be easily adapted to other tasks beyond emotion. In the first part of this chapter, we describe experiments related to the first case study, and describe experiments related to the second case study in the second part.

In this chapter, we tackle each case study differently, due to the complexity and setup of each case. The first case study is simply a binary classification problem, focusing on determining the presence and absence of ADRs. Whereas, the second case study contains 21 questions, each of which is a multi-class classification problem. In addition, the data size of the second case study is scarce, only consisting of 90 users. The above-mentioned reasons make the second case study more complex than the first one, and require an alternative solution to address it. We conclude the chapter with a summary of each case study and report on the analysis of our experiments for both case studies. It is worth pointing that this chapter is drawn from both [Alhuzali and Ananiadou \(2019\)](#) and [Alhuzali et al. \(2021\)](#)¹.

¹Only the work undertaken by Hassan Alhuzali is included in this chapter.

5.1 Use Case on Adverse Drug Reaction

The availability of large-scale and real-time data on social media has motivated research into adverse drug reactions (ADRs). ADR classification helps to identify negative effects of drugs, which can guide health professionals and pharmaceutical companies in making medications safer and advocating patients' safety. Based on the observation that in social media, negative sentiments/emotions are frequently expressed towards ADRs, this study presents a neural model that combines emotion knowledge with transfer learning techniques to improve ADR detection in social media postings. Our model is firstly trained to classify sentiment/emotion in tweets concerning current affairs, using two SemEval corpora. We then apply transfer learning to adapt the model to the task of detecting ADRs in social media postings. We show that, in combination with rich representations of words and their contexts, transfer learning is beneficial, especially given the large degree of vocabulary overlap between the current affairs posts in the SemEval17-task4A corpus and posts about ADRs. We compare our results with previous approaches, and show that our model can outperform them by up to 5% F-score.

5.1.1 Introduction

Social media generate a huge amount of data for health and are considered to be an important source of information for pharmacovigilance (Sloane et al., 2015; Harpaz et al., 2014; Kass-Hout and Alhinnawi, 2013). ADR detection from social media has attracted a large amount of interest as a source of information regarding morbidity and mortality. In this respect, social networks are an invaluable source of information, allowing us to extract and analyse ADRs from health communication threads between thousands of users in real-time. Several ADR models have utilised features related to the sentiment of words to boost their model performance (Wu et al., 2018; Kiritchenko et al., 2017; Alimova and Tutubalina, 2017; Korkontzelos et al., 2016; Sarker and Gonzalez, 2015). Korkontzelos et al. (2016) analyse the impact of sentiment analysis features on extracting ADR from tweets. The authors observed that users frequently express negative sentiments when tweeting/posting about ADRs and that the use of sentiment-aware features could improve ADR sequence labelling and classification.

We observe that the language used to express sentiment/emotion is often common across different domains/topics. Consider, for example, the tweet "I hate how

[drug_name] makes me over think everything and it makes me angry about things that I shouldn't even be angry about". The keywords used in this tweet to express the author's negative sentiment/emotion towards an ADR, i.e., hate and anger, are not specific to ADRs, and may be used to express sentiment/emotion towards many different kinds of topics. Based on this observation, we hypothesise that we can leverage transfer learning techniques by using sentiment/emotion data to boost the detection of ADRs. Our ADR detection model firstly trains a classifier on the SemEval17-task4A data and the SemEval-2018 multi-label data, which consist of Tweets on the subject of current affairs. This pre-trained classifier is then adapted to the task of detecting ADRs, using datasets of social media postings that are annotated according to the presence or absence of ADRs. To our knowledge, *this is the first attempt to apply transfer learning techniques to adapt a sentiment analysis and an emotion classifier to the task of detecting ADRs*. In contrast to previous research, we use generalised neural methods that avoid the use of hand-crafted features, since these are time-consuming to generate, and are usually domain-dependent. We also explore different fine-tuning methods to determine which one performs best in our scenario. Our main contributions, which are stated in Chapter 1, are also summarised below:

- **Experiment (1):** an initial experiment that investigates the benefits of the pre-trained classifier on sentiment data in improving the detection of ADRs. In this experiment, we propose a novel neural model that detects ADRs by firstly learning to classify sentiment, using a publicly available corpus of Tweets that is annotated with sentiment information and then using transfer learning to adapt this classifier to the detection of ADRs in social media postings.
- **Experiment (2):** a second experiment that *adapts our SpanEmo architecture*. In this experiment, we follow the findings from the initial experiment that helps improve the detection of ADRs by pre-training a model on multi-label emotion data and then applying the pre-trained model to ADRs.
- An in-depth analysis that illustrates the advantages and utility of using emotion data and SpanEmo model in enhancing the detection of ADRs.

The first part of this chapter is focused on the first case study (i.e., ADRs) and is organised as follows: Section 5.1.2 provides a review of related work. Section 5.1.3 presents the two datasets used to create our model. Section 5.1.4 describes our method and model for the initial experiment, whereas Section 5.1.5 describes our method and

model for the second experiment. We report on the analysis of results in Section 5.1.4.2 and Section 5.1.5.2 for the initial experiment and the second experiment, respectively.

5.1.2 Related Work

There is a growing body of literature concerned with the detection and classification of ADRs in social media texts (Wang et al., 2018; Huynh et al., 2016; Ebrahimi et al., 2016; Liu and Chen, 2015). Recent work has employed sentiment analysis features to improve the classification of ADRs (Wu et al., 2018; Kiritchenko et al., 2017; Alimova and Tutubalina, 2017; Korkontzelos et al., 2016; Sarker and Gonzalez, 2015).

Nikfarjam et al. (2015) exploited a set of features, including context features, ADR lexicon, part of speech (POS) and negation, to enhance the performance of ADR extraction. The authors chose Conditional Random Fields as their classifier (CRF). Korkontzelos et al. (2016) followed the same research hypothesis, but focused on the evaluation of sentiment analysis features as an aid to extracting ADRs, based on the correlation between negative sentiments and ADRs. Alimova and Tutubalina (2017) built a classification system for the detection of ADRs for which they used a Support Vector Machine (SVM), instead of CRF. The authors also explored different types of features, including sentiment features and demonstrated that they improved the performance of ADR identification. Wu et al. (2018) utilised a set of hand-crafted features (i.e. sentiment features learned from lexica), similar to all of the other studies introduced above. However, the main difference is that the model is based on a neural network architecture, including word and character embeddings, Convolutional neural network (CNN), Long Short-Term Memory (LSTM) and multi-head attentions. This was the best performing system in the 2018 ADRs shared-task², which is part of the social media mining for health workshop (SMM4H).

In contrast to the models proposed in the above studies, it is possible to leverage sentiment analysis features automatically, without relying on any hand-crafted features. One common approach is to pre-train a classifier on a corpus annotated with sentiment information and then to adapt this pre-trained classifier to the detection of ADRs. The advantage of this approach is that the target model only needs access to the pre-trained model, but not the original sentiment corpus, which can be important for storage and data regulation issues. This method has been investigated by various researchers (Devlin et al., 2018; Howard and Ruder, 2018; Felbo et al., 2017). Felbo

²<https://healthlanguageprocessing.org/smm4h/>

et al. (2017) learned a rich representation for detecting sentiment, sarcasm, and emotion using millions of emojis’ dataset, acquired from Twitter. They demonstrated that this approach performs well and can achieve results that are competitive with state of the art systems. Recently, Devlin et al. (2018) built a deep bidirectional representation from transformers known as (BERT), which can be fine-tuned to different target tasks with an additional output layer.

Compared to the above approaches, our work uses a simpler network architecture and does not require any feature engineering. Furthermore, we take advantage of transfer learning techniques acquired knowledge from sentiment analysis data. Our work is motivated by Felbo et al. (2017) who constructed a pre-trained classifier on emoji’s data and then adapted it to sentiment and emotion detection. The full details of our architecture are described in Section 5.1.4.1.

5.1.3 Data

Datasets	#ADRs	#None
Training		
DailyS.	900	417
Twitter	390	384
Validation		
DailyS.	600	278
Twitter	260	256
Test		
DailyS.	533	225
Twitter	236	192

Table 5.1: Data statistics (DailyS. = DailyStrength)

ADR Corpora. Several datasets have been created for ADRs. Some of these are gathered from specialised social networking forums for health (Thompson et al., 2018; Sampathkumar et al., 2014; Yates and Goharian, 2013; Yang et al., 2012), while others are collected from social media (Ginn et al., 2014; Jiang and Zheng, 2013; Bian et al., 2012). In this work, we chose a widely used dataset (i.e., containing postings from Twitter and DailyStrength³) that are annotated according to the presence or absence of ADRs in each post (Nikfarjam et al., 2015). The authors partitioned the data into a training (75%) and test (25%) sets. We further divided the training set into a 60% for

³DailyStrength is a specialised social networking website for health.

training and 40% for validation. The validation set is used to develop our model before it is evaluated on the original test set (i.e. 25% of the complete corpus). Our model is designed to perform binary classification, to determine whether or not a given tweet or post mentions an ADR. Table 5.1 presents the number of tweets/posts belonging to each category in the three different partitions of the data. More detailed information about the datasets can be found in [Korkontzelos et al. \(2016\)](#) and [Nikfarjam et al. \(2015\)](#). Figure 5.1 presents the pipeline of the two conducted experiments in the first part of this chapter, as described below in greater detail.

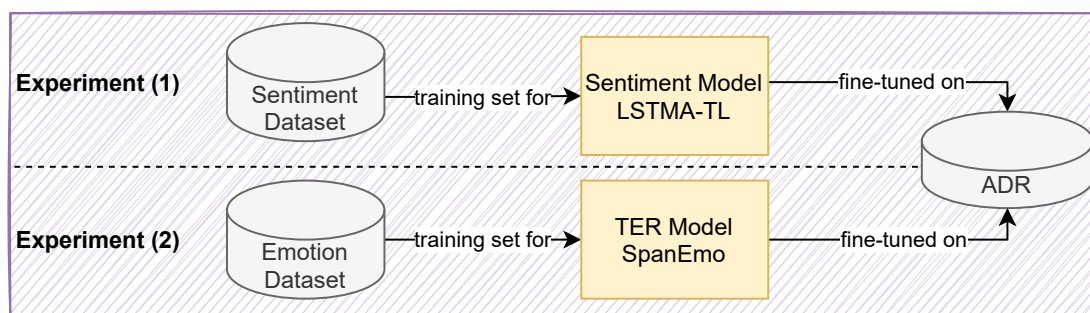


Figure 5.1: An illustration of the two conducted experiments in the first part of this chapter.

Sentiment Analysis corpus. In experiment (1), we firstly train a sentiment analysis model on Twitter data from the SemEval17-task4A, which focuses on classifying the sentiment polarity of tweets on the subject of current affairs into pre-defined categories, e.g. positive, negative, and neutral. The dataset is partitioned into a training set of 50,000 tweets and a test set of 12,000 tweets ([Rosenthal et al., 2017](#)). A description of the model is provided in Section 5.1.4.1.

Multi-label Emotion corpus. In experiment (2), we firstly train a multi-label emotion classification model on Twitter data from the SemEval-2018, which focuses on classifying a tweet into multiple emotions (e.g., *anger*, *fear*, *disgust*, *love*, etc.). The dataset is partitioned into a training set of 6,838 tweets, a validation set of 886 and a test set of 3,259 tweets ([Mohammad et al., 2018](#)). A description of the model is provided in Section 5.1.5.1.

Preprocessing. Since Twitter data possesses specific characteristics, including informal language, misspellings, and abbreviations, we pre-process the data before applying the methods described in the next section. We use the “ekphrasis” tool ([Baziotis et al., 2017](#)) that is specifically designed for the Twitter domain. The tool provides

a number of different functionalities, such as tokenisation, normalisation, spelling-correction, and segmentation. We use ekphrasis to tokenise the text, to convert words to lower-case, to correct misspellings, and to normalise user mentions, urls and repeated-characters.

5.1.4 Experiment (1)

5.1.4.1 Proposed Approach

This section discusses our model architecture, which is composed of two stages: the first stage involves building a sentiment analysis model, while the second stage adapts this model to a target task, which in our case is the detection of ADRs. We describe our architectures in the following subsections.

Network Architecture. Our architecture consists of an embedding layer (Mikolov et al., 2013a), a Long Short-Term Memory (LSTM) layer (Hochreiter and Schmidhuber, 1997), a self-attention mechanism (Bahdanau et al., 2014) and a classification layer. Figure 5.2 depicts the network architecture of our model.

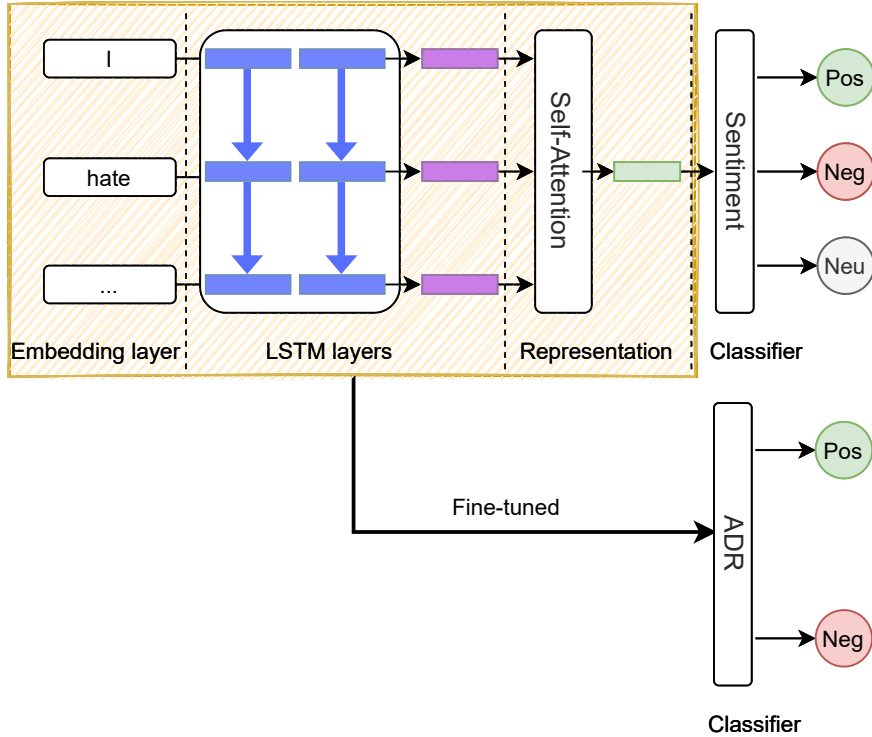


Figure 5.2: Description of our framework.

In our different experiments, we use both an LSTM and a bi-directional LSTM

(BiLSTM). Both are able to capture sequential dependencies especially in sequential data, of which language can be seen as an example. The model’s weights are initialised from the *word2vec* embedding with 300 dimensional size, which was trained on 550M English Twitter messages⁴. Additionally, the model consists of two LSTM/BiLSTM layers. For regularisation, we apply a dropout rate of 0.2 and 0.3 on the embedding output and after the second hidden layer, respectively, to prevent the network from over-fitting to the training set (Hinton et al., 2012). We choose Adaptive Moment Estimation Algorithm (ADAM) (Kingma and Ba, 2014) for optimisation and select 0.001 as the learning rate. We train the network for 10 epochs and only the best performing cycle is retained. It should be mentioned that the above set of hyper-parameters was determined using the validation set. Table 5.2 summarises the network architecture and hyper-parameters.

Hyper-Parameter	Value
embed-dim	300
layers	2
units	{200, 300, 400*}
batch size	{32*, 64}
epochs	10
sequence length	30
embed-dropout	0.2
lstm-dropout	{0.3, 0.4*}
learning rate	0.001

Table 5.2: Network architecture and hyper-parameters. The asterisk (*): denotes the best performing setting

Embedding layer. T is a sequence of words $\{w_1, w_2, \dots, w_n\}$ in a tweet/post and each w_i is a d dimensional word embedding for the i -th word in the sequence, where n is the number of words in the tweet. T should have the following shape n -by- d .

LSTM/Bi-LSTM layer. An LSTM layer takes as its input a sequence of word embeddings and generates word representations $\{h_1, h_2, \dots, h_n\}$, where each h_i is the hidden state at a time-step, retaining all the information of the sequence up to w_i . Additionally, we experiment with a BiLSTM where the vector representation is built as a concatenation of two vectors, the first running in a forward direction \vec{h} from left-to-right and the second running in a backward direction \overleftarrow{h} from right-to-left $h_i = [\vec{h}; \overleftarrow{h}]$.

⁴<https://github.com/alexandra-chron/ntua-slp-semeval2018>

Self-attention. A self-attention mechanism has been shown to attend to the most informative words within a sequence by assigning a weight a_i to each hidden state h_i . The representation of the whole input is computed as follows:

$$e_i = \tanh(W_h h_i + b_h) \quad (5.1)$$

$$a_i = \text{softmax}(e_i) \quad (5.2)$$

$$r = \sum_{i=1}^T a_i \cdot h_i \quad (5.3)$$

, where W_h, b_h are the attention's weights.

Classification layer. The vector r is an encoded representation of the whole input text (i.e. a tweet or post), which is eventually passed to a fully-connected layer for classification. A binary classification decision is made according to whether or not the input text mentions ADRs.

Transfer Learning. After training the sentiment classification model, we exclude its output layer and replace it by an ADR output layer. Finally, the network is fine-tuned to detect the ADRs adopting the same architecture and hyper-parameters as the original model. We analyse the fine-tuning methods in Section 5.1.4.3.

5.1.4.2 Results

Table 5.3 presents the performance of our models in terms of F-score of the positive class (i.e., instances labelled as containing the mentioning of ADRs)⁵, and compares these to the three of the best performing models from recently published research. For our own results, we report the results of three different experiments. Firstly, the baseline (LSTMA) is trained to detect ADRs using the ADR datasets mentioned above, without the use of transfer learning. The other two models (LSTMA-TL and BiLSTMA-TL) apply transfer learning, making use of pre-training of a sentiment analysis model using the SemEval17-task4A dataset. These latter two models differ in terms of whether they use a single direction or bi-directional LSTM, respectively. For experiments related to previous work, we replicated the three models following their details as described in Huynh et al. (2016), Alimova and Tutubalina (2017) and Wu et al. (2018).

⁵The reason for choosing this metric is because the task of ADRs is binary.

Datasets	DailyS.	Twitter
Models	F1	F1
Previous Work		
Huynh et al. (2016)	0.89	0.75
Alimova and Tutubalina (2017)	0.89	0.78
Wu et al. (2018)	0.90	0.79
Contextualised Embeddings		
Devlin et al. (2018)	0.89	0.82
This Work		
LSTMA (baseline)	0.90	0.79
LSTMA-TL	0.92	0.82
BiLSTMA-TL	0.92	0.81

Table 5.3: Comparison of our models to those reported in previous work. **LSTMA**: refers to LSTM with self-attention mechanism, while **LSTMA-TL**: means the same thing except the addition of the transfer learning model. **BiLSTM-TF**: uses a BiLSTM with transfer learning model. Best: bold.

Previous Work. [Alimova and Tutubalina \(2017\)](#) used an SVM model with different types of hand-crafted features (i.e. sentiment and corpus-based features). Their model performed to a high degree of accuracy, which is not surprising, due to the power of the SVM model when applied to small data. Similarly, [Huynh et al. \(2016\)](#) exploited different neural networks, i.e CNN and a combination of both CNN and Gated Recurrent Units (GRU). They found that CNN obtained the best performance. For this reason, the results reported in Table 5.3 are those obtained for the CNN model. On the Twitter dataset, the performance of the CNN is even lower than the performance of our baseline model on this dataset. However, the performance on the DailyStrength dataset is considerably higher. The model developed by [Wu et al. \(2018\)](#) obtained the best results among the three compared systems; indeed, the results reach the same level as our baseline system. However, it is important to note that in contrast to our model architecture, that of [Wu et al. \(2018\)](#) is more complex and it relies on hand-crafted features as well as deep neural architectures.

Contextualised Embeddings. In this work, we also compared our model to contextualised embedding (i.e. BERT) since it has been shown to achieve high results for various NLP tasks, including text classification ([Devlin et al., 2018](#)). We use the open-source PyTorch implementations⁶ and only consider the pre-trained “bert-base-uncased” model. The model is trained on the default hyper-parameters except that the

⁶<https://github.com/huggingface/pytorch-pretrained-BERT>

number of batch-size and sequence length are chosen as follows 32 and 30, respectively, to match our model hyper-parameters for these two values. As shown in Table 5.3, BERT model achieves the same performance as our best model “LSTMA-TL” when applied to the Twitter data, although its performance is 3% lower than our best performing model when applied to the DailyStrength dataset. Even though transfer learning is beneficial, it can achieve better performance when learned from a related domain to the problem under investigation.

