DETECTING DISCRETE EMOTIONS IN TEXT USING NEURAL NETWORKS

by

Seyed Armin Seyeditabari

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Computing & Information Systems

Charlotte

2020

Approved by:

Dr. Wlodek Zadrozny

Dr. Minwoo Lee

Dr. Samira Shaikh

Dr. Xi Niu

Dr. James Walsh

ABSTRACT

SEYED ARMIN SEYEDITABARI. DETECTING DISCRETE EMOTIONS IN TEXT USING NEURAL NETWORKS. (Under the direction of DR. WLODEK ZADROZNY)

In recent years, emotion detection in text has become increasingly popular because of its many potential applications in a range of areas, such as marketing, political science, psychology, human-computer interaction, artificial intelligence. Access to huge amounts of textual data, especially opinionated and self-expression text, has also contributed to bringing attention to this field.

Here, we first review work that has already been done in identifying emotion expressions in text and then proceed to argue that existing techniques, methodologies, and models are incapable of capturing the nuance of emotional language. This is mostly because by using handcrafted features and lexicons, they lose the sequential information inherent in the text and are unable to capture the context. Because existing methods cannot grasp the intricacy of emotional expressions, they are insufficient for creating a reliable and generalizable methodology for emotion detection.

Understanding these limitations, we developed a deep neural network model with bidirectional Gated Recurrent Units (GRUs) and an attention mechanism that does consider the sequential information of text and that can capture the contextual meaning of words. Because our emotion detection model captures a more informative representation of the text, its performance is significantly better than conventional machine learning models. Specifically, our model increases the F-measure on the test data by 26.8 points, and by 38.6 points on a dataset never seen before. We also compared our model to fine-tuned transformer model (BERT), and found that the performance was slightly better specially using emotional embeddings, and importantly, required

only a fraction of the computational power. In addition to this model, we also developed a new methodology for creating emotionally fitted embeddings, and showed that they can perform up to 11 percent better compared to standard embedding models in cosine similarity metrics, and furthermore, can improve performance of emotion detection models.

ACKNOWLEDGEMENTS

I want to thank my Ph.D. advisor Dr. Wlodek Zadrozny, and my dissertation committee members, Dr. Minwoo Lee, Dr. Samira Shaikh, Dr. Xi Niu, and Dr. James Walsh. I would also like to express my gratitude to the graduate school of the University of North Carolina at Charlotte and the Department of Computer Science for their support and for all I have learned during my Ph.D. study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

BERT Bidirectional Encoder Representations from Transformers.

BOW An acronym for Bag Of Words.

CNN An acronym for Convolutional Neural Network.

GloVe An acronym for Global Vectors.

GRU An acronym for Gated Recurrent Unit.

kNN An acronym for k Nearest Neighbors

LSTM An acronym for Long Short Term Memory.

NER An acronym for Name Entity Recognizer.

RNN An acronym for Recurrent Neural Network.

TDA An acronym for Topological Data Analysis.

CHAPTER 1: INTRODUCTION

1.1 Problem Statement and Proposed Approach

In computational linguistics, emotion detection is the process of distinguishing discrete emotions that are present in the text. It can be seen as an evolution of sentiment analysis for finding more fine-grained affective information. However, this field is still young and will need more research before it can match the success and ubiquity of sentiment analysis. With thousands of articles written about its methods and applications, sentiment analysis is a well-established field in natural language processing. It has proven very useful in several applications such as marketing, advertising [2, 3], question answering systems [4, 5, 6], summarization[7], as part of recommendation systems [8] and improving information extraction [9], and many more.

As successful as sentiment analysis has been at classifying text as being positive or negative, it is limited because a far wider range of emotions are present in all text. Being able to identify these more fine-grained emotions could improve many of the applications that sentiment analysis is currently used for and open a path to new applications. For instance, the two emotions *Anger* and *Fear* both express a negative sentiment towards something, but *Anger* is more relevant in assessing responses to a marketing campaign or in sociopolitical monitoring of the public sentiment towards an event or incident. This is because "fearful people tend to have a pessimistic view of the future, while angry people tend to have a more optimistic view" [10]. Moreover, "fear generally is a passive emotion, while anger is more likely to lead to action" [11].