This Work. As Table 5.3 demonstrates, our proposed model is able to outperform all compared systems on the DailyStrength dataset, and all systems apart from BERT when applied to the Twitter Dataset. More specifically, the “LSTMA-TL” obtained the best results, thus demonstrating the utility and advantages of transfer learning techniques. The “BiLSTMA-TL” also demonstrates competitive results for the DailyStrength dataset, but it is 1% less than the “LSTMA-TL” for the Twitter dataset. This may be due to the size of data and the architecture used in this work. Although the sentiment analysis model is trained on Twitter data, our ADR detection system still demonstrated substantial improvement on the DailyStrength dataset. Specifically, we obtained 3% and 2% improvement over our baseline model (i.e. LSTMA) on the Twitter and DailyStrength datasets, respectively. Even though our experiments are based on a small dataset, the model demonstrated strong performance for ADR classification. Recent research claims that transfer learning techniques (i.e. fine-tuning) are beneficial for downstream tasks even if the target data size is small (Howard and Ruder, 2018; Alhuzali et al., 2018d).

5.1.4.3 Analysis

Impact of fine-tuning. We evaluate different methods to fine-tune our model, i.e. Last, Chain-thaw, Full and Simple Gradual unfreezing (GU). The first three techniques are adopted from Felbo et al. (2017) while the fourth one is described by Chronopoulou et al. (2019). “Last” refers to the process of fine-tuning only the last layer (i.e. output layer), while the other layers are kept frozen. “Chain-thaw” method aims to firstly fine-tune each layer independently and then fine-tuned the whole network simultaneously. “GU” is similar to the Chain-thaw method except that the fine-tuning is performed at different epochs. In this work, we experimented with these methods and selected the one that achieved the highest results for both datasets (i.e. Twitter and DailyStrength). The results of these four methods are reported in Figure 5.3.

“Last”, which is the standard technique in fine-tuning, achieved the lowest performance; this is not surprising, because it contains the least general knowledge. In contrast, “Chain-thaw” achieved better results than “Last”. The “Full” and “GU” obtained the best results for ADR classification. When we fine-tuned the whole network, we modified the “Full” method such that the embedding layer is frozen and we called it “Full-no-Emb”, instead. The intuition behind this is that the embedding layer computes a word-based representation, which does not take into account the context of a word. This method obtains the best performance for both Twitter and DailyStrength datasets.

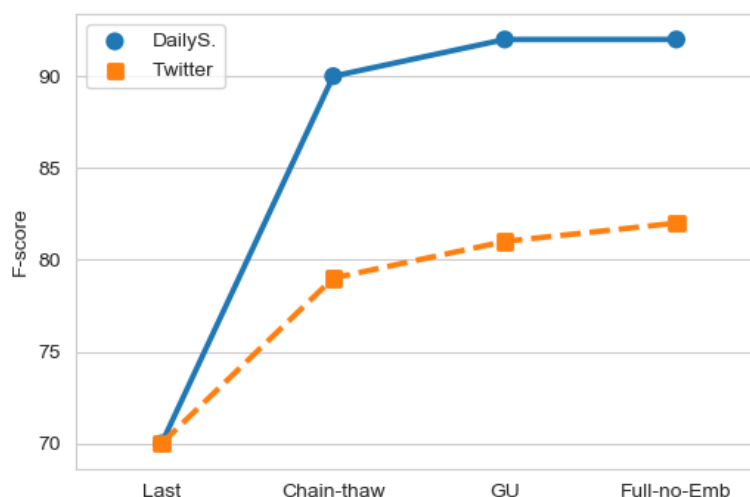


Figure 5.3: F-score for our model with a different set of fine-tuning methods.

Word Coverage. We observed that the vocabularies used in the sentiment analysis dataset and the ADR datasets share a large proportion of common words. To further investigate this, we measured the degree of common word coverage between the training and test parts of each dataset (i.e. Twitter and DailyStrength). The SemEval17-task4A training set is also included in this comparison. It should be noted that we compute the word coverage after pre-processing the data. Table 5.4 shows percentage of shared-vocabulary between the datasets. As shown in Table 5.4, the percentage of shared words between the training and test set of ADR Twitter data is 56.50%, while it is 74.22% between the SemEval17-task4A training set and the ADR Twitter test set. A similar pattern is also observed for the DailyStrength dataset, although there is a greater proportion of shared vocabulary between the training and test sets of DailyStrength. The vocabulary of the SemEval17-task4A dataset exhibits a large degree of overlap with the test sets of both Twitter and DailyStrength.

Dataset	Train	SE17-4A	Δ %
Twitter test	56.50%	74.22%	17.72%
DailyS. test	68.03%	78.22%	10.19%

Table 5.4: Word coverage. “SE17-4A”: corresponds to the training set of the SemEval17-task4A. $\Delta\%$: represents the difference between the two percentages for each dataset in a row.

We hypothesise a number of reasons could account for this finding. Intuitively, users often use non-technical keywords when they post or tweet about ADRs. In other words, they do not employ terms found in medical lexicons. This allows users to express their opinion towards ADRs using terms which may be used to express sentiment towards other different topics. Additionally, several datasets have been collected for ADRs. However, most of them have not been made available for the research community. In contrast, there are dozens of sentiment analysis datasets available online, including SemEval17-task4A⁷, Yelp reviews⁸, Amazon reviews⁹ and Stanford¹⁰, among others. Thus, this confirms our initial observations and helps to reinforce that the ADR model can benefit from the proliferation of sentiment analysis data available online, which is the primary motivation of this work.

Error Analysis. We experiment with small data in this work and this may limit our interpretation and analysis in this section. Nevertheless, performing error analysis can reveal some strengths and weaknesses of the proposed models and identify room for future work. For this analysis, we selected examples which are incorrectly classified by the proposed model in this paper (i.e. LSTMA-TL) and previous work (i.e. (Huynh et al., 2016; Alimova and Tutubalina, 2017)). Figure 5.4 presents the number of false positive and false negative classifications for each model. As can be seen in Figure 5.4a, the number of miss-classified examples as false negative is higher than false positive for the DailyStrength dataset, while the opposite pattern is observed for the Twitter dataset as shown in Figure 5.4b. Our model also demonstrated balanced error classifications for both false positive and false negative. In contrast, the other two models, proposed by previous research, obtained unbalanced error classifications except Alimova and Tutubalina (2017)’s model achieved quite balanced errors for the Twitter dataset. For future work, it might be useful to investigate different ensemble

⁷<http://alt.qcri.org/semeval2017/task4/index.php?id=data-and-tools>

⁸<https://www.yelp.com/dataset>

⁹<https://s3.amazonaws.com/amazon-reviews-pds/readme.html>

¹⁰<https://nlp.stanford.edu/sentiment/index.html>

methods that can help to reduce the false positive and false negative classifications and improve the classification of ADR.

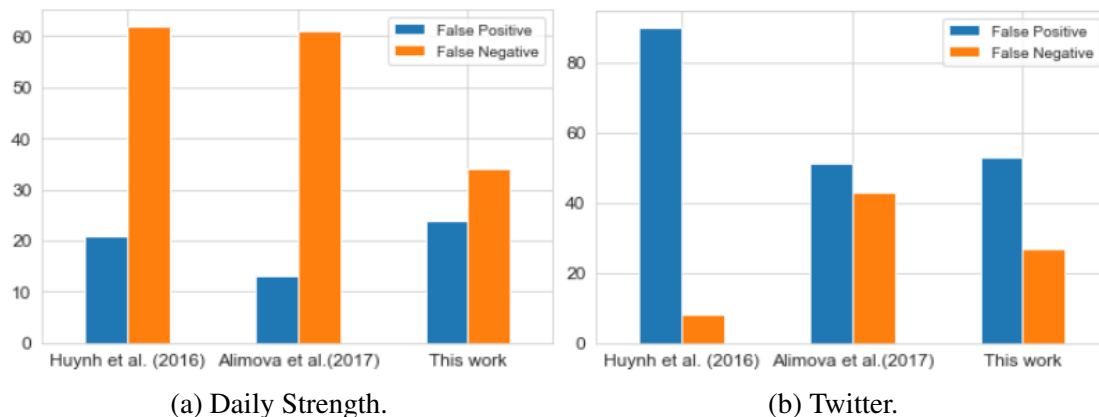


Figure 5.4: The number of miss-classified examples by the proposed models of this work and previous research. This work: refers to the proposed model in this chapter (i.e. LSTMA-TL).

In addition, we analysed examples (described below) classified correctly by our model, and observed that our model is able to classify examples carrying non-specific keywords to ADRs, but to sentiments in general. This shows the importance of sentiment features to ADRs. Examples 1-3 below illustrate the instances that are correctly predicted by our model. The first two examples are part of the Twitter test set, while the third example is part of the DailyStrength test set.

- Example 1: is it hot in here or is [durg_name] just kicking in?.
- Example 2: anyone ever taken [durg_name]? i've been on it for a week, not too sure how i feel about it yet. anyone want to share their experience?.
- Example 3: loved it , except for not being able to be woken up at night . . yeah that blew.

We also inspected examples that our model failed to correctly classify. For instance, example (4) below was extracted from the Twitter test set and it was predicted as negative for the presence of ADR, whereas the true label is positive for the presence of ADR. Example (5) also illustrates the same observation, but is part of the DailyStrength test set. We anticipate that our model failed to classify examples (4) and (5) due to the lack of context and unambiguous keywords. Example (4) can also be interpreted as either positive or negative for the presence of ADRs. This may explain that the true label can be sometimes misleading and requires further examination.

- Example 4: moved on to something else when it quit working.
- Example 5: i'm with you. even though the [durg_name] works, i still don't feel fully human.

5.1.5 Experiment (2)

5.1.5.1 Proposed Approach

We aim to apply our previously introduced model in Chapter 4, to the adverse drug reaction problem for social media data. In a similar vein, the architecture consists of four components: an input encoding, an input encoding network, a feed-forward network and, finally, an output classification layer. The main differences with our previous model are featured only in the encoding of input and the output layer. In addition, we follow our previous findings discussed in Section 5.1.4, and utilise the best performed transfer learning strategy. We describe these components in more detail below.

Input Encoding. One distinction between the task of multi-label emotion classification and ADR classification lies in the number of labels associated with each input. For the former, each input can take more than one emotion label, whereas the latter can only take a single label (i.e., presence/absence of ADR). We formulate the input as follows:

$$\mathbf{H}_i = \text{Encoder}([\text{CLS}] + |\mathbf{C}| + [\text{SEP}] + \mathbf{x}_i), \quad (5.4)$$

where $\{[\text{CLS}], [\text{SEP}]\}$ are special tokens and $|\mathbf{C}|$ denotes the size of ADR labels $\in \{pos, neg\}$, which are encoded into the input as `positive_reaction` and `negative_reaction`, respectively. \mathbf{x}_i represents the i^{th} input representation.

Classification Layer. One of the goals of SpanEmo is to learn correlations between emotions, and this has been leveraged by using the co-occurrence statistics among different emotion categories from the emotion corpus. However, this is not needed for the ADR task due to the number of outputs being binary. Thus, we exclude the correlation component for this task and only use the output from the feed-forward component directly for prediction. We also replace the sigmoid activation function by the softmax function. This can help the model in selecting either one of the two labels.

$$\hat{\mathbf{y}} = \text{softmax}(\text{FFN}(\mathbf{H}_i)), \quad (5.5)$$

where \mathbf{H}_i represents the hidden representation for each input and FFN is responsible

for transforming the hidden representation of each token into a single score. Tokens correspond to the label segment are selected for final predictions and the selected predictions are then passed into a softmax activation function.

Transfer Learning. Based on our previous evaluation and analysis, we demonstrate that transfer learning is beneficial for ADR and it can achieve better performance when learned from a related domain to the task of ADR. For this reason, we initialise the model weights using SpanEmo trained on the multi-label emotion corpus. We hypothesise that if training the model on sentiment data improves the performance, then training it on an emotion corpus should help too, due to the fine-grained nature of the corpus which captures more nuances beyond those existing in the sentiment corpus.

5.1.5.2 Results and Analysis

Table 5.5 shows the results of our model in terms of F-score, and compares it to the five of the best performing models from recently published research on both Twitter and Daily Strength data sets (see detailed description about these four models in Section 5.1.4.2).

Datasets	DailyS.	Twitter
Models	F1	F1
Previous Work		
Huynh et al. (2016)	0.89	0.75
Alimova and Tutubalina (2017)	0.89	0.78
Wu et al. (2018)	0.90	0.79
Devlin et al. (2019)	0.89	0.82
Alhuzali and Ananiadou (2019)	0.92	0.82
This Work		
ours w/ TL	0.95	0.84
ours w/o TL	0.93	0.84

Table 5.5: Comparison of our method to those reported in prior work. *TL* stands for transfer learning.

Our model outperforms all previously developed models on the two data sets, with a high improvement of 3% for the Daily Strength data set and 2% for the Twitter data set. Consider the example “I hate how *drug-name* makes me over think everything and it makes me angry about things that I shouldn’t even be angry about”. A sentiment

model is more likely to predict a negative sentiment for this example (i.e., coarse-grained), whereas an emotion model is more likely to provide fine-grained predictions (e.g., anger and disgust). The fine-grained predictions carry additional useful information for the task of ADR beyond those found in coarse-grained predictions. This confirms our hypothesis that emotion features are not only beneficial, but also more useful for ADR classification.

Testing Generalisability. To evaluate the generalisation capability of the proposed approach, we performed training on Twitter data and testing on Daily Strength, as well as vice versa. For this evaluation, we select two models as both achieve high results on the two datasets (i.e., SpanEmo and BERT). Figure 5.5 presents the results in terms of F1-score. It is worth noting that when the model is tested on the same test

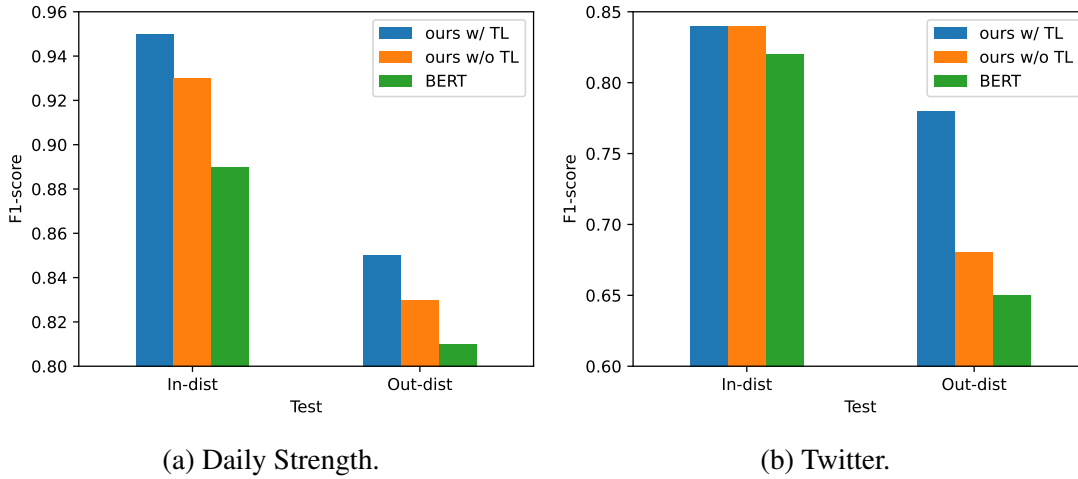


Figure 5.5: The results of Daily Strength and Twitter when evaluated against the same test set (In-distribution) vs the other test set (Out-distribution).

set, we call this “In-distribution”, whereas we call it “Out-distribution” when tested on the other test set. We obviously see some drop in performance, but the drop of our model is not as much as that of BERT. For example, BERT’s performance drops up to 0.65, whereas our model drops up to 0.78, with a marginal difference of 11%. In addition, we notice that the drop in performance is higher for the Daily Strength data set than the Twitter data set. This is attributed to the fact that our model is pre-trained on emotion data collected from Twitter. Although our model’s result has dropped down, it still demonstrates strong performance, which can be beneficial for cases when there is no or small labelled data to use. This evaluation again shows the potential of transfer learning, especially when pre-trained on a related domain to the one under investigation, as well as the potential of our model, which makes use of descriptive label names

in order to prime the model to focus on associations between the labels and input.

5.2 Use Case on Mental Health

In the first part of this chapter, we investigated both the benefits of emotional knowledge and the adoption of SpanEmo model to adverse drug reaction. We demonstrated the advantages and utility of sentiment/emotion corpora, transfer learning techniques and more importantly “SpanEmo” for improving the performance of ADR detection in social media. We now turn to take advantages of the findings discovered from the ADR experiments and adapt them to an application of mental health (i.e., depression).

Predicting and understanding how various mental health conditions present online in textual social media data has become an increasingly popular task. The main aim of using this type of data lies in utilising its findings to prevent future harm as well as to provide help. In the second part of this chapter, we describe our approach and findings in participating in sub-task 3 of the CLEF e-risk shared task. Our approach follows-up our findings described in the first part of this chapter, more specifically the use of SpanEmo model and transfer learning. We use both to extract features for all user’s posts and then feed them into a classifier. We achieve better results than prior approaches on this shared task, with an average hit rate of 35.65% and an average depression category hit rate of 48.85%.

5.2.1 Introduction

There have been many previous iterations of the CLEF e-risk shared task over recent years (Losada et al., 2020, 2019; Losada and Crestani, 2016), where the collective goal of these tasks is to connect mental health issues to language usage. However, previous work in this area has not been able to produce convincing solutions that connect language to psychological disorders and it therefore remains a challenging task to produce accurate models. This year’s CLEF e-risk-2021 shared task (Parapar et al., 2021) provided three different tasks, which are focused on pathological gambling (T1), self-harm (T2) and depression (T3). We only focus on T3 of the shared-task.

Depression is one of the most common mental disorders, affecting millions of people around the world (James et al., 2018). The growing interest in building effective approaches to detect an early sign of depression has been motivated by the proliferation of social media and online data, which have made it possible for people to communicate and share opinions on a variety of topics. In this respect, social media are an invaluable source of information, allowing us to analyse users who present signs of depression in real-time. Taking advantage of social media data in early detection of

depression helps to benefit individuals who may suffer from it and their loved ones, as well as to give them access to professional assistance who could advocate their health and well-being (Guntuku et al., 2017). In this work, we describe our contribution to T3 of eRisk-2021, which focused on detecting early risk of depression from a thread of user posts on Reddit.

The second part of this chapter is focused on an application of mental health (i.e., depression) and is organised as follows: Section 5.2.2 provides a review of related work. Section 5.2.3 discusses some experimental details, including the data/task settings, evaluation metrics and our proposed method. Section 5.2.3.2 discusses our results and analyses, while Section 5.2.3.3 highlights negative results.

5.2.2 Related Work

There is a large body of literature on early sign detection of depression (Guntuku et al., 2019; Aragón et al., 2019; Chen et al., 2018; Schwartz et al., 2014; Wang et al., 2013; Van Rijen et al., 2019; Wang et al., 2018; Cacheda et al., 2018). Some of these studies make use of the temporal aspect plus affective features in identifying early signs of depression. For example, Chen et al. (2018) attempted to identify early signs of depression of Twitter users by incorporating a progression of emotion features over time, whereas Schwartz et al. (2014) examined changes in degree of depression via Facebook users by taking advantage of sentiment and emotion lexicons. In addition, Aragón et al. (2019) introduced a method called “Bag of Sub-Emotions (BoSE)” aiming at representing social media texts by using both an emotion lexical resource and sub-word embeddings. The choice of posted images and users’ emotion, demographics and personality traits are also shown to be strong indicators of both depression and anxiety (Guntuku et al., 2019). The above mentioned studies highlight the important role of both emotion features and the temporal aspect in early detection of depression on social media. Due to the increased interest in this area, the CLEF e-risk lab has run a sub-task of measuring the severity of depression since 2018.

Some of the participant teams in this shared task present different approaches, including those based on standard machine learning algorithms (ML), deep learning and transformer-based models. Oliveira (2020) participated in the e-risk shared-task of 2020 and proposed a model named “BioInfo”. This model used a Support Vector Machine with different types of hand-crafted features (i.e. bag of words, TF-IDF, lexicons and behavioural patterns), and it ranked the top-1 model of the competition. Martinez-Castano et al. (2020) was also one of the participant teams who utilised BERT-based

transformers, achieving competitive results to that of the BioInfo model.

Our work is motivated by research focused on ML algorithms (Oliveira, 2020) and transformer-based models (Martinez-Castano et al., 2020). Our work differs from these two studies in the following ways: i) Our method combines the two approaches instead of relying on one of them. In this respect, we use the former to learn a single representation per user while utilising the latter to train on the learned representations. ii) We use the SpanEmo encoder (Alhuzali and Ananiadou, 2021b) that is trained on a multi-label emotion dataset. iii) We do not fine-tune both the transformer-based models as well as the SpanEmo encoder on the shared-task data. In other words, we treat them as feature extraction modules.

5.2.3 Experiments

Data and Task Settings. For our participation in T3 of eRisk-2021, we combined the 2019 and 2020 sets provided by the organisers, and then randomly sampled 80% and 20% for training and validation, respectively. Both sets consist of Reddit data posted by users who have answered the Beck’s Depression Inventory (BDI) questionnaire (Beck et al., 1961). The questionnaire contains 21 questions, each of which has 4 possible answers except for question #15 and #17, which have 7 possible answers. This questionnaire aims to assess the presence of feelings like sadness, pessimism, loss of energy, self-dislike, etc. Figure 5.6 presents an illustration of two questions with their possible answers¹¹.

Q1. Sadness	Q2. Pessimism
A1. I do not feel sad.	A1. I am not discouraged about my future.
A2. I feel sad much of the time.	A2. I feel more discouraged about my future than I used to be.
A3. I am sad all the time.	A3. I do not expect things to work out for me.
A4. I am so sad or unhappy that I can't stand it.	A4. I feel my future is hopeless and will only get worse.

Figure 5.6: Illustration of two questions from the questionnaire with their answers.

¹¹The rest of questions and their possible answers can be found in <https://erisk.irlab.org/2021/index.html>.

To pre-process the data, we adopt the following steps: 1) We remove empty, duplicate and broken posts (i.e., those that either break the Reddit rule or are removed). 2) We tokenise the text, convert words to lowercase, normalise URLs and repeated-characters. Table 5.6 presents the summary of all three sets, including the number of subjects/posts in the train, valid and test sets. The number of depression categories across the three sets is also included.

	Train	Valid	TEST
#subjects	72	18	80
#posts	35,537	6,207	30,382
avg #posts/subject	493	344	379
#minimal subjects	11	3	6
#mild subjects	21	6	13
#moderate subjects	18	4	27
#severe subjects	22	5	34

Table 5.6: Data Statistics.

Evaluation Metrics. For evaluating the results of our submission, we used four metrics, which measure different properties (e.g. distance between correct and predicted answers). The four metrics are¹²:

- Average Hit Rate (AHR): computes the percentage of predicted answers that are the same as the ground-truth responses.
- Average Closeness Rate (ACR): computes the absolute difference between predicted answers and the correct ones. In other words, the CR measure evaluates the model’s ability to answer each question independently.
- Average Difference between Overall Depression Levels (ADODL): computes the absolute difference between overall depression level score (i.e., sum of all the answers) and the real score.
- Depression Category Hit Rate (DCHR): evaluates the results among four categories which are based on the sum of all answers of the 21 questions. The four categories are minimal, mild, moderate and severe depression. DCHR computes the fraction of cases in which the produced category is equivalent to that of the real questionnaire.

¹²More details about the evaluation metrics can be found in [Losada et al. \(2019\)](#)

Finally, the results of each metric are computed for each user, and are then averaged over all users in the data set.

5.2.3.1 Proposed Approach

We developed a host suite of models based on neural networks for the task of predicting the severity of depression, and experimented with both feature-based transfer learning and two-stage fine-tuning. More specifically, we used feature-based transfer learning to extract a feature vector for each user, whereas we utilised two-stage fine-tuning to train a model directly on the depression data. Through extensive experiments, we observed that using feature-based transfer learning for feature extraction achieved the best results on the validation set; it was selected for conducting our experiments, as described below. Although we focused our experiments on feature-based transfer learning, we reported on the negative results of two-stage fine-tuning in Section 5.2.3.3.

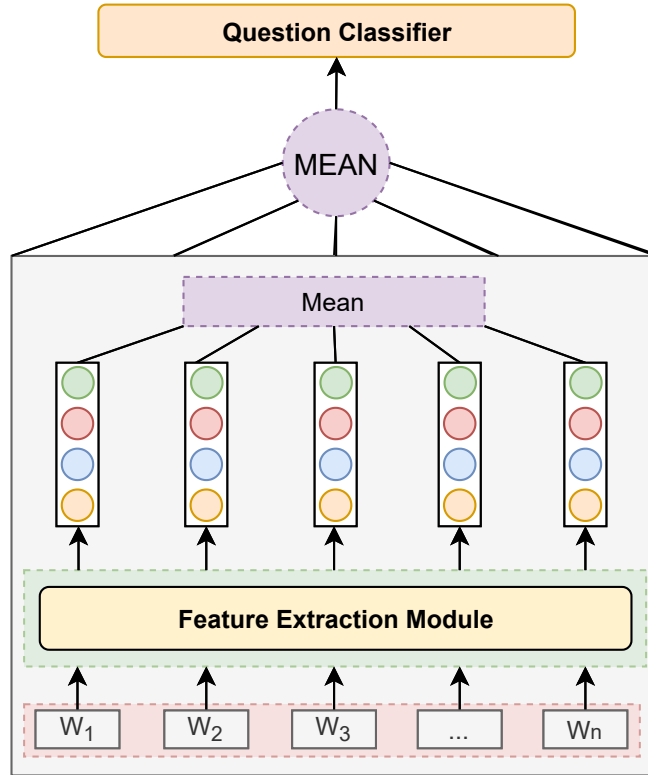


Figure 5.7: Illustration of our framework.

Let $\{p_i\}_{i=1}^N$ be a set of N posts, where each p_i consists of a sequence of M words $= (w_1, w_2, \dots, w_M)$. Figure 5.7 presents an illustration of our framework, which takes a sequence of words. In this work, we experiment with two settings: 1) processing

user posts as chunks and 2) processing individual posts independently. The first setting treats all user posts as a single document, whereas the second setting treats each user post on its own. We now turn to describing each setting in more detail:

- 1) The first setting treats all user posts as chunks of a maximum of 512 tokens. Each chunk is fed into a feature extraction module (f^{13}) and it is responsible for computing the hidden representation for each user (u) as in Equation (5.6):

$$\mathbf{u} = \frac{1}{M} \sum_{j=1}^M \mathbf{f}(w_j), \quad \mathbf{f}(w_j) \in R^d \quad (5.6)$$

where the above equation computes the mean over all tokens, with “ d ” denotes the dimensional size. This process attempts to obtain a single vector for each user that is ultimately fed to the classifier. Finally, each separate classifier is trained to predict one of the possible answers for each question.

- 2) In comparison to the first setting, we process user posts based on the following assumptions: i) we anticipate that we should be able to predict users who present an early sign of depression by using a small set of their posts instead of using all of their posts. It has been shown that depressed users tend to use similar patterns in their language use and online activity (Guntuku et al., 2017). ii) As Table 5.6 shows, the average number of posts per user consists of more than 300 posts across the three sets, i.e., training, validation and test. This means that there are posts that may not be useful at all and may even mislead our model in selecting its correct prediction. Thus, we decide to limit the number of posts that the model has access to by using a threshold value τ , which determines the number of posts it can use to make its prediction. To achieve this, we modify how each user posts are processed and then fed into the feature extraction module. Instead of concatenating all user posts, we process each post individually to extract features for each word in the post. Then, the extracted features per word are aggregated via the mean operation, which leads to a single vector representing the whole post. This process is performed on all the chosen user posts via the threshold value. After that, another mean operation is utilised to transform all post representations into a single vector representing a user. Equation (5.7) and (5.8) demonstrate how the feature extraction computes the hidden representation

¹³We will describe the feature extraction modules used in this work below.

for each post and user, respectively:

$$\mathbf{p}_i = \frac{1}{M} \sum_{j=1}^M \mathbf{f}(w_j), \quad \mathbf{f}(w_j) \in R^d \quad (5.7)$$

$$\mathbf{u} = \frac{1}{N_\tau} \sum_{i=1}^{N_\tau} \mathbf{p}_i, \quad \mathbf{p}_i \in R^d \quad (5.8)$$

where the first equation computes the mean over all tokens in the post while the second one computes the mean over all posts. \mathbf{f} represents the feature extraction module, M represents the number of words, N represents the number of posts, u represents a user, τ represents the threshold value and d represents the dimensional size.

Classification. Once we use the feature extraction module to obtain a single vector per user, we pass this vector into a specific question classifier. We experiment with Support Vector Machine (SVM) and Random forest (RF). We also run SVM using two optimisers, i.e., gradient descent (GD) and stochastic gradient descent (SGD).

Implementation Details. We used both PyTorch (Paszke et al., 2017) and scikit-learn (Pedregosa et al., 2011) for implementation and ran all experiments on an Nvidia GeForce GTX 1080 with 11 GB memory. We ran our experiments using the following metrics for evaluation, i.e., AHR, ACR, ADODL and DCHR. We selected three feature extraction modules, two of which were trained on a general domain (i.e., ELMo (Peters et al., 2018b) and BERT (Devlin et al., 2019)), whereas the third one (i.e., SpanEmo (Alhuzali and Ananiadou, 2021b)) was trained on an emotion corpus. We briefly describe each of these models below:

- ELMo¹⁴ is trained on a dataset of Wikipedia, which we use as our extraction module. More specifically, we extract the weighted sum of the 3 layers (word embedding, Bi-lstm1, and Bi-lstm2).
- BERT¹⁵ is trained on the BooksCorpus and Wikipedia. It includes a special classification token ($[CLS]$), which can be used as the aggregate input representation. The output of the ($[CLS]$) token is used for feature extraction.
- SpanEmo¹⁶ is trained on the SemEval-2018 multi-label emotion classification

¹⁴<https://github.com/allenai/bilm-tf>

¹⁵<https://huggingface.co/transformers/index.html>

¹⁶<https://github.com/hasanhuz/SpanEmo>

data set (Mohammad et al., 2018). It focuses on both learning emotion-specific features/associations and integrating the correlations between emotions into the loss function. We hypothesise that using a feature extraction model trained on a related domain to the problem under investigation can further boost the model performance compared to those models trained on a general domain.

5.2.3.2 Results and Analysis

Table 5.7 presents the results of our extended experiment and compares it to previously reported approaches on the same task. The first set of results describes two baselines, which are generated based on trivial rules like assigning all predictions as either zero (0) or one (1). The results show that the second baseline achieves quite strong performance, especially for the metrics taken distance between answers into account as ACR and ADODL. In this respect, these two baselines set the lower-bound for this shared task. The second set of results describes the teams that achieve the best scores, i.e., DUTH (Spartalis et al., 2021), CYUT (Wu and Qiu, 2021) and UPV (Basile et al., 2021). We briefly explain each team's approach below:

- The DUTH team implemented three different methods, including feature-based transfer learning, feature-based transfer learning with training machine learning classifiers and transfer learning with fine-tuning. The first method utilised “Sentence-BERT” to obtain the vector representation of each user posts and then aggregated the extracted representations via a mean operation. This process was also performed for the responses of each question in order to apply the cosine similarity between the representations of each user posts and the responses. Lastly, the response with the highest value was chosen. The second method is similar to the first one, but trained a classifier on top of the extracted representations. The first and second methods achieved strong performance compared to the third method. This is also in line with what we have observed in our experiments.
- The CYUT team addressed the task through the use of the RoBERTa pre-trained model for each BDI question by concatenating the last four layers. Since each user has plenty of posts, the authors experimented with three different aggregation methods, where the first one is simply a majority vote, the second one assigns fixed weight to each response, and the third one is based on a threshold. They found that the second aggregation method achieved the best results.

Model	AHR	ACR	ADODL	DCHR
Baselines				
All 0s Baseline	23.03%	54.92%	54.92%	7.50%
All 1s Baseline	32.02%	<u>72.90%</u>	81.63%	33.75%
Prior Work				
DUTH (Spartalis et al., 2021)	<u>35.36%</u>	67.18%	73.97%	15.00%
UPV (Basile et al., 2021)	34.17%	73.17%	82.42%	32.50%
CYUT (Wu and Qiu, 2021)	32.62%	69.46%	83.59%	<u>41.25%</u>
This Work				
1 st setting				
RF (Elmo)	31.43%	64.54%	74.98%	18.75%
RF (Bert)	31.55%	65.00%	75.04%	21.25%
RF (SpanEmo-Encoder)	32.86%	66.67%	76.23%	22.50%
2 nd setting				
RF ($\tau = 40$)	34.46%	64.74%	77.00%	25.00%
SVM w/o SGD ($\tau = 30$)	35.65%	65.26%	76.88%	27.50%
SVM w/ SGD ($\tau = 20$)	31.96%	63.35%	<u>83.23%</u>	48.75%

Table 5.7: Experimental results on the test set. τ : refers to the number of posts used by the respective model during the training phase. The best results are marked in bold, whereas the second best results are underlined.

- The UPV team used a temporal approach based on features from the NRC lexicon and three categories of LWIC. For each post, the number of words corresponding to the chosen features are counted and then normalised by the post length. For the temporal aspect, the authors group posts that were posted within the same day, and they then employed two metrics to compare users from the test set to those in the training set. Although a classifier based on RoBERTa was used, it did not perform well compared to the temporal approach.

Next, we discuss the results of our two settings. Table 5.7 presents the results of the first setting on all four metrics (i.e., AHR, ACR, ADODL and DCHR). We report our results on all three feature extraction modules (i.e., ELMo, BERT and SpanEmo-encoder). The third feature extraction module based on SpanEmo-encoder achieved the best results, thus demonstrating the utility and advantages of using a trained model on a related task to the one under investigation. This confirms our initial observation and helps to reinforce that our model can benefit from the similarity between the two tasks in detecting signs of depression, given that some of the BDI questions are also

related to emotion concepts, such as sadness, pessimism, loss of pleasure, self-dislike, etc.

We also report our results regarding the second setting using the same random forest classifier (RF). We also include the results of SVM with two optimisers, i.e., GD and SGD. As shown in Table 5.7, the results of SVM outperforms all models on both AHR and DCHR metrics, while achieving competitive performance to the “CYUT” on the ADODL metric, with a marginal difference equivalent to 0.36. The results of SVM trained with SGD produced the best results for ADODL and DCHR metrics. Although our model is trained on a small number of posts, it performs strongly or even better compared to prior approaches, which make use of all posts. This makes our approach distinctive in the sense that it is able to achieve competitive performance while utilising a small number of posts for each user.

Evaluating the Results of Different Layers. We evaluated different layers in the SpanEmo-Encoder to determine which one is best for obtaining the highest results. The evaluation is presented in Figure 5.8, which reveals that different layers achieve different scores depending on the chosen metric, especially for AHR and DCHR. Based on this evaluation, we selected the results of the ninth layer for our submission as it demonstrates strong performance on almost all four metrics.

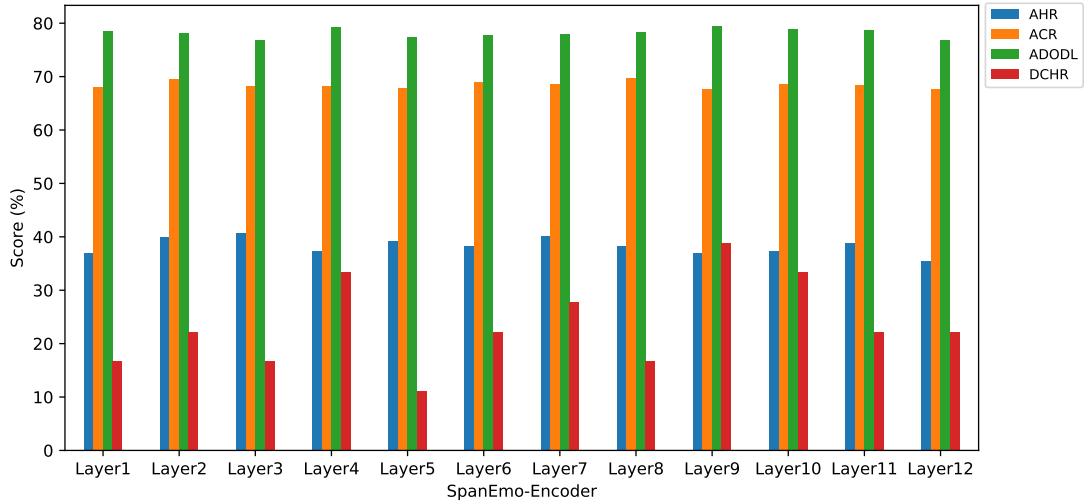


Figure 5.8: The results of each SpanEmo-Encoder layer when applied to the validation data set of the depression task.

Evaluating the impact of #posts (AHR vs ACR). Based on our previous evaluation, we observed that classifiers achieved different performance depending on the number of posts used to train them. Figure 5.9 presents the analysis of those classifiers

when the number of posts are varied from 10 to 100. The top figure corresponds to the results of AHR, whereas the bottom figure corresponds to the results of ACR. We can clearly observe the following: i) Regarding the AHR metric, as the number of posts increases, the performance of classifiers generally improves. Once the number of posts reaches 40, the performance starts to drop apart from the performance of the SVM classifier trained with *SGD*. Since the SVM classifier trained without *SGD* obtains the best performance when the number of posts equals 30, we choose this value for the AHR metric. ii) Regarding the ACR metric, we see similar patterns that the performance improves as the number of posts increase, but once the number of posts reaches 30 or 40, the performance begins to drop again. These results demonstrate that we can predict the level of depression to some extent while using less data, which is sensible since the selected posts carry information about the user's behaviour and emotions.

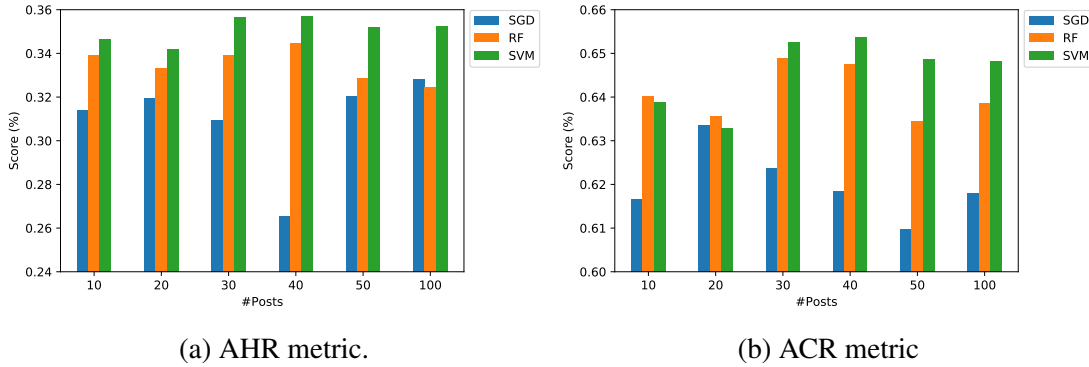


Figure 5.9: The results of varying the number of posts.

Evaluating the impact of posts (ADODL vs DCHR). Figure 5.10 presents the results of both ADODL and DCHR. For these two metrics, we observe that the best performance is achieved by the SVM classifier trained with *SGD*, with a high marginal improvement compared to the other two classifiers. For the ADODL metric, we notice that increasing the number of posts did not contribute much to the performance, but using only 10 posts produced the best result. However, the performance for the DCHR improves as the number of posts increases. After the number of posts reaches 40, the performance deteriorates dramatically. Again, the results of these two metrics help to enforce that, by using a small number of posts, our approach demonstrates strong performance.

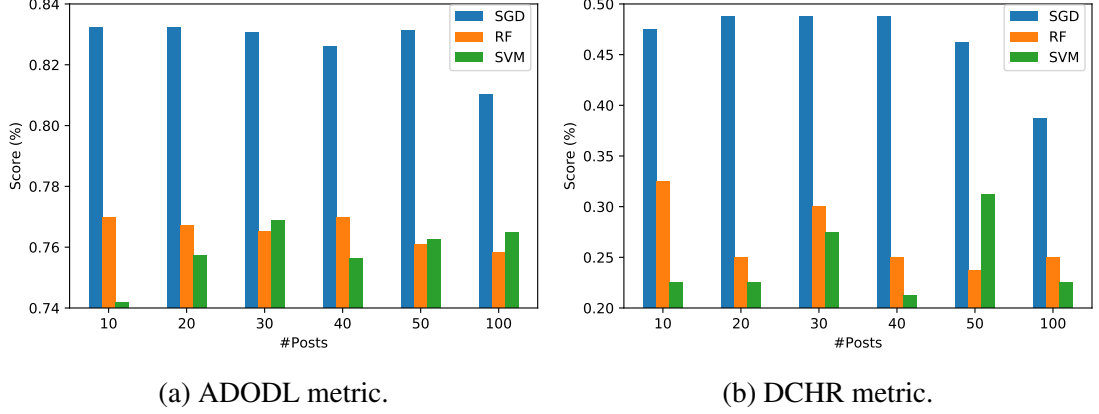


Figure 5.10: The results of varying the number of posts.

5.2.3.3 Negative Results

We experimented with multi-task learning (MTL), for which we trained a single model for all 21 questions. More specifically, a shared BERT-based encoder was utilised to obtain a hidden representation for each post, and a specific head was then used for each question. To achieve a single output for each user, we aggregated the produced outputs from all posts via either averaging or summing. We also employed dynamic weighting of question-specific losses during the training process (Kendall et al., 2018) as follows:

$$\mathcal{L}_{MTL} = \sum_q^{21} \frac{1}{2\sigma_q^2} \mathcal{L}_q + \log \sigma_q^2 \quad (5.9)$$

where q denotes a question and both \mathcal{L}_q and σ_q represent the question-specific loss and its variance. After computing all the 21 losses, we average them. However, the results of MTL were not as high as the one discussed in Section 5.2.3.2. This may be attributed to a number of factors. Firstly, we used only a simple aggregate function that did not take the temporal aspect into consideration. This could be useful for detecting early sign of depression in users posts¹⁷. Secondly, there were no annotations provided at the post-level which could help identify posts expressing severity signs of depression from those that do not. Thirdly, we observed that the MTL model overfits with respect to the training data after the third or fourth epoch although we used a dropout to regularise that. This may be because the size of the data is quite small (i.e., roughly 70 users to train on), making the model unable to learn effectively from them.