With the capability to detect more precise types of human emotions that emotion

detection makes possible, and its potential applications, there has been a recent surge of research papers in this field. Emotion detection can be used used for understanding emotions in all manner of fields, from political science [12, 13] and psychology [14], to marketing [15] and human-computer interaction [16]. As an example in marketing, emotion detection can be used to analyze consumer reaction to products and services, allowing marketers to better decide which aspect of the product should be changed, which in turn creates a better relationship with customers and increases customer satisfaction [17]. Emotion detection can be used to improve recommender systems and human-computer interactions [18], so that they better reflect and incorporate a user's emotional state [19]. Further, emotion detection systems can be used as a sub-task or input to other NLP systems, like we see in Rangel et al. [20], who created author profiles by analyzing the presence of various emotions in their writings.

With the importance of assessing emotional states when making decisions [21], better monitoring of user reactions can profit any organization that needs to assess the effect of their services, products, or actions on their customer base. This gives them the power to respond to user reactions in a more nuanced and appropriate manner than is possible when classifying emotions as only positive or negative, besides being a benefit to any entity or organization (commercial, political, or governmental institutes, etc.). It is arguable that capturing and analyzing emotional responses may also be imperative for creating better artificial intelligence tools, such as smart assistants or chatbots.

It is clear from the literature that detecting emotions, especially in text, is a hard task. Besides the complexity of human emotions, two other factors contribute to this difficulty. First, identifying emotions in the text is a multi-class classification task which combines various problems we face in machine learning and natural language processing; second, the expression and representation of emotions in text is elusive,

which comes from the vague and context-sensitive nature of the emotional language (e.g., implicit expression of emotions, context-dependency, etc.)

While research into emotion detection has seen a surge in recent years, it still is not a standard tool in natural language processing. Recent work either lacks models elaborate enough to capture the nuances of emotional language, or there is not dataset large enough to train a complex model with a huge number of parameters.

1.2 Psychological Models of Emotion

The prerequisite for talking about extracting emotions is having a general idea about the emotion models and theories in psychology. This body of research provides us with definitions, terminology, models, and theories. Here we introduce the most general and well-accepted theories in a short section to give the reader the basic information needed for the rest of this document.

In psychology, and based on the appraisal theory, emotions are viewed as *states that reflect evaluative judgments (appraisal) of the environment, the self, and other social agents, in light of the organism's goals and beliefs, which motivate and coordinate adaptive behavior* [22]. In psychology, emotions are categorized into basic emotions and complex emotions (i.e., emotions that are hard to classify under a single term such as guilt, pride, shame, etc.). In this document, when we talk about emotions, we mostly mean basic emotions.

Although there is no universally accepted model of emotions, some of the most widely accepted models that have been used in emotion detection literature can be divided based on two viewpoints: emotions as discrete categories, and dimensional models of emotions. According to Discrete Emotion Theory, some emotions are distinguishable based on neural, physiological, behavioral, and expressive features regardless of culture [23]. A well known and most used example is Ekman's six basic