¹⁷Due to resource constraints, we could not train our model on the user's timeline in a sequential minor.

5.3 Summary

In this chapter, we aimed to investigate the benefits of emotional knowledge on two case studies, i.e., adverse drug reaction and an application of mental health-depression. We began the description of each case study with some introductory background, including our motivation and contribution. We then discussed some related work, corpora and experiments relevant to each case study.

In the first case study, we demonstrated our novel neural network architecture applied for ADR classification. Our approach exploits the fact that in social media, ADRs are frequently expressed with sentimental and emotional expressions. Taking advantage of the readily available sentiment and emotion datasets that are available online, our architecture firstly trained a sentiment/emotion classifier on tweets, and then adapted the trained classifier to detect ADRs in social media. Our empirical results have demonstrated that the application of the fine-tuned model to ADR datasets obtained a substantial improvement over previously published models. Additionally, the word coverage analyses revealed that sentiment data share a significant amount of vocabulary with ADR data, which is even higher than the correlation between the words in training and test sets of the same ADR dataset. We discussed the advantages and utility of both sentiment/emotion corpora and transfer learning techniques for improving the performance of ADR detection in social media and specialised health-related forums, and provided some error analyses and potential future work.

In the second case study, we proposed a framework aimed at detecting signs of depression from users posts. Our approach benefits from the architecture of SpanEmo model in extracting features for users posts without even training it on the depression data. In this work, we experimented with two settings, and found that the second setting achieved better performance, especially for the AHR and DCHR metrics. Our evaluation further showed that different SpanEmo-encoder layers produced different results. The choice of which layer to choose depends on the metric of interest. Moreover, we analysed the results of varying the number of posts that the model used for both training and prediction. We found that our model outperformed prior approaches while utilising a small set of user posts. Finally, we reported some negative experiments, and hope that it will inspire the community to investigate further the vital role of learning a single model for all the 21 questions. This is motivated by the fact that some questions have some correlations/associations, and inferring the answer for one question may help infer others as well.

For both case studies, we observe that emotional knowledge can help improve the

performance of downstream applications, especially when the task under investigation shares common patterns with the task of emotion. Our SpanEmo model demonstrated that it can be easily adapted to other tasks and can capture generalisable emotion features. For example, we used SpanEmo to extract features for user posts without even training it on the depression data. This means that SpanEmo learned meaningful representations, which can benefit downstream applications. Our analysis in [Section 5.1.5.2](#) evaluated the generalisability of SpanEmo model by training it on one of the ADR corpora and testing it on the other one. This analysis revealed the utility and advantages of SpanEmo model in terms of improving the performance of ADR model even if tested on a different dataset from the one it was trained on, especially when the task under investigation contains a small size of data.

Chapter 6

Incorporating Intra- and Inter-Class Variations into Textual Emotion Recognition

In Chapter 4, we proposed SpanEmo for multi-label emotion classification to disentangle positive emotions (i.e., highly correlated emotions) from negative emotions. Our SpanEmo model takes advantage of label co-occurrences in a multi-label emotion corpus. This requirement is not available in single-label emotion classification, on which most of the existing emotion corpora are built. In this chapter¹, we propose a novel objective for single-label emotion classification aimed at disentangling highly confused emotions (i.e., positive emotions from negative ones). This chapter specifically addresses our third research question (**RQ#3**). To disentangle highly confused emotions, we introduce a variant of triplet centre loss as an auxiliary task to emotion classification.

Prior research has tackled the automatic classification of emotion expressions in text by maximising the probability of the correct emotion class using cross-entropy loss, which does not account for intra- and inter-class variations within and between emotion classes. In contrast, our approach to Textual Emotion Recognition (TER) accounts for both intra- and inter-class variations in TER. It builds upon the work of He et al. (2018), who leveraged both intra- and inter-class variations for object recognition. Our work differs from (He et al., 2018) in the following ways: i) We leverage

¹This chapter is drawn from Alhuzali, H., & Ananiadou, S. (2021). Improving Textual Emotion Recognition Based on Intra- and Inter-Class Variation. IEEE Transactions on Affective Computing (In Press). IEEE © 2021. Reprinted, with permission.

intra- and inter-class information to recognise emotion expressions in text, rather than objects. To the best of our knowledge, this is the first attempt to apply this approach, in conjunction with triplet centre loss, to text. ii) We employ an alternative method to compute inter-class distance, so as to disentangle the positive emotion label (i.e., ground truth) from negative ones. iii) We empirically quantify the influence of intra- and inter-class variations directly for each emotion class by introducing an evaluation method. Finally, we present analyses that illustrate the benefits of our method in terms of improving the prediction scores as well as producing discriminative features.

6.1 Motivation

The growing interest in TER has been motivated by the proliferation of social media and online data, which have made it possible for people to communicate and share opinions on a variety of topics. Interest in TER has also given rise to new Natural Language Processing (NLP) methods focusing on TER identification and classification (Akhtar et al., 2019; Klinger et al., 2018b; Mohammad and Turney, 2013b; Mohammad, 2012a; Tang et al., 2013; Wang et al., 2012; Strapparava and Mihalcea, 2008; Aman and Szpakowicz, 2007). Research into TER has contributed to a wide range of real-world applications, e.g., health and well-being (Aragón et al., 2019; Chen et al., 2018; Khanpour and Caragea, 2018), author profiling (Volkova and Bachrach, 2016; Mohammad and Kiritchenko, 2013), human-machine interaction (Rashkin et al., 2018; Fung, 2015; Picard, 2000), education (Voigt et al., 2017; Suero Montero and Suhonen, 2014), financial technology (Li and Shah, 2017; Mansar et al., 2017; Liu et al., 2016) and consumer analysis (Alaluf and Illouz, 2019; Herzig et al., 2016).

#	Sentence	GT
S1	I love you so much and i am [trigger_word] because you do not know that i exist.	sadness
S2	I get so [trigger_word] when parents smoke right next to their little kids.	disgust

Table 6.1: Example Tweets from IEST dataset (Klinger et al., 2018b). GT represents the ground-truth labels (© 2021 IEEE).

The majority of previous research has focused on emotion classification as a single-label prediction problem by selecting the most dominant class for a given emotion expression. This approach makes use of cross-entropy loss, which attempts to maximise the probability of the correct class. However, it does not account for cases in which

certain emotions (e.g., anger, disgust or sadness) may be confused with each other. Consider S1 in Table 6.1, which contains a strong expression of “joy”, even though it is generally more negative oriented. This can lead TER models to choose the “joy” over “sadness” emotion. S2 is annotated with “disgust”, while at the same time it could be possibly labelled with “anger”, due to the missing of explicit emotion-based keywords for the “disgust” emotion, as well as their similarities in linguistic expressions. This linguistic overlap between different emotion classes can cause TER models to mislabel emotions and affect their performance in selecting the correct label. Based on these observations, we hypothesise that taking into account variations both within and between different classes of emotion can better support TER models in learning discriminative features and improve their prediction capability. In this chapter, we refer to examples sharing the same emotion class as “intra-class”, while examples belonging to different emotion classes are referred to as “inter-class”. Our contributions, which are discussed in Chapter 1, are further stated below:

- a novel loss function aimed at incorporating intra- and inter-class variations into TER. More specifically, we introduce a variant of triplet centre loss (VTCL) as an auxiliary task to emotion classification loss (i.e., cross-entropy loss). The objective of VTCL is to minimise the distance of the examples from the centre within the same emotion class (intra-class), while maximising their distances from the centres of other emotions classes (inter-class).
- a new evaluation method to quantify the impact of intra- and inter-class variations on each emotion class.
- an in-depth analysis that reveals the benefits of taking intra- and inter-class variation into account, which can improve model performance compared to previous approaches, even without the use of external resources. Empirical evaluations and analysis demonstrate that both intra- and inter-class variations can help the model to achieve high prediction scores, and to be a better discriminator against highly confused emotions.

The rest of this chapter is organised as follows: Section 6.2 provides an overview of triplet centre loss and describes how our method improves upon it. Section 6.3 discusses experimental details. We evaluate our method and compare it to related methods in Section 6.4. We report on the analysis of results in Section 6.5, while we conclude in Section 6.6.

6.2 Methodology

6.2.1 Triplet Centre Loss

Triplet centre loss (TCL) is a combination of triplet loss (TL) (Schroff et al., 2015) and centre loss (CL) (Wen et al., 2016). TL defines a triplet as an anchor sample, a positive sample and a negative sample; the first two samples belong to the same class, while the last one belongs to a different class. The objective of TL is to minimise the distance between an anchor sample and a positive sample, while increasing the distance to a negative sample by at least a margin m . However, the number of triplets can grow cubically as the number of samples increases, which requires a long training period. In addition, the performance of TL is highly dependent on the choice of triplet mining technique, which is also computationally expensive. The above-mentioned reasons make TL models hard to train.

An alternative choice to TL is CL, which learns the centre for the samples of each class, with the objective of pulling them as close as possible to their respective centre. Although CL is easier to implement, it runs the risk of degrading all features and centres to zero (Wen et al., 2016). To address this problem, CL is trained in conjunction with cross-entropy loss, since the latter can act as a guide to learn better centres. Nevertheless, CL does not guarantee that the centres of different classes are pushed sufficiently far from each other. This is because CL only focuses on minimising intra-class distance, but it does not directly address the issue of maximising inter-class distance.

In response to the above, He et al. (2018) proposed TCL, which follows the same method as TL, while simultaneously avoiding its complexity. TCL only requires access to a sample (i.e., its corresponding centre and its nearest negative centre). In this respect, TCL leverages the benefits of both TL and CL, in that it pulls samples as close as possible to their corresponding centre, while pushing the same samples as far away as possible from their nearest negative centre.

6.2.2 Variant Triplet Centre Loss

Our proposed method is an enhancement of He et al. (2018) triplet centre loss, which we call VTCL. In VTCL, we assume that the features of emotion expressions from one class could be shared by expressions from other emotion classes. This makes our approach distinct from TCL for two reasons. Firstly, since TCL only considers the nearest negative centre, the difference between intra- and inter-class distances for

multiple (possibly very similar) emotion classes cannot be established. Secondly, TCL randomly initialises the parametric centres, making the process of selecting the nearest negative centre hard to achieve. This is particularly problematic for a task like TER, in which multiple classes could be used as negative centres, due to the close association between certain emotion classes (e.g., anger, disgust and sadness).

To address the above challenges, we map each emotion class to one corresponding centre and treat all but the one positive class centre as negative centres. This simplifies our method by obviating the need to determine the closest negative centre. In other words, examples belonging to the same class should be as close as possible to each other (intra-class), while the same examples should be as far away as possible from other emotion classes (inter-class). This ensures that the intra-class distance plus the margin are always smaller than the inter-class distance. Our experiments in Section 6.5.4 show the impact of choosing different numbers of negative centres. We compute VTCL as follows:

$$\mathcal{L}_{\text{VTCL}} = \max(\text{intra} + m - \text{inter}, 0), \quad (6.1)$$

where intra- and inter-class distances are computed by using the Squared Euclidean Distance as shown in Equation (6.2) and Equation (6.3), respectively. m is a marginal difference between the intra- and inter-class distances.

$$\text{intra} = \frac{1}{2} \sum_{i=1}^B \|f_i - c_{y^i}\|_2^2, \quad (6.2)$$

$$\text{inter} = \frac{1}{2} \sum_{i=1}^B \sum_{j \neq y^i}^C \|f_i - c_j\|_2^2, \quad (6.3)$$

where B is the training batch size, C corresponds to the number of emotion classes, $f_i \in \mathbb{R}^d$ is the i^{th} input representation, $c_{y^i} \in \mathbb{R}^d$ is the centre of class y^i and $c_j \in \mathbb{R}^d$ is the centre of other emotions, with d defining the dimensional size.

6.2.3 Training Objective

As VTCL initialises the parametric centres randomly and updates them based on the mini-batches, it is difficult to achieve accurate class centres. To mitigate this problem, we train VTCL jointly with the Cross-Entropy Loss function (CEL). VTCL applies metric learning to the learned feature representation directly, while CEL focuses on

mapping examples to their emotion classes, helping to achieve discriminative as well as compact features, respectively. The overall training objective can be defined as follows:

$$\mathcal{L}_{\text{JOINT}} = \mathcal{L}_{\text{CEL}} + \lambda \mathcal{L}_{\text{VTCL}}, \quad (6.4)$$

where the first term refers to the CEL, which is computed as in shown equation (6.5), while the second term corresponds to VTCL. $\lambda \in [0, 1]$ denotes the value used to control the trade-off between L_{CEL} and L_{VTCL} .

$$\mathcal{L}_{\text{CEL}} = - \sum_{i=1}^B \sum_{j=1}^C \mathbb{1}\{y_i = j\} \log \frac{e^{a_j^{(i)}}}{\sum_{j=1}^C e^{a_j^{(i)}}}, \quad (6.5)$$

where the indicator function $\mathbb{1}\{condition\} = 1$ if the *condition* is satisfied, or 0 otherwise. $a_j^{(i)}$ represents the activation values of emotion classes in the last fully-connected layer for an example.

6.3 Experiments

In this work, we run our method on two widely used networks for TER: the first network is based on a CNN architecture (Kim, 2014), while the other network is based on BERT (Devlin et al., 2019)². Figure 6.1 illustrates the proposed method, which takes advantage of the same feature representation obtained via either BERT or CNN.

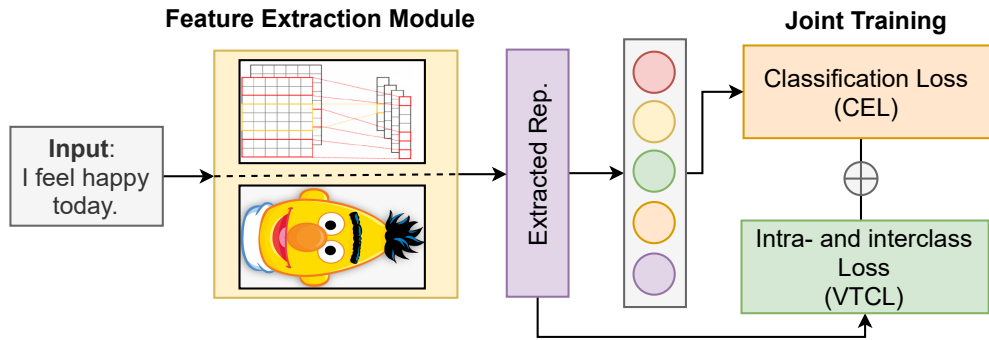


Figure 6.1: Illustration of our method. Given the input, we use a feature extraction module based on BERT/CNN to learn the input representation and then feed it into our method, which includes the joint supervision of CEL and VTCL (© 2021 IEEE).

²We train BERT on the default hyper-parameters using the open-source Hugging Face implementation (Wolf et al., 2019).

6.3.1 Implementation Details

The CNN network’s weights were initialised from *Word2Vec* (Mikolov et al., 2013a) embedding with a size of 300 dimension, and it included filter windows of (3, 4, 5) with 100 feature maps each, a batch size of 64 and a dropout rate of 0.5. We used the standard normal distribution to initialise the centres and we set the margin (m) double the number of negative centres³. Adaptive Moment Estimation Algorithm (ADAM) was selected for optimisation (Kingma and Ba, 2014) with a learning rate of 1e-3 for the network, as well as for the centres. All experiments were performed with a fixed initialisation seed using PyTorch (Paszke et al., 2017) and an Nvidia GeForce GTX 1080 with 11 GB memory. Table 6.2 summarises the hyper-parameters used in this work, including those related to BERT.

Parameters	CNN	BERT
Window sizes	{3, 4, 5}	-
Feature maps	100	-
Feature dimension	300	768
Batch size	64	32
Dropout	0.5	0.1
Learning rate	1e-3	2e-5
Margin (m)	$2 \times NC $	
Optimiser	Adam	
Early stop patience	10	

Table 6.2: Hyper-parameters. $|NC|$: denotes the number of negative centres (© 2021 IEEE).

6.3.2 Datasets and Task Settings

We evaluated our method on three widely used single-label datasets (i.e., IEST, ISEAR and TEC) and conducted our experiments in a stratified 10-fold cross-validation setup, ensuring that all folds contain an approximately equal sample of emotion classes. In this chapter, we focus on Ekman’s (Ekman, 1992) 6 basic emotions $\{anger, disgust, fear, joy, sadness, and surprise\}$ because two of the datasets (i.e., IEST and TEC) we used are annotated with those 6 emotions. Table 6.3 provides a summary of each dataset, including the domain (i.e. the source from which the dataset is collected), the size, the number of words and the average length of sentences/tweets for each dataset.

³The m parameter is set by observing the F1-score curve on validation set.

For pre-processing the data, we utilise the “ekphrasis⁴” tool (Baziotis et al., 2017) designed for the specific characteristics of Twitter, e.g., misspellings and abbreviations since two of the datasets we used are collected from Twitter. The tool offers different functionalities, such as tokenisation, normalisation, spelling correction, and segmentation. For all three datasets, we used “ekphrasis” to tokenise the text, convert words to lower case, and normalise user mentions, URLs and repeated-characters.

Dataset	IEST	ISEAR	TEC
Domain	Tweets	Events	Tweets
# Sentences	30k	7k	21k
# Words	24,803	8,293	21,853
# Avg.length	23.47	25.5	17.56
# Sentences/Tweets per class			
Anger	5,000	1,096	1,555
Disgust	5,000	1,096	761
Fear	5,000	1,095	2,816
Joy	5,000	1,094	8,240
Sadness	5,000	3,285	3,830
Surprise	5,000	—	3,849

Table 6.3: Statistics of datasets. *Avg.length* refers to the average length of sentences/tweets (© 2021 IEEE).

6.4 Evaluation

6.4.1 Results

Table 6.4 presents the performance of VTCL on each dataset, in terms of precision, recall and F1-score, and compares it to previously reported state-of-the-art approaches to TER, contextualised embedding and strong loss functions. The results reported in Table 6.4 are an average of stratified 10-fold cross-validations. In the sections below, we briefly describe the methods that we have compared, including methods that learn a joint loss function to improve the results of emotion classification and those that only use CEL.

⁴<https://github.com/cbaziotis/ekphrasis>

6.4.1.1 Relevant Work

[Klinger et al. \(2018c\)](#) used a Maximum Entropy classifier (MaxEnt) with a bag of words features for detecting emotion expressions in text. This model exhibits the lowest performance among all compared approaches, as it was trained only on simple features. [Islam et al. \(2019b\)](#) built a Multi-Channel-CNN (MCC), which attempts to learn embeddings for each input instance and additional features that occur in the same input instance (e.g., emojis, emoticons and hashtags). This model achieves better results than the MaxEnt classifier, and it also obtains the highest recall apart from BERT and its variations over all models on the TEC dataset. The fact that MCC used additional hand-crafted features (e.g. tweet-specific, affect and sentiment features) may explain its high recall on this dataset. We only report on the results of MCC on the IEST and TEC datasets because it was specifically developed for Twitter. [Zhang et al. \(2018\)](#) proposes a Multi-Task-Loss approach (MTL) involving learning of both emotion distribution and classification (i.e., CEL plus Kullback-Leibler (KL) loss). The MTL model achieves higher recall and f1 scores than “CNN (ours)” on the ISEAR dataset. However, it should be noted that, in contrast to our approach, it relies on emotion lexicons to generate label distributions.

Finally, we compare strong variants of loss functions aimed at learning intra- and inter-class variations, i.e., including CEL+CL ([Tripathi et al., 2020](#)) and CEL+TCL ([He et al., 2018](#)). Based on experimental results, we observe that including intra- and inter-class information improves the model performance; CL achieves higher results than TCL on almost all metrics and datasets, proving our earlier hypothesis in Section 6.2.2 that determining the nearest negative centre is indeed not possible for TER. The same patterns are also observed in BERT experiments, which are discussed below.

6.4.1.2 Contextualised Embeddings

We also compared and applied our method to BERT for two reasons: i) it can serve as a strong baseline and ii) it can demonstrate the usefulness of VTCL when tested on a different network. To create a sentence representation, we stack a softmax activation layer over the hidden state corresponding to [CLS] in BERT and only consider the “bert-base-uncased” model. As shown in Table 6.4, BERT trained with the other variants of loss functions apart from VTCL achieves higher results than previously reported approaches to TER on all three datasets. Nevertheless, “CNN (ours)” outperforms the results of BERT trained with the other loss functions apart from VTCL on

Dataset	IEST			ISEAR			TEC		
Model/Metric	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
Relevant Work									
MaxEnt	49.41	48.90	48.85	61.20	63.60	62.20	49.83	48.50	49.00
CNN (CEL)	55.74	55.20	55.22	64.21	63.97	63.66	57.79	51.43	53.55
MCC (CEL) [†]	56.20	56.95	56.09	—	—	—	55.90	56.50	55.60
MTL (CEL+KL) [‡]	57.17	56.93	56.95	67.11	66.91	66.80	62.10	52.57	56.94
CNN (CEL+CL)	57.20	56.62	56.63	65.80	64.12	64.27	64.47	53.96	56.96
CNN (CEL+TCL)	57.27	56.56	56.60	65.30	64.03	63.59	63.27	54.18	56.90
CNN (ours)	58.08	57.77	57.71	70.35	64.14	65.79	65.85	54.95	58.19
Contextualised Embedding									
BERT (CEL)	56.84	56.23	56.27	68.60	67.50	66.92	58.71	58.72	57.67
BERT (CEL+CL)	59.09	56.77	56.98	68.52	67.84	67.42	60.12	57.87	57.93
BERT (CEL+TCL)	57.88	56.70	56.85	68.03	67.01	66.83	59.07	56.46	57.31
LS (CEL+Corr)	57.74	57.00	57.06	67.32	67.10	67.08	59.31	56.51	57.08
BERT (ours)	60.20	59.34	59.38	70.73	69.44	68.89	60.18	60.24	59.47

Table 6.4: Comparison of our method to previous approaches as well as contextualised embedding applied to IEST, ISEAR and TEC datasets. (P%), (R%) and (F1%): refers to precision, recall and f1-score. Note that [†] indicates that the model uses hand-crafted features (e.g. tweet-specific, affect and sentiment features), while [‡] indicates that the model utilises emotion lexicons to generate label distribution. “Corr”: refers to correlation. The best result in each part is marked in bold (© 2021 IEEE).

both IEST and TEC datasets. Although BERT obtains competitive results to “CNN (ours)”, its trained parameters are much larger than those of CNN trained jointly with VTCL.

The fact that BERT scores are higher on the ISEAR dataset than “CNN (ours)” may be because this dataset is quite similar to its pre-training corpus. To investigate this, we measured the degree of common word coverage between the “bert-base-uncased” vocabulary and the training set of each dataset. We found that the percentage of shared words between the “bert-base-uncased” vocabulary and the training set of ISEAR is 74%, while it is less than 50% for the other two datasets. This confirms our above observation that BERT is pre-trained on a corpus more similar to the ISEAR dataset than the IEST and TEC datasets.

We considered further a Label Semantic (LS) approach (Gaonkar et al., 2020a) which adopted BERT as its encoder and aimed at learning emotion classification and correlation via a joint loss function. Although the LS model used a joint loss function as well as learning the input representation from BERT, it achieved lower performance than our method. It is also worth mentioning that this model takes longer to train than our method because it casts the task as a multiple choice answering task.

6.4.1.3 Our Method (CEL+VTCL)

Table 6.4 demonstrates that “CNN (ours)” outperforms all compared models on the IEST and TEC datasets, and all models apart from MTL and BERT when applied to the ISEAR dataset. However, when BERT is trained jointly with VTCL, it achieves the highest results across the three datasets. A further observation is that CNN/BERT trained on VTCL outperforms across all the three datasets compared to CNN/BERT trained on the other loss functions (i.e., CEL, CEL+CL and CEL+TCL). This proves the strength of VTCL against these loss functions as well as against both the MTL and LS approaches. In addition, VTCL does not rely on any external resources unlike MTL, which relies on emotion lexicons to generate label distribution. Moreover, VTCL only requires a small number of parameters to be trained, equivalent to the number of emotion classes multiplied by the size of the feature dimension. Even though VTCL is tested on the simple CNN network architecture, it shows strong performance because, unlike other approaches, it benefits from taking into account intra- and inter-class variation, whose impact on model performance is assessed in the next section via an ablation study.

6.4.2 Ablation Study

We undertake an ablation study of the results using two settings: firstly, the model is trained without inter-class variations and subsequently, it is trained without intra-class variations. Training the model without these two types of information is equivalent to training it only with CEL. Table 6.5 shows the results. As Table 6.5 shows, the results

Dataset	IENT	ISEAR	TEC
Model	F1 (%)	F1 (%)	F1 (%)
CNN (ours)	57.71	65.79	58.19
-inter	56.63 (↓ 2%)	64.27 (↓ 2%)	56.96 (↓ 2%)
-intra	55.22 (↓ 4%)	63.66 (↓ 3%)	53.55 (↓ 8%)
BERT (ours)	59.38	68.89	59.47
-inter	56.98 (↓ 4%)	67.42 (↓ 2%)	57.93 (↓ 3%)
-intra	56.27 (↓ 5%)	66.92 (↓ 3%)	57.67 (↓ 3%)

Table 6.5: Ablation experiment results. The proportions in parentheses indicate the relative change with respect to ours (© 2021 IEEE).

of CNN and BERT drop by 2-4% f1-score when the inter-class is removed. When the intra-class is additionally removed, the performance drop increases to 3-8% in f1-score. These results demonstrate the benefits of incorporating intra- and inter-class

variations into TER, supporting our hypothesis that taking into account both types of information can improve the model performance substantially.

6.4.3 Intra- and inter-class evaluation

We evaluate the ability of our method to distinguish between intra- and inter-class variations with respect to each emotion. Since there is no existing metric for evaluating the impact of intra- and inter-class variations on each emotion class, we choose the confusion matrix. The confusion matrix provides a summary of the model performance per class, where correct predictions are represented in the diagonal, while incorrect predictions are shown outside the diagonal. For example, if a row represents joy, we then obtain the values of “joy-to-joy” (i.e., correctly labelled examples), “joy-to-anger” (i.e., mislabelled examples), “joy-to-disgust” (i.e., mislabelled examples), etc. We use the value of correctly labelled examples to represent the intra-class performance, while utilising the values of incorrectly labelled examples to represent the inter-class performance. The inter-class values are then summed up by following Equation (6.3). We can then quantify the impact of intra- and inter-class results with respect to each emotion class. Table 6.6 depicts an illustration of how intra- and inter-class values are computed for six emotion classes.

Emotion	Anger	Disgust	Fear	Joy	Sadness	Intra	Inter
Anger	0.52	0.12	0.09	0.05	0.22	0.52	0.48
Disgust	0.13	0.53	0.07	0.06	0.21	0.53	0.47
Fear	0.06	0.07	0.61	0.05	0.21	0.61	0.39
Joy	0.06	0.03	0.07	0.63	0.21	0.63	0.37
Sad	0.12	0.06	0.09	0.07	0.66	0.66	0.34

Table 6.6: Illustration of how intra- and inter-class scores are computed. The inter-class score is obtained by summing all the non-diagonal values in the same row.

Table 6.7 presents the results of intra- and inter-class performance per emotion class on all three datasets. We compare the performance of TCL and VTCL, because both are optimised for the objective of minimising the intra-class distance (i.e., within samples sharing the same emotion) and maximising the inter-class distance (i.e., between samples sharing dissimilar emotions). As Table 6.7 demonstrates, compared to TCL, our VTCL method achieves higher values for intra-class distance, and lower

Dataset	Loss	Mode/Label	anger	disgust	fear	joy	sadness	surprise
IEST	TCL	intra (\uparrow)	49.80	49.40	57.20	64.20	58.40	61.00
		inter (\downarrow)	50.20	50.60	42.80	35.80	41.60	39.00
		Δ (\uparrow)	-0.40	-1.20	14.40	28.40	16.80	22.00
	VTCL	intra (\uparrow)	54.20	53.00	62.60	65.60	61.80	56.60
		inter (\downarrow)	45.80	47.00	37.40	34.40	38.20	43.40
		Δ (\uparrow)	8.40	6.00	25.20	31.20	23.60	13.20
ISEAR	TCL	intra (\uparrow)	46.36	40.91	48.18	59.09	77.03	—
		inter (\downarrow)	53.64	59.09	51.82	40.91	24.22	—
		Δ (\uparrow)	-7.27	-18.18	-3.64	18.18	52.81	—
	VTCL	intra (\uparrow)	51.82	52.73	60.91	62.73	66.26	—
		inter (\downarrow)	48.18	47.27	39.09	37.27	33.74	—
		Δ (\uparrow)	3.64	5.45	21.82	25.45	32.52	—
TEC	TCL	Intra (\uparrow)	44.32	41.67	47.87	55.83	57.70	55.54
		inter (\downarrow)	55.68	58.33	52.13	44.17	42.30	44.46
		Δ (\uparrow)	-11.36	-16.66	-4.26	11.65	15.40	11.08
	VTCL	intra (\uparrow)	54.19	47.56	64.77	65.90	61.88	56.88
		inter (\downarrow)	45.81	52.44	35.23	34.10	38.12	43.12
		Δ (\uparrow)	8.39	-4.88	29.54	31.80	23.76	13.77

Table 6.7: The results (%) of intra- and inter-class values on three datasets (i.e., IEST, ISEAR and TEC). Note that \uparrow after the mode indicates the larger the better, while \downarrow after the mode indicates the smaller the better. Δ represents the difference between intra -and inter-class scores. The best scores are highlighted in bold. It should be mentioned that ISEAR is not annotated with the “surprise” emotion (© 2021 IEEE).

values for inter-class distance among almost all emotion classes apart from the disgust class in the TEC dataset. We attribute this to the small number of tweets belonging to the disgust class as shown in Table 6.3, which contains 761 tweets for both the training and test set. We observe that some emotions are easier to distinguish than others. For example, the “joy”, “fear” and “sadness” emotions achieved higher marginal difference between the intra- and inter-class than “anger” and “disgust”. This finding is consistent with the studies of [Agrawal et al. \(2018\)](#) and [Mohammad and Bravo-Marquez \(2017a\)](#), both of which report the same issue with negative emotions of “anger” and “disgust”, as they are easily confused with each other.

In contrast to our VTCL method, TCL fails to properly distinguish the difference between intra- and inter-class variations for some emotions in the three datasets. This confirms our observations introduced in Section 6.2.2 that TCL’s selection of the nearest negative centre is problematic for TER, in which it is often important to use multiple

centres as negative centres. Nonetheless, VTCL proved effective in increasing the variance between negative emotions, which are often positively correlated with each other, demonstrating the benefits of taking all negative emotions into account instead of only the nearest negative centre as is the case in TCL.

6.5 Analysis

6.5.1 Model Predictions

We analysed the model predictions on two different objectives: firstly, the model is trained only with the cross-entropy loss, and subsequently, it is jointly trained with VTCL. Our research hypothesis is that including VTCL in the emotion classification loss (i.e., cross-entropy) can generate more discriminative features and thus increase the model prediction scores. For this analysis, we use the CNN network architecture and hyper-parameters discussed in Section 6.3. For each dataset, we randomly selected one example per emotion class whose scores are correctly predicted by the two objectives mentioned above and extracted their prediction scores with respect to each emotion class.

In Figure 6.2, the graphs illustrate prediction scores when the model is trained without VTCL (left-hand graphs) and with VTCL (right-hand graphs). The sub-figures from top to bottom correspond to the instances extracted from IEST, ISEAR and TEC datasets, respectively. In the top group of sub-figures (corresponding to examples from the IEST dataset), it can be observed that for the model trained without VTCL overlaps with other emotion classes and as a result, the prediction score for the correct emotion class is low and is close to the prediction scores for other emotion classes. However, when the model is trained jointly with VTCL, a much higher prediction score is achieved for the correct emotion, which is well distinguished from all the other emotion classes. Figure 6.2b shows the scores of the “disgust” class (i.e., without vs with VTCL), demonstrating the improvement brought by our approach in increasing the correct prediction score, as well as reducing the overlap with other highly correlated emotions (e.g., anger and fear). A similar pattern can be observed for the other instances belonging both to the same dataset (i.e., IEST) and to the ISEAR and TEC datasets, thus supporting our hypothesis that the incorporation of both intra- and inter-class variations into the task of TER increases performance by introducing discriminative features.

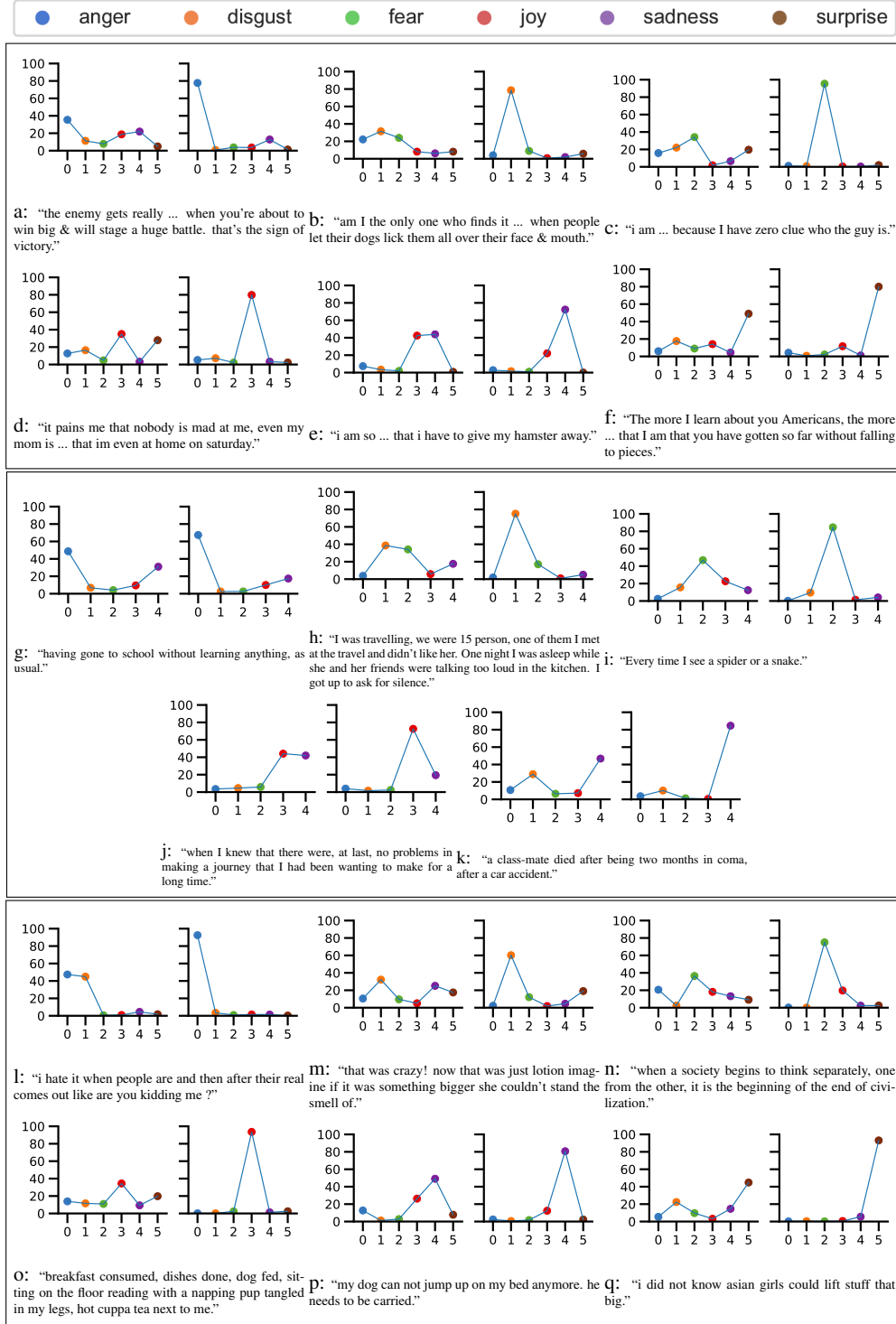


Figure 6.2: Prediction scores (y-axis) across emotions (x-axis). Each sub-figure shows the scores of the two evaluated objectives, i.e., without VTCL (left) vs with VTCL (right). The corresponding instance to be classified is included at the bottom of each sub-figure. The frame from top-to-bottom represents instances belonging to IEST, ISEAR and TEC datasets, respectively. "...": refers to the removed triggered word from the IEST dataset (© 2021 IEEE).

6.5.2 Visualisation of Learned Representations

To provide insights into the ability of our method to introduce discriminative features, we selected the penultimate layer of BERT and CNN, and then used t-SNE (Maaten and Hinton, 2008) to visualise the learned features. For this analysis, we randomly chose 1,000 examples from the test set of IEST data and then trained models by following the same two objectives discussed in the above section (i.e., training the model without VTCL vs with VTCL).

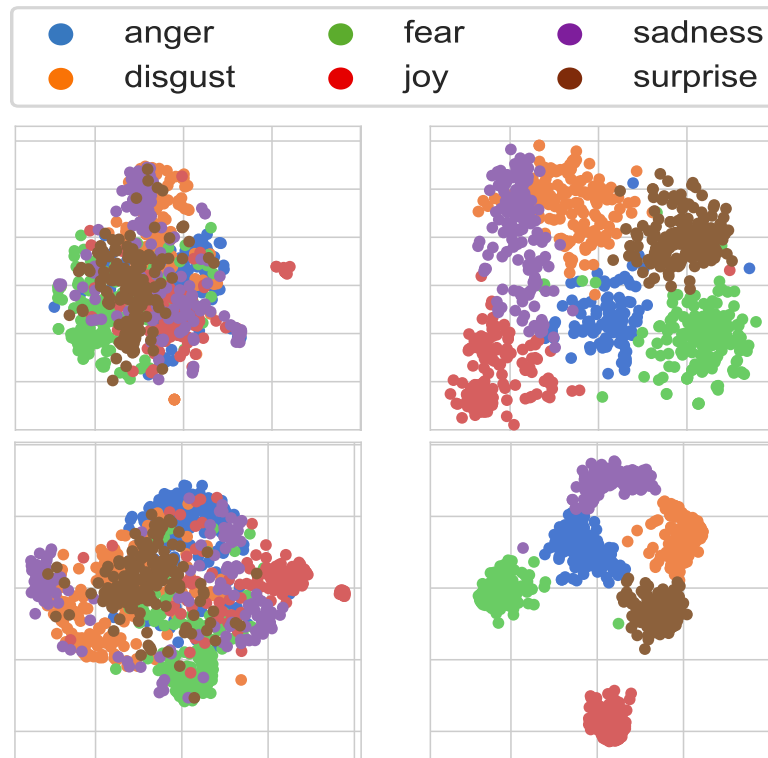


Figure 6.3: t-SNE feature visualisation of CNN (top graphs) and BERT (bottom graphs). The left- and right-hand graphs illustrate features of the model trained without VTCL and with VTCL, respectively. All four plots share the same colour scheme, as defined at the top of the figure (© 2021 IEEE).

Figure 6.3 visualises the learned features for each emotion label, from which we observe some positive properties: i) The first objective performs poorly in learning compact and discriminative features, whereas the second one is able to simultaneously create compact and more clearly separated clusters. In other words, our method ensures that the learned embeddings of the same emotion label are as close as possible to each other, but also as distant as possible from other emotions. ii) The deeply learned representations from BERT are more clearly separated and compact than the

ones obtained from CNN, which is not surprising, as it achieves the highest results when trained jointly with our method. Overall, the visualisations serve to reinforce the benefits of our method in terms of decreasing intra-class variance between examples sharing the same emotion as well as increasing their inter-class variances with other emotions.

6.5.3 Qualitative analysis

We carried out a qualitative analysis of the predictions made by each objective. We observe that in many cases, the second objective (i.e., training the model with VTCL) performs better than the first objective (i.e., training it without VTCL). Table 6.8 presents the analysis. Since some emotions share similarities in linguistic expressions, the model can easily confuse and mislabel emotions. This problem mainly appears in negative emotions (i.e., anger, fear, disgust and sadness). We also note that the main sources of errors made by the first objective are cases involving strong expressions of one emotion over another, implicit emotions and certain lexical units.

Dataset	Text	Actual	w/ VTCL	w/o VTCL
IEST	I get so [trigger_word] when parents smoke right next to their little kids.	disgust	disgust	anger
	I think i will finally be [trigger_word] when i go to a fete. just need to get rid of this stress.	joy	joy	sadness
	I love you so much and i am [trigger_word] because you do not that i exist.	sadness	sadness	joy
ISEAR	Someone told me that i was chosen for english lectures because the class leader is going out with me (not true).	anger	anger	disgust
	When i heard that a woman of my community had aborted and got rid of the foetus by throwing it in the drain.	disgust	disgust	sadness
	Doing unexpectedly well in an examn.	joy	joy	sadness
TEC	The cock who keeps pushing his chair onto my legs needs to stop.	anger	anger	sadness
	That feeling you get when you open up a bill and there's a credit. no payment required.	joy	joy	anger
	Ever wish you could go back a few years , and do it all differently.	sadness	sadness	joy

Table 6.8: Analysis of the model predictions trained on two settings, i.e., w/ VTCL vs w/o VTCL (© 2021 IEEE).

For example, the first, second and final examples of the IEST dataset show implicit emotion instances, which led the model to select incorrect predictions. Moreover, the presence of the strong expressions “love you so much” and “get rid of this stress” in

the fourth and fifth examples of the IEST dataset confuse the model with the first objective, such that it selects incorrect predictions. In contrast, the model trained with the second objective is able to overcome these potential confusions and predict the correct emotion. Similar patterns are also seen in ISEAR and TEC datasets. Overall, introducing discriminative features helps the model overcome the above-discussed problems and predict the correct emotion labels with high probabilities, thus supporting our hypothesis regarding the importance of incorporating intra- and inter-class variations for TER.