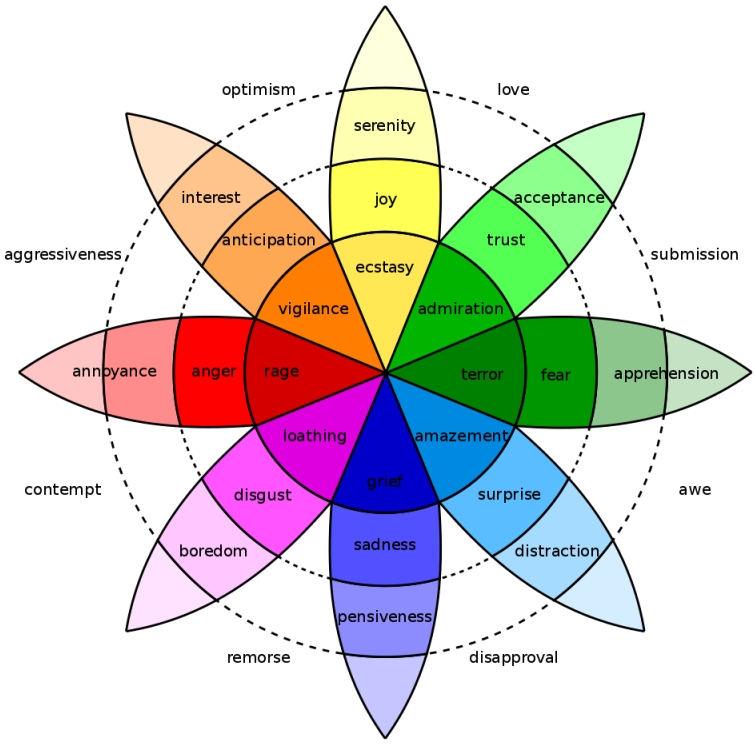

Figure 1.1: Plutchik wheel of emotions.(Source Wikipedia)

emotions [24]. Ekman et al., in a cross-cultural study, found six basic emotions of *sadness, happiness, anger, fear, disgust, and surprise*. Most papers in emotion detection used this model for detecting emotions as a multi-class classification problem, along with some that are based on Plutchik's wheel of emotions [25] in which he categorized eight basic emotions (*joy, trust, fear, surprise, sadness, disgust, anger, and anticipation*) as pairs of opposite emotions (see Figure 1.1). Shaver et al. [1], in his layered categorization of basic emotion, considered six primary emotions of *love. joy, surprise, anger, sadness, and fear* as the basic emotional categories in the first layer, followed by 25 finer emotions in the second layer. And then he assigned more fine-grained emotions in the third layer. This model was also presented in [26](Table 1.1).

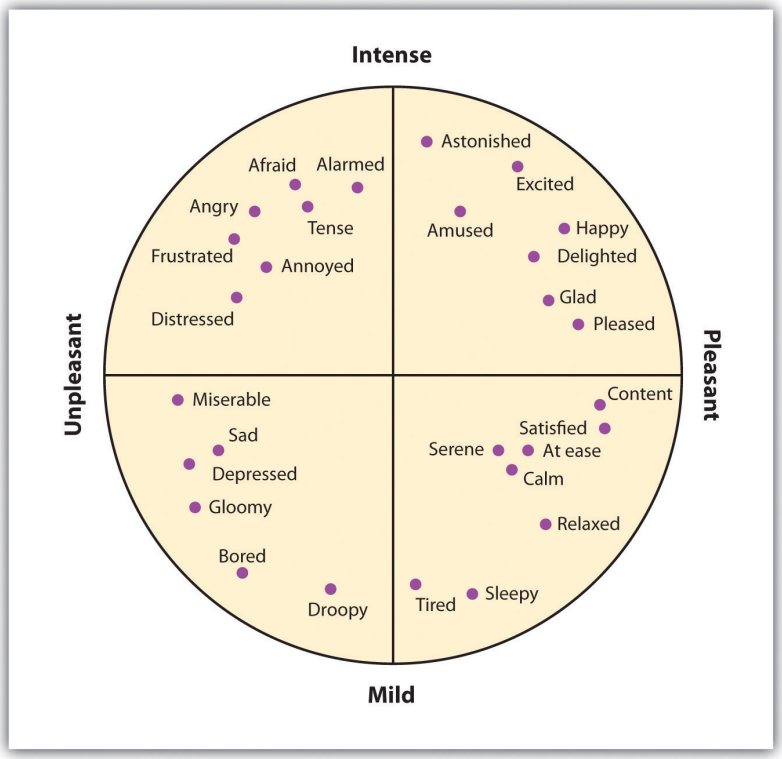

Figure 1.2: Russell's Circumplex model of affect. (Source Wikipedia)

As opposed to discrete emotion models, which consider emotions as to correspond to separate neurological subsystems in the brain, the dimensional models of emotion are based on the hypothesis that all emotions result from a heavily interconnected neurophysiological system. Dimensional model of emotions, see emotions from a different perspective. They define emotions in an n-dimensional space. One such model is the Circumplex model, developed in the early 80s by Russell [27]. This model suggests that "emotions can be shown in a two-dimensional circular space, with one dimension for arousal, i.e., intensity, and one for valance, i.e., pleasantness" (See Figure 1.2). Although these models have been utilized very scarcely in the emotion detection literature, in cases that they were used, they have been shown to be promising as a model to represent emotions [28].

1.3 Complexity of Emotional Expressions

Emotional expression in humans is very context-sensitive and complex, with many nuances. Ben-Ze'ev [29] associates these subtleties to three points: first, "its sensitivity to multiple personal and contextual circumstances"; second, "to the fact that these expressions often consist of a cluster of emotions rather than merely a single one"; and finally, "the confusing linguistic use of emotional terms." In his work, Bazzanella [30] states that the complexity of emotional expressions can be seen in various levels: "the nested interplay with mind/language/behavior/culture, the lexical and semantic problem, the number of correlated physiological and neurological features, their universality or relativity, etc.". As one can even notice in everyday life, it is sometimes tough to distinguish between emotions, especially facing only with a piece of text.

Also, in emotional text, the context is of utmost importance and is crucial in distinguishing emotions [31]. Most recent textual emotion detection studies in NLP are based on the explicit expression of emotion using emotion-bearing words. However, emotion expression is mostly done by expressing the emotion-provoking situation, which can be interpreted in an affective manner [32, 33]. This fact has greatly restricted the ability to identify emotions as a considerable portion of emotional expressions is implicit. Therefore more emphasis should be placed on methodologies that can capture implicit expressions of emotions [34].

There are limited works in the literature that tackle the problem of detecting the implicit emotional expressions, but there have been some attempts in sentiment analysis literature. For instance, Greene et al. [35] utilized syntactic packaging for ideas to analyze the implicit sentiment in text, and was able to improve the state of the art techniques in sentiment analysis. Cambria et al.'s [36] approach were to overcome

the issue by constructing a knowledge-base that merges affective knowledge with Common Sense. Their goal was to move past the reliance on explicit expressions, i.e., verbs, adjectives, and adverbs of emotion. Their reasoning for taking this approach was based on this notion that emotions are mostly expressed through concepts with emotional valence. For instance, 'be laid off' or 'go on a first date' contain emotional wights without having any emotional lexicon.

In a case study about *Anger* Lakoff [37], talks about "conceptual content behind emotions". He emphasized that emotions have complex conceptual structures, and these structures "could be studied by systematic investigation of expression that is understood metaphorically." He stated that expressions of anger are metaphorical in many cases, therefore, could not be considered by their literal meaning (e.g. 'You make my blood boil.' or 'He lost his cool.'). This makes it even more difficult to build a lexical or a conventional machine learning method to detect emotions in text, without first considering the problem of understanding metaphorical expressions.

The importance of context, the complexity of human emotions and emotional expressions, along with frequent use of metaphors and implicit expressions in emotional language, not to mention cross-cultural and intra-cultural variations of emotions, raises the emotion detection problem above a simple multi-class classification problem as has been the default approach in the most research that has been done in the field.

1.4 Thesis summary

In the next chapter, we will begin by introducing some of the most prominent and publicly available data sources, which can be divided into two groups: labeled texts and emotion lexicons. We will also briefly cover vector space models as another potential resource, and review the literature for current supervised and unsupervised

methodologies in emotion detection. In the third chapter, we present an approach for incorporating emotional information of words into these models. This is made possible by adding a second training stage, which uses an emotional lexicon and a psychological model of basic emotions. Next, we show that a neural network we designed based on a bidirectional GRU model with attention mechanism performs better than widely used fine-tuned transformer model (BERT) in capturing emotion from text. In the final chapter, we present a summary of our findings and future directions.