6.5.4 Selecting the number of negative centres

Figure 6.4 presents the results of selecting varying numbers of negative centres for each dataset. It should be noted that ISEAR contains a maximum of four negative centres, because it only consists of five classes of Ekman’s (Ekman, 1992) basic emotions (i.e., anger, disgust, fear, joy and sadness).

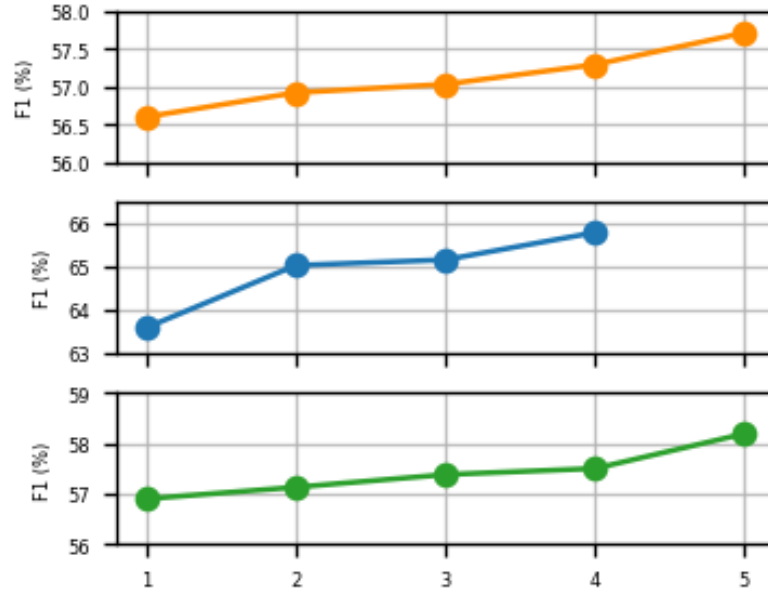


Figure 6.4: Our method with a range of C negative centres (x-axis). For the computation of inter-class, the numbers from 1-4 represents the top- k negative centres, while the last one combines all negative centres via summation (i.e. VTCL). The sub-figures from top to bottom represent IEST, ISEAR and TEC datasets, respectively.

Figure 6.4 shows that the greater the number of negative centres that are combined together in the computation of inter-class distance, the better the performance; the same trend is observed across all three datasets. These findings also confirm our hypothesis

that combining all negative centres (i.e., VTCL) via summation helps to simplify our method as well as to ensure that the intra-class distance is minimised within the same emotion class, while the inter-class distance is maximised between different emotion classes. In other words, our method optimises the inter-class distance to be larger than the intra-class distance plus the margin. In short, our method addresses the problem of selecting the nearest negative centre, which is the case in TCL, by combining all negative centres. This is especially beneficial for TER, where multiple centres can be potentially used as negative ones.

6.6 Summary

In this chapter, we addressed our third research question (**RQ#3**) and proposed a novel objective for emotion classification. We specifically introduced variant triplet centre loss (VTCL) that aims to disentangle positive emotions (i.e., correct labels) from negative ones. We further presented a new evaluation method to quantify the benefits of intra- and inter-class variations on each emotion class.

VTCL contains two terms or components, which are responsible for computing both intra- and inter-class variations within and between emotions. The intra-class term modelled examples labelled with the same emotion class, whereas the inter-class modelled the same set of examples with the other emotions. In this respect, the first term pulls the hidden representation (i.e., features) of each example as close as possible to its corresponding centroid, while the second term pushes the hidden representation of the same example as far as possible from the other emotion centroids.

We evaluated our method on three popular single-label emotion corpora. We demonstrated that VTCL outperformed previous approaches reported in the literature on the three emotion corpora. Our empirical evaluation also showed the effectiveness of incorporating both intra- and inter-class information into TER, demonstrating the ability of this information not only to increase the model prediction scores, but also to help more clearly distinguish between different emotions, especially those highly confused with each other. Our evaluation further demonstrated the advantages and utility of VTCL as an auxiliary loss for emotion classification.

We conducted an in-depth analysis of VTCL by focusing on three aspects: 1) the benefits of VTCL to improve predictions and to reduce overlap, 2) the ability of VTCL to introduce discriminative features, 3) the selection of the number of negative centres. Our observation regarding the first analysis demonstrated that VTCL can help

TER models improve their performance and discriminator ability. For example, when VTCL was added to TER models, it helped them achieve high prediction scores and reduce the overlap with other emotions. The second analysis also revealed the usefulness of VTCL in introducing discriminative features between examples that share dissimilar emotions as well as learning compact features between examples that share the same emotion. The last analysis further corroborated the advantages of considering all but the one positive class centre as negative centres. This helped to simplify our method and ensure that the intra-class distance is minimised within the same emotion class, whereas the inter-class distance is maximised between different emotion classes.

From the above discussion, we concluded that the task of recognising emotion expressions can be modelled better by incorporating intra- and inter-class variations. The incorporation of both intra- and inter-class variations into TER models can help address the problem of the highly confused emotions to some extent. Our method can be easily applied to any types of neural networks without requiring any modifications.

Finally, the main attributes of our work can be summarised as follows: 1) the incorporation of both intra- and inter-class variations into TER models, 2) the introduced evaluation method to quantify the benefits of intra- and inter-class variations on each emotion class, 3) the improvement of the discriminative ability of TER models and 4) the independence from emotion lexicons as well as theories of emotion in incorporating both intra- and inter-class variations.

Chapter 7

Conclusion

This thesis studies the task of Textual Emotion Recognition (TER), i.e., the classification of examples into predefined emotion classes (e.g., anger, fear, disgust or joy, among others). The chapters presented in this thesis were organised into: an introductory chapter, a technical background chapter, an overview chapter of TER, three main chapters and the conclusion.

In Chapter 2, we discussed some basic terminologies and tools of neural networks related with our methodology. We started the description with the basic background about the building block of neural networks from one Perceptrons up to multi-layer Perceptrons. We then discussed relevant training procedures, i.e., classification, loss function and learning (i.e., how to train a neural network). Next, we described relevant popular networks, e.g., word representations, convolutional neural networks, recurrent neural networks, attention mechanisms and pre-training of deep bidirectional transformers for language understanding. We finally concluded with different fine-tuning and deep metric learning methods.

In Chapter 3, we first introduced the definitions used in this thesis. Secondly, we described some concepts related to emotion and how they are similar to and different from each other. After that, we conceptualised emotion in text, more specifically the process of experiencing an emotion up until it is received by someone to understand and interpret the emotion expression. Third, we described models of emotion that are concerned with defining how emotions can be categorised and classified into a taxonomy. Then, we presented existing emotion corpora, approaches to TER and common evaluation metrics. Finally, we concluded with some of the limitations and gaps in previous research that this thesis has addressed.

7.1 Contributions

The main aim of this thesis was to build novel computational methods for TER that specifically take correlations/associations into account due to their effectiveness in improving model performance and making it more robust against highly correlated emotions. The proposed computational methods have also enabled TER models to recognise emotion expression, and to incorporate intra- and inter-class variations. We discuss below each of the computational methods developed in this thesis for TER.

Firstly, we proposed SpanEmo casting multi-label classification as span-prediction. Chapter 4 discussed in greater detail the SpanEmo approach and how it learned correlations between emotions as well as associations between emotions and words in the input example. This chapter specifically tackled the first research question (i.e., **RQ#1**) that was concerned with addressing potential ambiguities, in which multiple emotions overlap. To overcome this problem of ambiguity, we utilised co-occurrence statistics from a multi-label emotion corpus, in which each input was labelled with one or multiple emotions. These co-occurrence statistics captured emotion classes that co-exist together, and those that do not. Then, we incorporated the correlations between emotions into the training objective to aid the model to take them into account during the training phase. Since we treated the task as a span-prediction problem, the training objective also enabled the model ability to improve its feature learning. This is because the selected span of emotions was passed into the training objective. In this respect, the learned representations captured associations between emotions and words in the input as well as emotion correlations.

The proposed approach was evaluated both on well-known benchmark multi-label emotion corpus and against state-of-the-art models. Our approach was further tested in three languages, i.e., English, Arabic and Spanish. Based on the evaluations and analyses, we observed the following: 1) Embedding emotion classes with the input improved the model’s ability to recognise emotion expressions, as well as to learn associations between the emotions and words in the input instance. 2) Allowing the model to select a span of emotions directly from the label segment enhanced the learned representations with respect to the correct emotion set that is associated with the input. 3) The training objective modelled the co-existing emotions which were updated during the training phase to account for such information. 4) Our approach can be easily adapted to other languages (i.e., other than English) without requiring any external resources. 5) The model predictions were improved as the number of emotions increased. 6) It also demonstrated that it can learn correlations and associations both at the word-level

and tweet-level. Finally, we advocate the following:

- The task of recognising emotion expressions can be modelled better by taking emotion-specific associations and emotion correlations into account due to the nature of the task in the sense that it is subjective. Taking emotion correlations into account can help overcome the problem of ambiguity between highly correlated emotions, in which they could all be associated with an emotive expression. In this respect, correlations are indispensable for multi-label emotion classification.
- Embedding descriptive label information with an input instance can help TER models learn associations between emotions and words, which in turn reduce the effect of highly correlated emotions and enhance their performance considerably.
- SpanEmo can be easily applied to other TER corpora and languages without requiring any modification in its architecture.

Secondly, we introduced the work of “Incorporating Intra- and Inter-Class Variations into Textual Emotion Recognition”, in which a Variant of Triplet Centre Loss (VTCL) was proposed. Chapter 6 discussed this work in greater detail. More specifically, we defined the concept of intra- and inter-class variations. The intra-class represents examples that share the same emotion class, while the inter-class represents examples that share different emotion classes. The implementation of VTCL is similar to the logic of one-vs-rest, which is optimised to pull examples that share the same emotion class as close as possible to each other, but pushes the same examples as far as possible from other emotion classes. Such work is important for TER models, especially those applied to single-label emotion corpora. Previous research has pointed out that some emotions can be easily confused with each other due to their manifestation in linguistic expression (Mohammad and Bravo-Marquez, 2017a; Agrawal et al., 2018). To address this challenge, we attempted to answer **RQ#3**, which is focused on adapting the concept of correlation (that is always studied in multi-label emotion classification) to single-label emotion corpora. Single-label emotion corpora unfortunately lack multi-label information, making it hard to learn correlations. We based the presented work in this chapter on the idea that semantically similar examples (i.e., those sharing the same emotion) are more likely to have similar emotional expressions. On the other hand, the same examples are more likely to have emotional expressions

dissimilar to the other emotion classes. In this respect, the objective of VTCL aimed to minimise the distance of the examples from the centre within the same emotion class (i.e., intra-class), while maximising their distances from the centres of other emotions classes (i.e., inter-class).

Since our work was the first attempt to learn intra- and inter-class information in this way for TER, we also proposed an evaluation method that helped us test the contribution of our approach to each emotion class. Our evaluations and analyses demonstrated that taking intra- and inter-class variations into account can improve model performance compared to previous approaches, even without using any external resources. The incorporation of both intra- and inter-class variations into the model can help achieve high prediction scores and a better discriminator against highly confused emotions. We conclude the following:

- Intra- and inter-class information are beneficial for textual emotion recognition.
- Training TER models with VTCL can lead to better results and feature learning.
- Intra- and inter-class evaluation demonstrates the benefits of intra- and inter-class variations on each emotion class.

Finally, Chapter 5 discussed two case studies, on which we experimented with our SpanEmo approach. These two studies correspond to two different domains, i.e., Adverse Drug Reaction (ADR) and depression. Prior research has shown that research into TER can contribute to a wide range of applications, from health and well-being to author profiling, marketing, and consumer analysis, among others. We firstly wanted to investigate the contributions of TER to these two tasks. Next, we adapted the architecture of SpanEmo because it is straightforward and easily adaptable to other tasks. In addition, both ADR and depression share some patterns with emotion in the sense that their expressions often contain emotional expressions/keywords.

The first part of Chapter 5 described experiments related to ADRs. We observed through our initial analysis that negative sentiment/emotion is frequently expressed towards ADRs. Based on this observation, we presented a neural model that combines sentiment analysis with transfer learning techniques to improve ADR detection in social media postings. In this respect, our model was first pre-trained on sentiment data

and then fine-tuned on the ADR corpora, following the widely-used two-stage training in Natural Language Processing (NLP). We show that, in combination with rich representations of words and their contexts, transfer learning is beneficial, especially given the large degree of vocabulary overlap between the current affairs posts in the sentiment corpus and posts about ADRs. We also noticed that using the architecture of SpanEmo boosted the model performance by up to 5% in F1-score. This gain demonstrates the straightforward use of SpanEmo for other tasks as well as the important role of embedding the task's classes with the input, which we called the "label-segment". Moreover, we tested the generalisability of the model when tested on data coming from different distributions rather than the one on which it was trained. Although the model performance was dropped, it demonstrated strong performance compared to the baseline, with a marginal difference of approximately 5% on the Daily Strength dataset, while roughly 17% on the Twitter dataset. The high difference between these two datasets is expected because the pre-trained data are collected from Twitter too. This can especially be beneficial for cases when there is no or small labelled data to use for the ADR, which was the case in our work. The evaluation again shows the potential of transfer learning, especially when the model is pre-trained on a related domain to the one under investigation. The use of descriptive label names is also beneficial because it primes the model to focus on associations between the labels and input.

In the task of depression, we instead used our SpanEmo as a feature extractor module due to both the complexity of the task set-up and the small size of data. The rationale for using SpanEmo is because the questions, whose answers our model needs to predict, are related to the emotion classes annotated for the SemEval-2018 datasets (e.g., sadness, pessimism, self-dislike, loss of pleasure, etc.). Through extensive experiments, we noticed that we can predict the level of depression by using a small set of posts written by the same user. This is different from the first setting described in Chapter 5, in which we used all users' posts and achieved low performance. This can be attributed to a number of factors: 1) there were no annotations provided at the post-level which could help identify posts expressing severity signs of depression from those that do not. 2) The second setting is more sensible compared to the first one, because a small set of posts should reveal whether the user shows any sign of depression or not. For example, [Guntuku et al. \(2017\)](#) reported that depressed users can be distinguishable from non-depressed users by patterns in their language and online activity. 3) Using a small number of posts can reduce the noise coming from the large pool of user posts which may not provide useful information about the task of depression.

From our experiments presented in this chapter, we reach the following conclusions:

- Tasks, such as ADR and depression, can benefit from TER because many of the expressions found in the task of TER are similar to those expressions found in ADR and depression.
- The architecture of SpanEmo can be easily applied to other tasks.
- Both transfer learning and SpanEmo can improve the generalisability of the model in cases when there is no or small labelled data.

7.2 Limitations and Future Work

7.2.1 Task Limitations

In this section, we described two broad limitations: The first one is concerned with various issues that influence the performance of TER models (e.g., sarcasm, metaphor, common sense knowledge and social media symbols), and the second one is concerned with the introduced models for TER in this thesis. It should be mentioned that emotional verbal cues are expressed in various ways beyond the use of explicit emotion-based keywords, e.g., joy, anger, sadness, pleased, etc. Such cues come into existence because of social media platforms that enable their users to express emotions in interesting ways by using, for example, emoticons, emojis, hashtags and informal language.

First of all, TER is a subjective task. In some cases, not even a single emotion could be assigned to a single input. The subjectivity of the task requires better handling of emotion expressions which may evoke multiple interpretations. Each of them has specific interpretations that can be assigned with an intensity value among all sets of emotions. We addressed this challenge to some extent in this work by taking multi-label information into account when designing our models. Nevertheless, we require more research to determine the level of interpretation the input can generally express in text. The AffectText is a distribution corpus that contains the intensity value of each emotion for each headline ([Strapparava and Mihalcea, 2007](#)). The intensity value is based on a 100-point scale, which could easily be normalised to percentages between zero and one. Although this corpus was published in 2007 and included 1,200 headlines, it is the only available corpus with distribution information that determines the multiple interpretations occurring in a given input. For example, the headline “Iraq car

bombings kill 22 people, wound more than 60” is annotated with the intensity values of 33%, 40%, 15% and 12% corresponding to fear, surprise, anger and sadness, respectively. We can clearly see how this example is linked with each emotion of Ekman’s six basic emotions. Having such fine-grained information can address the problem of subjectivity and reduce the confusion caused by different interpretations. In this respect, future work in TER would greatly benefit from creating a corpus with the intensity values of an input among all emotions. This corpus can help TER models address the subjectivity of the task because it contains fine-grained information regarding emotion distribution. Such a corpus can also enable the study of all three tasks simultaneously, i.e., single-label emotion classification, multi-label emotion classification and emotion distribution learning.

Secondly, sometimes the problem is not related to the multiple interpretations that a text may convey, but to finding verbal cues to describe feelings using only language. The best example is “an image is worth a thousand words”, which reveals that people can use different modalities to express their feeling/emotion beyond language. Consider the example, “I feel something, but I could not describe it via language”; this does not express any emotion because the author could not find language to describe his/her feelings. In this respect, other modalities can aid TER models to overcome the problem of missing verbal cues to describe feelings of the person who writes the given example. The above example illustrates another limitation of TER models that does not take other modalities into account. We attribute this limitation to the fact that different modalities require different pre-processing and neural network architectures, which increase the requirement of GPU resources.

Figurative Examples (Discussed ↓): Source [Liew et al. \(2016b\)](#)

- S1. “it was!!! I am still on cloud nine! I say and watched them for over two hours. I couldn’t leave! They are incredible!”. (idiomatic Ex)
- S2. “#americanairlines thanks for canceling my flight and rebooking it a day later. You book a specific return time and day for a reason!”. (sarcastic Ex)
- S3. “Loving the #IKEAHomeTour décor #ideas! Between the showroom and the catalog I am in heaven” (metaphoric Ex).

Thirdly, emotion expressions can include figurative language, e.g., sarcastic, idiomatic, metaphoric expressions. If these figurative forms are overlooked, the TER model can then mislabel emotion expressions because the meaning is not explicitly

stated, but involves understanding of non-literal expressions. *S1* illustrates an instance of idiomatic expression, where the phrase “on cloud nine” corresponds to “extremely happy”. These verbal cues are vital for determining the overall emotion expressed in this instance. *S2* also shows an instance of sarcastic expression, causing TER models to be confused by the phrase “so much fun” and as a result they may select the wrong emotion. Lastly, *S3* presents an instance of metaphoric expression that the phrase “in heaven” summarises the overall emotion of the writer. The meaning is not meant to be literal in this context. These variations of expressions have become even widely used in social media due to restrictions of writing up to a maximum of 240 characters in the case of Twitter. The reason we did not examine different types of figurative language extensively in our work is because such a task is beyond the scope of the current work.

Fourth, a sentence such as “Mum, I’ll invite my friends home” does not contain any verbal cues that are directly related to potential emotions. The use of implicit knowledge encountered in everyday events, activities and situations are commonly expressed in text; this common sense knowledge would limit TER models to infer the correct emotion as they do not effectively understand when a given situation or event may trigger which type of emotion. If TER models are enriched by common sense knowledge, they would be able to link the word “friends” with possible options, such as socialisation, having a party, watching a movie, etc. As a result, this sentence would then be more likely labelled with “joy” as humans tend to feel happy when gathering with their friends at home. In this respect, enriching TER models with knowledge related to various events and activities can help them tackle cases that trigger specific emotions as well as those that do not contain explicit cues.

Finally, the challenges discussed above make the task of TER complex. We recommend that future work examines their inter-connection to emotion expression and find potential approaches that can help TER models to incorporate those challenges effectively.

7.2.2 TER Models

In this thesis, we proposed two approaches for the task of TER, i.e., SpanEmo and VTCL. The former was developed for multi-label emotion classification by taking label information, label-label correlations and label-word associations into account in an end-to-end fashion. However, the latter was proposed for single-label emotion classification by firstly extending the notion of correlations into single-label emotion and then incorporating them into the classification training objective. We now turn to describe

each of their limitations and some penitential future directions that can improve them.

SpanEmo. Let us first emphasise the main components of our SpanEmo approach, i.e., the input, the encoder network, the feed-forward network and the training objective. For the input component, we embedded label information with the input instance to enable the model to drive associations between them. This was possible because the model assigned different embedding segments for each one of them. Then, the input component was fed into the encoder based on BERT to learn a feature representation for each token. The BERT encoder can process up to a maximum of 512 tokens, which was not a problem for our work since we dealt with social media data; they are often short. Nevertheless, this can be a problem when working with long text or documents that require us to find ways in order to address the window restriction of the encoder. Also, embedding all labels with each input is a reasonable and straightforward approach to prevent the model from only relying on the correct set of emotions. It would be more interesting to include the correct set, while ensuring that the model does not over-fit them.

Next, the feed-forward component is responsible for transforming each token representation into a single score so that scores corresponding to the label segment can be used directly for prediction. Apart from only transforming each token into a single score, this component can be extended further to model each emotion label and its associations with any words explicitly, which is similar to a question answering task. In question answering, the model is provided with a question and it has to find an answer to the question in a passage. This way of finding the answer is known as span-prediction (Joshi et al., 2020). In this respect, the question can be an emotion class and the passage can be the input instance. It would be interesting to investigate this approach and its applicability to emotion classification.

Finally, the training objective is concerned with incorporating label-label correlations taken from co-occurrence statistics in the multi-label emotion corpus and is then jointly optimised with the binary cross-entropy loss. Both objectives are combined via an alpha parameter that is constant during the training phase. However, allowing the model to learn this parameter dynamically can be beneficial for the task, where some mini-batches may have inputs with insufficient information about correlations. In this case, the binary cross-entropy should be assigned more weights than the correlation loss, but if the input carries more information about the correlations, the correlation loss should be assigned more weights than the binary cross-entropy loss. This is also motivated by the analysis performed in Figure 4.5, in which the influence of the alpha

parameter was tested.

VTCL. We proposed VTCL as an auxiliary task to emotion classification. In Chapter 6, we conceptualised VTCL based on two terms, i.e., intra- and inter-class variations. The intra-class represents inputs that share the same emotion class, but the inter-class represents inputs that share dissimilar emotions. The overall objective of VTCL is to minimise the distance of the examples from the centre within the same emotion class (intra-class), while maximising their distances from the centres of other emotions classes (inter-class). In this respect, we treated the input with the correct emotion class as positive, while treating the same input to other emotion classes as negative. This is motivated by the notion of the One-Versus-all logic in terms of disentangling the positive example from the features of negative centres. Although this setup makes sense for single-label emotion classification, it is also possible to find some examples which may be associated with more than one emotion. Consider the example, “I love you so much and i am [trigger word] because you do not know that i exist.”, which is labelled with “sadness” although “anger” makes sense in this context. Based on this example, having all emotion classes that are incorrect as negative, is sub-optimal for cases like this one, where the anger class should not be treated the same way as joy. It would be interesting to capture such variations between emotions even if they are negative from the viewpoint of single-label emotion corpora. This can be an important extension direction for future work.

Another issue emerged from the use of VTCL is the initialisation of centres. We initialised the parametric centres randomly in VTCL and updated them based on the mini-batches. At the beginning of the training phase, it was often difficult to achieve accurate class centres and thus, we opted to train it jointly with the classification loss to overcome this issue. However, there can be other ways to compute the parametric centres, which require further examination in future work. It would even be interesting to enable the VTCL to perform classification instead of feature discrimination. This is because VTCL and cross-entropy loss share common patterns in the sense that both optimise the centres weights as well as the weights of the fully-connected layer to be close to the deep features in terms of the chosen distance metric (e.g. Euclidean distance). This direction can help overcome the problem of initialising the centres from scratch, by benefiting from the weights of the fully-connected layer.

7.2.3 Future Work

The Adoption of SpanEmo to Other Corpora, Languages and Tasks. One of the contributions of SpanEmo is that it can be easily applied to other corpora and languages without requiring any modification in its architecture. This was confirmed by testing it on three different languages (i.e., English, Arabic and Spanish) and on two different case studies. In addition, SpanEmo was specifically proposed for the task of multi-label emotion classification, but it can be still applied to other multi-label classification tasks. In a similar vein, multi-class classification can also benefit from the SpanEmo model as is the case with the task of ADR, which is simply a binary classification problem. The above-discussed points demonstrate the strength of our model in making it easily adaptable to other languages, corpora and tasks. The same can be also applied to our **VTCL method**, which was tested on three different emotion corpora as well as on two different networks.

The Benefits of SpanEmo on Explainable Artificial Intelligence (XAI). This thesis came over the notion of XAI via the analysis of SpanEmo results in Chapter 4. More specifically, we illustrated in Figure 4.3 how SpanEmo can support interpretation of the model behaviour, from which we found associations between emotion labels and words in the input instance. We observed through this example that the model can take advantage of contextual information and emotion correlations. Furthermore, the model was able to capture the association between a phrase (i.e., about to join the police) and an emotion label (i.e., anticipation) that was not part of the correct label set. By using such analysis, we can seek answers for some questions as follows: 1) Can the model handle negations?, 2) Can the model learn from contextual information?, Can the model learn association between an emotion label and a phrase? and Can the following idiom holds "Birds of a feather fly together" in the case of predicting highly correlated emotions? These are just a few questions that can be possibly answered by looking at Figure 4.3. We believe that this type of analysis is crucial for interpreting and understanding the model behaviour in selecting one emotion label over another. Therefore, research in XAI can take advantage of such work in understanding model behaviours on tasks beyond the one being investigated in this thesis. In NLP research, this type of analysis is also known as "saliency-based visualisations" (SBVs). [Danilevsky et al. \(2020\)](#) states that SBVs are commonly used now days because they highlight explanations in a visual way, which make it easily understood by different types of end users.

Bibliography

- Muhammad Abdul-Mageed and Lyle Ungar. 2017. [EmoNet: Fine-Grained Emotion Detection with Gated Recurrent Neural Networks](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 718–728.
- Ameeta Agrawal and Aijun An. 2012. Unsupervised emotion detection from text using semantic and syntactic relations. In *Proceedings of the The 2012 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01*, pages 346–353. IEEE Computer Society.
- Ameeta Agrawal and Aijun An. 2016. Selective co-occurrences for word-emotion association. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1579–1590.
- Ameeta Agrawal, Aijun An, and Manos Papagelis. 2018. Learning emotion-enriched word representations. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 950–961.
- Shad Akhtar, Deepanway Ghosal, Asif Ekbal, Pushpak Bhattacharyya, and Sadao Kurohashi. 2019. All-in-one: Emotion, sentiment and intensity prediction using a multi-task ensemble framework. *IEEE Transactions on Affective Computing*.
- Yaara Benger Alaluf and Eva Illouz. 2019. Emotions in consumer studies. *The Oxford Handbook of Consumption*, page 239.
- H. Alhuzali and S. Ananiadou. 2021a. [Improving textual emotion recognition based on intra- and inter-class variation](#). *IEEE Transactions on Affective Computing*, (01):1–1.
- Hassan Alhuzali, Muhammad Abdul-Mageed, and Lyle Ungar. 2018a. [Enabling deep](#)

- learning of emotion with first-person seed expressions. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, pages 25–35, New Orleans, Louisiana, USA. Association for Computational Linguistics.
- Hassan Alhuzali, Muhammad Abdul-Mageed, and Lyle Ungar. 2018b. Enabling deep learning of emotion with first-person seed expressions. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, pages 25–35.
- Hassan Alhuzali and Sophia Ananiadou. 2019. [Improving classification of adverse drug reactions through using sentiment analysis and transfer learning](#). In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 339–347, Florence, Italy. Association for Computational Linguistics.
- Hassan Alhuzali and Sophia Ananiadou. 2021b. Spanemo: Casting multi-label emotion classification as span-prediction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1573–1584.
- Hassan Alhuzali, Mohamed Elaraby, and Muhammad Abdul-Mageed. 2018c. [UBC-NLP at IEST 2018: Learning implicit emotion with an ensemble of language models](#). In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 342–347, Brussels, Belgium. Association for Computational Linguistics.
- Hassan Alhuzali, Mohamed Elaraby, and Muhammad Abdul-Mageed. 2018d. [UBC-NLP at IEST 2018: Learning Implicit Emotion With an Ensemble of Language Models](#). In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 342–347.
- Hassan Alhuzali, Tianlin Zhang, and Sophia Ananiadou. 2021. Predicting sign of depression via using frozen pre-trained models and random forest classifier. In *CLEF-2021*, pages 888–896.
- Ilseyar Alimova and Elena Tutubalina. 2017. Automated detection of adverse drug reactions from social media posts with machine learning. In *International Conference on Analysis of Images, Social Networks and Texts*, pages 3–15. Springer.

- Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat. 2005. Emotions from text: machine learning for text-based emotion prediction. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 579–586. Association for Computational Linguistics.
- Nourah Alswaidan and Mohamed El Bachir Menai. 2020a. Hybrid feature model for emotion recognition in arabic text. *IEEE Access*, 8:37843–37854.
- Nourah Alswaidan and Mohamed El Bachir Menai. 2020b. A survey of state-of-the-art approaches for emotion recognition in text. *Knowledge & Information Systems*, 62(8).
- D. G. Altman. 1991. *Practical Statistics for Medical Research*. Chapman & Hall / CRC, London.
- Saima Aman and Stan Szpakowicz. 2007. Identifying expressions of emotion in text. In *International Conference on Text, Speech and Dialogue*, pages 196–205. Springer.
- Saima Aman and Stan Szpakowicz. 2008. Using roget’s thesaurus for fine-grained emotion recognition. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*.
- Mario Ezra Aragón, Adrian Pastor López-Monroy, Luis Carlos González-Gurrola, and Manuel Montes. 2019. Detecting depression in social media using fine-grained emotions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1481–1486.
- Gilbert Badaro, Obeida El Jundi, Alaa Khaddaj, Alaa Maarouf, Raslan Kain, Hazem Hajj, and Wassim El-Hajj. 2018. [EMA at SemEval-2018 task 1: Emotion mining for Arabic](#). In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 236–244, New Orleans, Louisiana. Association for Computational Linguistics.
- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.
- Angelo Basile, Mara Chineza-Rios, Ana-Sabina Uban, Thomas Müller, Luise Rössler,

- Seren Yenikent, Berta Chulví, Paolo Rosso, and Marc Franco-Salvador. 2021. [Upv-symanto at erisk 2021: Mental health author profiling for early risk prediction on the internet](#). In *CLEF-2021*, pages 908–927.
- Christos Baziotis, Nikos Athanasiou, Alexandra Chronopoulou, Athanasia Kolovou, Georgios Paraskevopoulos, Nikolaos Ellinas, Shrikanth Narayanan, and Alexandros Potamianos. 2018. Ntua-slp at semeval-2018 task 1: predicting affective content in tweets with deep attentive rnns and transfer learning. *arXiv preprint arXiv:1804.06658*.
- Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017. Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 747–754.
- Aaron T Beck, Calvin H Ward, Mock Mendelson, Jeremiah Mock, and John Erbaugh. 1961. An inventory for measuring depression. *Archives of general psychiatry*, 4(6):561–571.
- Jiang Bian, Umit Topaloglu, and Fan Yu. 2012. Towards large-scale twitter mining for drug-related adverse events. In *Proceedings of the 2012 international workshop on Smart health and wellbeing*, pages 25–32. ACM.
- Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.
- Sven Buechel and Udo Hahn. 2017. Emobank: Studying the impact of annotation perspective and representation format on dimensional emotion analysis. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 578–585.
- Fidel CACHED, Diego Fernández Iglesias, Francisco Javier Nóvoa, and Victor Carneiro. 2018. Analysis and experiments on early detection of depression. *CLEF (Working Notes)*, 2125.
- Xin Cai, Li Liu, Lei Zhu, and Huaxiang Zhang. 2021. Dual-modality hard mining triplet-center loss for visible infrared person re-identification. *Knowledge-Based Systems*, 215:106772.

- Erik Cambria, Dipankar Das, Sivaji Bandyopadhyay, Antonio Feraco, et al. 2017. *A practical guide to sentiment analysis*. Springer.
- Rich Caruana, Steve Lawrence, and Lee Giles. 2001. Overfitting in neural nets: Back-propagation, conjugate gradient, and early stopping. *Advances in neural information processing systems*, pages 402–408.
- Augustin Cauchy et al. 1847. Méthode générale pour la résolution de systèmes d'équations simultanées. *Comp. Rend. Acad. Sci. Paris*, 25(1847):536–538.
- José Cañete, Gabriel Chaperon, Rodrigo Fuentes, and Jorge Pérez. 2020. Spanish pre-trained bert model and evaluation data. In *to appear in PMLADC at ICLR 2020*.
- Xuetong Chen, Martin D Sykora, Thomas W Jackson, and Suzanne Elayan. 2018. What about mood swings: Identifying depression on twitter with temporal measures of emotions. In *Companion Proceedings of the The Web Conference 2018*, pages 1653–1660.
- Jiajun Cheng, Shenglin Zhao, Jiani Zhang, Irwin King, Xin Zhang, and Hui Wang. 2017. Aspect-level sentiment classification with heat (hierarchical attention) network. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 97–106.
- Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 484–494.
- Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734.
- Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE.

- Alexandra Chronopoulou, Christos Baziotis, and Alexandros Potamianos. 2019. An embarrassingly simple approach for transfer learning from pretrained language models. *arXiv preprint arXiv:1902.10547*.
- Alexandra Chronopoulou, Aikaterini Margatina, Christos Baziotis, and Alexandros Potamianos. 2018. Ntua-slp at iest 2018: Ensemble of neural transfer methods for implicit emotion classification. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 57–64.
- Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What does bert look at? an analysis of bert’s attention. *ACL 2019*, page 276.
- Alex Clibbon. 2020. [Decoding emotion in consumer feedback to drive long-term brand growth](#). CX-Emotion.
- Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167.
- Marina Danilevsky, Kun Qian, Ranit Aharonov, Yannis Katsis, Ban Kawas, and Prithviraj Sen. 2020. [A survey of the state of explainable ai for natural language processing](#).
- Dorottya Demszyk, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. Goemotions: A dataset of fine-grained emotions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4040–4054.
- Jiawen Deng and Fuji Ren. 2021. A survey of textual emotion recognition and its challenges. *IEEE Transactions on Affective Computing*.
- Shrey Desai, Cornelia Caragea, and Junyi Jessy Li. 2020. [Detecting perceived emotions in hurricane disasters](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5290–5305, Online. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.
- ZHOU Deyu, Xuan Zhang, Yin Zhou, Quan Zhao, and Xin Geng. 2016. Emotion distribution learning from texts. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 638–647.
- Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. [The hitchhiker’s guide to testing statistical significance in natural language processing](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia. Association for Computational Linguistics.
- Monireh Ebrahimi, Amir Hossein Yazdavar, Naomie Salim, and Safaa Eltyeb. 2016. Recognition of side effects as implicit-opinion words in drug reviews. *Online Information Review*, 40(7):1018–1032.
- Paul Ekman. 1992. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200.
- Golnoosh Farnadi, Geetha Sitaraman, Mehrdad Rohani, Michal Kosinski, David Stillwell, Marie-Francine Moens, Sergio Davalos, and Martine De Cock. 2014. How are you doing? emotions and personality in facebook. In *Proceedings of the EMPIRE Workshop of the 22nd International Conference on User Modeling, Adaptation and Personalization (UMAP 2014)*.
- Hao Fei, Yafeng Ren, and Donghong Ji. 2019. Implicit objective network for emotion detection. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 647–659. Springer.
- Hao Fei, Yue Zhang, Yafeng Ren, and Donghong Ji. 2020. Latent emotion memory for multi-label emotion classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7692–7699.
- Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017.

- Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1615–1625.
- Pascale Fung. 2015. Robots with heart. *Scientific American*, 313(5):60–63.
- Junling Gao, Pinpin Zheng, Yingnan Jia, Hao Chen, Yimeng Mao, Suhong Chen, Yi Wang, Hua Fu, and Junming Dai. 2020. Mental health problems and social media exposure during covid-19 outbreak. *Plos one*, 15(4):e0231924.
- Radhika Gaonkar, Heeyoung Kwon, Mohaddeseh Bastan, Niranjan Balasubramanian, and Nathanael Chambers. 2020a. [Modeling label semantics for predicting emotional reactions](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4687–4692, Online. Association for Computational Linguistics.
- Radhika Gaonkar, Heeyoung Kwon, Mohaddeseh Bastan, Niranjan Balasubramanian, and Nathanael Chambers. 2020b. Modeling label semantics for predicting emotional reactions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4687–4692.
- Siddhant Garg, Thuy Vu, and Alessandro Moschitti. 2020. Tanda: Transfer and adapt pre-trained transformer models for answer sentence selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7780–7788.
- Xin Geng. 2013. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28:1734–1748.
- Diman Ghazi, Diana Inkpen, and Stan Szpakowicz. 2010. Hierarchical versus flat classification of emotions in text. In *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text*, pages 140–146. Association for Computational Linguistics.
- Rachel Ginn, Pranoti Pimpalkhute, Azadeh Nikfarjam, Apurv Patki, Karen O’Connor, Abeer Sarker, Karen Smith, and Graciela Gonzalez. 2014. Mining twitter for adverse drug reaction mentions: a corpus and classification benchmark. In *Proceedings of the fourth workshop on building and evaluating resources for health and biomedical text processing*, pages 1–8. Citeseer.

- Sujatha Das Gollapalli, Polina Rozenshtein, and See Kiong Ng. 2020. Ester: Combining word co-occurrences and word associations for unsupervised emotion detection. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1043–1056.
- José-Ángel González, Lluís-F. Hurtado, and Ferran Pla. 2018. [ELiRF-UPV at SemEval-2018 tasks 1 and 3: Affect and irony detection in tweets](#). In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 565–569, New Orleans, Louisiana. Association for Computational Linguistics.
- Margherita Grandini, Enrico Bagli, and Giorgio Visani. 2020. Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756*.
- Sharath Chandra Guntuku, Daniel Preotiuc-Pietro, Johannes C Eichstaedt, and Lyle H Ungar. 2019. What twitter profile and posted images reveal about depression and anxiety. In *Proceedings of the international AAAI conference on web and social media*, volume 13, pages 236–246.
- Sharath Chandra Guntuku, David B Yaden, Margaret L Kern, Lyle H Ungar, and Johannes C Eichstaedt. 2017. Detecting depression and mental illness on social media: an integrative review. *Current Opinion in Behavioral Sciences*, 18:43–49.
- Narendra Gupta, Mazin Gilbert, and Giuseppe Di Fabbrizio. 2010. Emotion detection in email customer care. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 10–16.
- Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don’t stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.
- Rave Harpaz, Alison Callahan, Suzanne Tamang, Yen Low, David Odgers, Sam Finlayson, Kenneth Jung, Paea LePendou, and Nigam H. Shah. 2014. [Text mining for adverse drug events: the promise, challenges, and state of the art](#). *Drug Safety*, 37(10):777–790.
- Huihui He and Rui Xia. 2018. Joint binary neural network for multi-label learning with applications to emotion classification. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 250–259. Springer.

- Xinwei He, Yang Zhou, Zhichao Zhou, Song Bai, and Xiang Bai. 2018. Triplet-center loss for multi-view 3d object retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1945–1954.
- Jonathan Herzig, Guy Feigenblat, Michal Shmueli-Scheuer, David Konopnicki, and Anat Rafaeli. 2016. Predicting customer satisfaction in customer support conversations in social media using affective features. In *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization*, pages 115–119.
- Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 328–339.
- Chenyang Huang, Amine Trabelsi, Xuebin Qin, Nawshad Farruque, Lili Mou, and Osmar Zaiane. 2021. [Seq2Emo: A sequence to multi-label emotion classification model](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4717–4724, Online. Association for Computational Linguistics.
- Trung Huynh, Yulan He, Alistair Willis, and Stefan Rüger. 2016. Adverse drug reaction classification with deep neural networks. *Coling*.
- Jumayel Islam, Robert E. Mercer, and Lu Xiao. 2019a. [Multi-channel convolutional neural network for twitter emotion and sentiment recognition](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1355–1365, Minneapolis, Minnesota. Association for Computational Linguistics.
- Jumayel Islam, Robert E Mercer, and Lu Xiao. 2019b. Multi-channel convolutional neural network for twitter emotion and sentiment recognition. In *Proceedings of*

- the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1355–1365.
- Carroll E Izard. 2010. The many meanings/aspects of emotion: Definitions, functions, activation, and regulation. *Emotion Review*, 2(4):363–370.
- Spencer L James, Degu Abate, Kalkidan Hassen Abate, Solomon M Abay, Cristiana Abbafati, Nooshin Abbasi, Hedayat Abbastabar, Foad Abd-Allah, Jemal Abdela, Ahmed Abdelalim, et al. 2018. Global, regional, and national incidence, prevalence, and years lived with disability for 354 diseases and injuries for 195 countries and territories, 1990–2017: a systematic analysis for the global burden of disease study 2017. *The Lancet*, 392(10159):1789–1858.
- Keyuan Jiang and Yujing Zheng. 2013. Mining twitter data for potential drug effects. In *International conference on advanced data mining and applications*, pages 434–443. Springer.
- Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.
- Neel Kant, Raul Puri, Nikolai Yakovenko, and Bryan Catanzaro. 2018. Practical text classification with large pre-trained language models. *arXiv preprint arXiv:1812.01207*.
- Jussi Karlgren, Magnus Sahlgren, Fredrik Olsson, Fredrik Espinoza, and Ola Hamfors. 2012. Usefulness of sentiment analysis. In *European Conference on Information Retrieval*, pages 426–435. Springer.
- Taha A Kass-Hout and Hend Alhinnawi. 2013. Social media in public health. *Br Med Bull*, 108(1):5–24.
- Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491.
- Hamed Khanpour and Cornelia Caragea. 2018. Fine-grained emotion detection in health-related online posts. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1160–1166.

- Yoon Kim. 2014. [Convolutional neural networks for sentence classification](#). In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.
- Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Svetlana Kiritchenko, Saif M Mohammad, Jason Morin, and Berry de Bruijn. 2017. Nrc-canada at smm4h shared task: classifying tweets mentioning adverse drug reactions and medication intake. *arXiv preprint arXiv:1805.04558*.
- Roman Klinger, Orphée De Clercq, Saif Mohammad, and Alexandra Balahur. 2018a. [IEST: WASSA-2018 implicit emotions shared task](#). In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 31–42, Brussels, Belgium. Association for Computational Linguistics.
- Roman Klinger, Orphee De Clercq, Saif Mohammad, and Alexandra Balahur. 2018b. Iest: Wassa-2018 implicit emotions shared task. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 31–42.
- Roman Klinger et al. 2018c. An analysis of annotated corpora for emotion classification in text. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2104–2119.
- Ioannis Korkontzelos, Azadeh Nikfarjam, Matthew Shardlow, Abeed Sarker, Sophia Ananiadou, and Graciela H Gonzalez. 2016. Analysis of the effect of sentiment analysis on extracting adverse drug reactions from tweets and forum posts. *Journal of biomedical informatics*, 62:148–158.
- Adam DI Kramer, Jamie E Guillory, and Jeffrey T Hancock. 2014. Experimental evidence of massive-scale emotional contagion through social networks. *Proceedings of the National Academy of Sciences*, 111(24):8788–8790.
- Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. [ALBERT: A lite BERT for self-supervised learning of language representations](#). *CoRR*, abs/1909.11942.

- Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.
- Quanzhi Li and Sameena Shah. 2017. Learning stock market sentiment lexicon and sentiment-oriented word vector from stocktwits. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 301–310.
- Zhaoqun Li, Cheng Xu, and Biao Leng. 2019a. Angular triplet-center loss for multi-view 3d shape retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8682–8689.
- Zhongyang Li, Xiao Ding, and Ting Liu. 2019b. Story ending prediction by transferable bert. *arXiv preprint arXiv:1905.07504*.
- Zhuowan Li, Quan Tran, Long Mai, Zhe Lin, and Alan L Yuille. 2020. Context-aware group captioning via self-attention and contrastive features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3440–3450.
- Jasy Liew, Suet Yan, and Howard R Turtle. 2016a. Exploring Fine-Grained Emotion Detection in Tweets. In *The 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 73–80.
- Jasy Liew, Suet Yan, Howard R Turtle, and Elizabeth D Liddy. 2016b. EmoTweet - 28: A Fine - Grained Emotion Corpus for Sentiment Analysis. In *Tenth International Conference on Language Resources and Evaluation, LREC 2016*, pages 1149–1156.
- Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130*.
- Bin Liu, Ramesh Govindan, and Brian Uzzi. 2016. Do emotions expressed online correlate with actual changes in decision-making?: The case of stock day traders. *PloS one*, 11(1).
- Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

- Xiao Liu and Hsinchun Chen. 2015. A research framework for pharmacovigilance in health social media: identification and evaluation of patient adverse drug event reports. *Journal of biomedical informatics*, 58:268–279.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. [Roberta: A robustly optimized BERT pretraining approach](#). *CoRR*, abs/1907.11692.
- D. Losada and F. Crestani. 2016. A test collection for research on depression and language use. In *Proc. of Experimental IR Meets Multilinguality, Multimodality, and Interaction, 7th International Conference of the CLEF Association, CLEF 2016*, pages 28–39, Evora, Portugal.
- David E Losada, Fabio Crestani, and Javier Parapar. 2019. Overview of erisk 2019 early risk prediction on the internet. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 340–357. Springer.
- David E Losada, Fabio Crestani, and Javier Parapar. 2020. Overview of erisk at clef 2020: Early risk prediction on the internet (extended overview).
- Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605.
- Sunghwan Mac Kim, Alessandro Valitutti, and Rafael A Calvo. 2010. Evaluation of unsupervised emotion models to textual affect recognition. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 62–70.
- Youness Mansar, Lorenzo Gatti, Sira Ferradans, Marco Guerini, and Jacopo Staiano. 2017. Fortia-fbk at semeval-2017 task 5: Bullish or bearish? inferring sentiment towards brands from financial news headlines. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 817–822.
- Rodrigo Martínez-Castano, Amal Htait, Leif Azzopardi, and Yashar Moshfeghi. 2020. Early risk detection of self-harm and depression severity using bert-based transformers.
- Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th*

international ACM SIGIR conference on research and development in information retrieval, pages 43–52.

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

Tomas Mikolov, Wen-Tau Yih, and Geoffrey Zweig. 2013b. [Linguistic regularities in continuous space word representations](#). *Proceedings of NAACL-HLT*, (June):746–751.

Saif Mohammad. 2012a. # emotional tweets. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics—Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 246–255.

Saif Mohammad. 2012b. [#emotional tweets](#). In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 246–255, Montréal, Canada. Association for Computational Linguistics.

Saif Mohammad. 2012c. [Portable features for classifying emotional text](#). In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 587–591, Montréal, Canada. Association for Computational Linguistics.

Saif Mohammad. 2018. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 english words. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 174–184.

Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. Semeval-2018 task 1: Affect in tweets. In *Proceedings of the 12th international workshop on semantic evaluation*, pages 1–17.

Saif Mohammad and Peter Turney. 2010. [Emotions evoked by common words and phrases: Using Mechanical Turk to create an emotion lexicon](#). In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and*

- Generation of Emotion in Text*, pages 26–34, Los Angeles, CA. Association for Computational Linguistics.
- Saif M. Mohammad and Felipe Bravo-Marquez. 2017a. Emotion intensities in tweets. In *Proceedings of the sixth joint conference on lexical and computational semantics (*Sem)*, Vancouver, Canada.
- Saif M. Mohammad and Felipe Bravo-Marquez. 2017b. WASSA-2017 shared task on emotion intensity. In *Proceedings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Copenhagen, Denmark.
- Saif M Mohammad and Svetlana Kiritchenko. 2013. Using nuances of emotion to identify personality. *arXiv preprint arXiv:1309.6352*.
- Saif M. Mohammad and Svetlana Kiritchenko. 2015a. [Using hashtags to capture fine emotion categories from tweets](#). *Computational Intelligence*, 31(2):301–326.
- Saif M Mohammad and Svetlana Kiritchenko. 2015b. Using hashtags to capture fine emotion categories from tweets. *Computational Intelligence*, 31(2):301–326.
- Saif M Mohammad and Peter D Turney. 2013a. [Crowdsourcing a word-emotion association lexicon](#). In *Computational Intelligence*, volume 29, pages 436–465.
- Saif M. Mohammad and Peter D. Turney. 2013b. Crowdsourcing a word-emotion association lexicon. 29(3):436–465.
- Hala Mulki, Chedi Bechikh Ali, Hatem Haddad, and Ismail Babaoğlu. 2018. [Tw-StAR at SemEval-2018 task 1: Preprocessing impact on multi-label emotion classification](#). In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 167–171, New Orleans, Louisiana. Association for Computational Linguistics.
- Myriam Munezero, Calkin Suero Montero, Erkki Sutinen, and John Pajunen. 2014. Are they different? affect, feeling, emotion, sentiment, and opinion detection in text. *IEEE transactions on affective computing*, 5(2):101–111.
- Azadeh Nikfarjam, Abeed Sarker, Karen O’connor, Rachel Ginn, and Graciela Gonzalez. 2015. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *Journal of the American Medical Informatics Association*, 22(3):671–681.

- Brendan O'Connor, Michel Krieger, and David Ahn. 2010. Tweetmotif: Exploratory search and topic summarization for twitter. In *Fourth International AAAI Conference on Weblogs and Social Media*.
- Luis Oliveira. 2020. Bioinfo@ uavr at erisk 2020: on the use of psycholinguistics features and machine learning for the classification and quantification of mental diseases.
- Mary Ellen O'Toole. 2009. *The school shooter a threat assessment perspective*. DIANE Publishing.
- Javier Parapar, Patricia Martín-Rodilla, David E Losada, and Fabio Crestani. 2021. Overview of erisk 2021: Early risk prediction on the internet. In *In Proceedings of the Twelfth International Conference of the Cross-Language Evaluation Forum for European Languages*. Springer, Cham.
- Sungjoon Park, Jiseon Kim, Jaeyeol Jeon, Heeyoung Park, and Alice Oh. 2019. Toward dimensional emotion detection from categorical emotion annotations. *arXiv preprint arXiv:1911.02499*.
- Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.
- F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- James W Pennebaker, Ryan L Boyd, Kayla Jordan, and Kate Blackburn. 2015. The development and psychometric properties of liwc2015. Technical report.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.
- Matthew Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018a. Dissecting contextual word embeddings: Architecture and representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509.

- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018b. Deep contextualized word representations. In *Proc. of NAACL*.
- Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018c. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.
- Matthew E Peters, Sebastian Ruder, and Noah A Smith. 2019. To tune or not to tune? adapting pretrained representations to diverse tasks. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 7–14.
- Jason Phang, Thibault Févry, and Samuel R Bowman. 2018. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *arXiv preprint arXiv:1811.01088*.
- Rosalind W Picard. 2000. *Affective computing*. MIT press.
- Robert Plutchik. 1980. *Emotion: A psychoevolutionary synthesis*. Harpercollins College Division.
- Robert Plutchik. 1984. Emotions: A general psychoevolutionary theory. *Approaches to emotion*, 1984:197–219.
- Daniel Preoȃiuc-Pietro, H Andrew Schwartz, Gregory Park, Johannes Eichstaedt, Margaret Kern, Lyle Ungar, and Elisabeth Shulman. 2016. Modelling valence and arousal in facebook posts. In *Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pages 9–15.
- Matthew Purver and Stuart Battersby. 2012. Experimenting with Distant Supervision for Emotion Classification. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 482–491.
- Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, pages 1–26.
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

- Hannah Rashkin, Antoine Bosselut, Maarten Sap, Kevin Knight, and Yejin Choi. 2018. Modeling naive psychology of characters in simple commonsense stories. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2289–2299.
- Ellen Riloff et al. 1993. Automatically constructing a dictionary for information extraction tasks. In *AAAI*, volume 1, pages 2–1. Citeseer.
- Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. Semeval-2017 task 4: Sentiment analysis in twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518.
- Sebastian Ruder. 2016. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.
- David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1986. Learning representations by back-propagating errors. *nature*, 323(6088):533–536.
- James A Russell. 1980. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161.
- James A Russell. 2003. Core affect and the psychological construction of emotion. *Psychological review*, 110(1):145.
- James A Russell and Albert Mehrabian. 1977. Evidence for a three-factor theory of emotions. *Journal of research in Personality*, 11(3):273–294.
- Ali Safaya, Moutasem Abdullatif, and Deniz Yuret. 2020. [Kuisail at semeval-2020 task 12: Bert-cnn for offensive speech identification in social media](#).
- Hariprasad Sampathkumar, Xue-wen Chen, and Bo Luo. 2014. Mining adverse drug reactions from online healthcare forums using hidden markov model. *BMC medical informatics and decision making*, 14(1):91.
- Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. [CARER: Contextualized Affect Representations for Emotion Recognition](#). In *Empirical Methods in Natural Language Processing*, pages 3687–3697.
- Abeed Sarker and Graciela Gonzalez. 2015. Portable automatic text classification for adverse drug reaction detection via multi-corpus training. *Journal of biomedical informatics*, 53:196–207.

- Klaus R Scherer and Harald G Wallbott. 1994. Evidence for universality and cultural variation of differential emotion response patterning. *Journal of personality and social psychology*, 66(2):310.
- Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823.
- Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681.
- H Andrew Schwartz, Johannes Eichstaedt, Margaret Kern, Gregory Park, Maarten Sap, David Stillwell, Michal Kosinski, and Lyle Ungar. 2014. Towards assessing changes in degree of depression through facebook. In *Proceedings of the workshop on computational linguistics and clinical psychology: from linguistic signal to clinical reality*, pages 118–125.
- Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083.
- Armin Seyeditabari, Narges Tabari, and Wlodek Zadrozny. 2018. Emotion detection in text: a review. *arXiv preprint arXiv:1806.00674*.
- Ameneh Gholipour Shahraki and Osmar R Zaiane. 2017. Lexical and learning-based emotion mining from text. In *Proceedings of the international conference on computational linguistics and intelligent text processing*, volume 9, pages 24–55.
- Abu Awal Md Shoeb and Gerard de Melo. 2020. [EmoTag1200: Understanding the association between emojis and emotions](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8957–8967, Online. Association for Computational Linguistics.
- Richard Sloane, Orod Osanlou, David Lewis, Danushka Bollegala, Simon Maskell, and Munir Pirmohamed. 2015. Social media and pharmacovigilance: a review of the opportunities and challenges. *British journal of clinical pharmacology*, 80(4):910–920.

- Christoforos Spertalis, George Drosatos, and Avi Arampatzis. 2021. [Transfer learning for automated responses to the bdi questionnaire](#). In *CLEF-2021*, pages 1046–1058.
- Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. [Dropout: A simple way to prevent neural networks from over-fitting](#). *Journal of Machine Learning Research*, 15(56):1929–1958.
- Carlo Strapparava and Rada Mihalcea. 2007. Semeval-2007 task 14: Affective text. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 70–74. Association for Computational Linguistics.
- Carlo Strapparava and Rada Mihalcea. 2008. Learning to identify emotions in text. In *Proceedings of the 2008 ACM symposium on Applied computing*, pages 1556–1560.
- Carlo Strapparava, Alessandro Valitutti, et al. 2004. Wordnet affect: an affective extension of wordnet. In *Lrec*, volume 4, page 40. Citeseer.
- Calkin Suero Montero and Jarkko Suhonen. 2014. Emotion analysis meets learning analytics: online learner profiling beyond numerical data. In *Proceedings of the 14th Koli calling international conference on computing education research*, pages 165–169.
- Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *China National Conference on Chinese Computational Linguistics*, pages 194–206. Springer.
- Jared Suttles and Nancy Ide. 2013a. Distant supervision for emotion classification with discrete binary values. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 121–136. Springer.
- Jared Suttles and Nancy Ide. 2013b. [Distant supervision for emotion classification with discrete binary values](#). In *International Conference on Intelligent Text Processing and Computational Linguistics*, volume 7817 LNCS, pages 121–136.
- Duyu Tang, Bing Qin, Ting Liu, and Zhenghua Li. 2013. Learning sentence representation for emotion classification on microblogs. In *Natural Language Processing and Chinese Computing*, pages 212–223. Springer.
- Paul Thompson, Sophia Daikou, Kenju Ueno, Riza Batista-Navarro, Jun’ichi Tsujii, and Sophia Ananiadou. 2018. Annotation and detection of drug effects in text for pharmacovigilance. *Journal of cheminformatics*, 10(1):37.

- Suraj Tripathi, Abhiram Ramesh, Abhay Kumar, Chirag Singh, and Promod Yenigalla. 2020. Learning discriminative features using center loss and reconstruction as regularizer for speech emotion recognition. In *Workshop on Artificial Intelligence in Affective Computing*, pages 44–53. PMLR.
- Jay J Van Bavel, Katherine Baicker, Paulo S Boggio, Valerio Capraro, Aleksandra Cichocka, Mina Cikara, Molly J Crockett, Alia J Crum, Karen M Douglas, James N Druckman, et al. 2020. Using social and behavioural science to support covid-19 pandemic response. *Nature human behaviour*, 4(5):460–471.
- Paul Van Rijen, Douglas Teodoro, Nona Naderi, Luc Mottin, Julien Knafo, Matt Jeffries, and Patrick Ruch. 2019. A data-driven approach for measuring the severity of the signs of depression using reddit posts. In *CLEF (Working Notes)*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.
- Christian Voigt, Barbara Kieslinger, and Teresa Schäfer. 2017. User experiences around sentiment analyses, facilitating workplace learning. In *International Conference on Social Computing and Social Media*, pages 312–324. Springer.
- Svitlana Volkova and Yoram Bachrach. 2016. Inferring perceived demographics from user emotional tone and user-environment emotional contrast. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1567–1578.
- C. Wang, H. Dai, F. Wang, and E. C. Su. 2018. [Adverse drug reaction post classification with imbalanced classification techniques](#). In *2018 Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, pages 5–9.
- Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. 2017. Deep metric learning with angular loss. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2593–2601.
- Wenbo Wang, Lu Chen, Krishnaprasad Thirunarayan, and Amit P Sheth. 2012. Harnessing twitter” big data” for automatic emotion identification. In *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom)*, pages 587–592. IEEE.

- Xiangyu Wang and Chengqing Zong. 2021. [Distributed representations of emotion categories in emotion space](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2364–2375, Online. Association for Computational Linguistics.
- Xinyu Wang, Chunhong Zhang, Yang Ji, Li Sun, Leijia Wu, and Zhana Bao. 2013. A depression detection model based on sentiment analysis in micro-blog social network. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 201–213. Springer.
- Yu-Tseng Wang, Hen-Hsen Huang, and Hsin-Hsi Chen. 2018. A neural network approach to early risk detection of depression and anorexia on social media text. In *CLEF (Working Notes)*.
- Amy Beth Warriner, Victor Kuperman, and Marc Brysbaert. 2013. Norms of valence, arousal, and dominance for 13,915 english lemmas. *Behavior research methods*, 45(4):1191–1207.
- Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. 2016. A discriminative feature learning approach for deep face recognition. In *European conference on computer vision*, pages 499–515. Springer.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R’emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface’s transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.
- Chuhan Wu, Fangzhao Wu, Junxin Liu, Sixing Wu, Yongfeng Huang, and Xing Xie. 2018. Detecting tweets mentioning drug name and adverse drug reaction with hierarchical tweet representation and multi-head self-attention. In *Proceedings of the 2018 EMNLP Workshop SMM4H: The 3rd Social Media Mining for Health Applications Workshop and Shared Task*, pages 34–37.
- Shih-Hung Wu and Zhao-Jun Qiu. 2021. [A roberta-based model on measuring the severity of the signs of depression](#). In *CLEF-2021*, pages 1071–1080.
- Rui Xia and Zixiang Ding. 2019. [Emotion-cause pair extraction: A new task to emotion analysis in texts](#). In *Proceedings of the 57th Annual Meeting of the Association*

for Computational Linguistics, pages 1003–1012, Florence, Italy. Association for Computational Linguistics.

Peng Xu, Zihan Liu, Genta Indra Winata, Zhaojiang Lin, and Pascale Fung. 2020a. Emograph: Capturing emotion correlations using graph networks. *arXiv preprint arXiv:2008.09378*.

Peng Xu, Andrea Madotto, Chien-Sheng Wu, Ji Ho Park, and Pascale Fung. 2018. Emo2vec: Learning generalized emotion representation by multi-task training. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 292–298.

Song Xu, Haoran Li, Peng Yuan, Youzheng Wu, Xiaodong He, and Bowen Zhou. 2020b. Self-attention guided copy mechanism for abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1355–1362.

Christopher C Yang, Haodong Yang, Ling Jiang, and Mi Zhang. 2012. Social media mining for drug safety signal detection. In *Proceedings of the 2012 international workshop on Smart health and wellbeing*, pages 33–40. ACM.

Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. Sgm: Sequence generation model for multi-label classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3915–3926.

Andrew Yates and Nazli Goharian. 2013. Adrtrace: detecting expected and unexpected adverse drug reactions from user reviews on social media sites. In *European Conference on Information Retrieval*, pages 816–819. Springer.

Chih-Kuan Yeh, Wei-Chieh Wu, Wei-Jen Ko, and Yu-Chiang Frank Wang. 2017. Learning deep latent spaces for multi-label classification. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 2838–2844.

Wenhao Ying, Rong Xiang, and Qin Lu. 2019. [Improving multi-label emotion classification by integrating both general and domain-specific knowledge](#). In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 316–321, Hong Kong, China. Association for Computational Linguistics.

- Jianfei Yu, Luis Marujo, Jing Jiang, Pradeep Karuturi, and William Brendel. 2018. Improving multi-label emotion classification via sentiment classification with dual attention transfer network. *ACL*.
- Samira Zad and Mark Finlayson. 2020. Systematic evaluation of a framework for unsupervised emotion recognition for narrative text. In *Proceedings of the First Joint Workshop on Narrative Understanding, Storylines, and Events*, pages 26–37.
- Min-Ling Zhang and Zhi-Hua Zhou. 2006. Multilabel neural networks with applications to functional genomics and text categorization. *IEEE transactions on Knowledge and Data Engineering*, 18(10):1338–1351.
- Yuxiang Zhang, Jiamei Fu, Dongyu She, Ying Zhang, Senzhang Wang, and Jufeng Yang. 2018. Text emotion distribution learning via multi-task convolutional neural network. In *IJCAI*, pages 4595–4601.
- Zhenjie Zhao and Xiaojuan Ma. 2019. Text emotion distribution learning from small sample: A meta-learning approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3948–3958.
- Deyu Zhou, Shuangzhi Wu, Qing Wang, Jun Xie, Zhaopeng Tu, and Mu Li. 2020. [Emotion classification by jointly learning to lexiconize and classify](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3235–3245, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Deyu Zhou, Yang Yang, and Yulan He. 2018. Relevant emotion ranking from text constrained with emotion relationships. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 561–571.
- Deyu Zhou, Xuan Zhang, Yin Zhou, Quan Zhao, and Xin Geng. 2016. Emotion distribution learning from texts. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 638–647.
- Suyang Zhu, Shoushan Li, and Guodong Zhou. 2019. Adversarial attention modeling for multi-dimensional emotion regression. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 471–480.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*.

Yimeng Zhuang and Huadong Wang. 2019. Token-level dynamic self-attention network for multi-passage reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2252–2262.