The major contributions of this work to the field of emotion detection can be summarized as follows. First, we designed and built a reliable and generalizable methodology that can capture language complexities inherent in emotional expression. Second, we provided a publicly available dataset, that is large enough to train such a model[1].

Specifically, we show that a bidirectional Gated Recurrent Unit (GRU) model with an attention mechanism can better capture the emotional context and composition of text, and that its performance is significantly improved. We show this by designing a deep neural network model with these elements that creates a better latent representation of the target text as a whole. We also show that by creating a better representation, the model's performance is significantly improved. The improvements in the quality of these latent representations were accomplished by adding two layers specifically designed to do so. The first layer improves the quality of sequential information in the latent representation, and the second layer improves its contextual information quality.

[1] Available at https://github.com/armintabari/Emotion-Detection-RNN/tree/master/data

Table 1.1: Three layered emotion classification. (Shaver et al. 1987)

Primary	**Secondary**	**Tertiary**
Liking	Affection	Adoration · Fondness · Liking · Attractiveness · Caring · Tenderness · Compassion · Sentimentality
	Lust/Sexual desire	Desire · Passion · Infatuation
	Longing	Longing
Joy	Cheerfulness	Amusement · Bliss · Gaiety · Glee · Jolliness · Joviality · Joy · De-light · Enjoyment · Gladness · Happiness · Jubilation · Elation · Satisfaction · Ecstasy · Euphoria
	Zest	Enthusiasm · Zeal · Excitement · Thrill · Exhilaration
	Contentment	Pleasure
	Pride	Triumph
	Optimism	Eagerness · Hope
	Enthrallment	Enthrallment · Rapture
	Relief	Relief
Surprise	Surprise	Amazement · Astonishment
Anger	Irritability	Aggravation · Agitation · Annoyance · Grouchy · Grumpy · Crosspatch
	Exasperation	Frustration
	Rage	Anger · Outrage · Fury · Wrath · Hostility · Ferocity · Bitter · Hatred · Scorn · Spite · Vengefulness · Dislike · Resentment
	Disgust	Revulsion · Contempt · Loathing
	Envy	Jealousy
	Torment	Torment
Sadness	Suffering	Agony · Anguish · Hurt
	Sadness	Depression · Despair · Gloom · Glumness · Unhappy · Grief · Sor-row · Woe · Misery · Melancholy
	Disappointment	Dismay · Displeasure
	Shame	Guilt · Regret · Remorse
	Neglect	Alienation · Defeatism · Dejection · Embarrassment · Homesickness · Humiliation · Insecurity · Insult · Isolation · Loneliness · Rejection
	Sympathy	Pity · Mono no aware · Sympathy
Fear	Horror	Alarm · Shock · Fear · Fright · Horror · Terror · Panic · Hysteria · Mortification
	Nervousness	Anxiety · Suspense · Uneasiness · Apprehension (fear) · Worry · Distress · Dread

CHAPTER 2: LITERATURE REVIEW

As opposed to sentiment analysis, textual datasets annotated with markers of emotional content are scarce. Any new methodology for emotion detection in text, based on conventional supervised classifiers or neural networks, requires a vast amount of annotated data for training and development. However, as a relatively new field in natural language processing, emotion detection as a multi-class classification problem faces a lack of available annotated data. In this chapter, we first introduce some of the most prominent and publicly available resources. These resources can be separated into two groups: labeled texts and emotion lexicons. We also briefly cover vector space models as another potential resource. Later, we review current supervised and unsupervised methodologies in emotion detection.

2.1 Labeled Text

Having a standard, free, and generalized annotated data makes it easier to train and test any new method, and is an important factor in any classification task. One of the most prominent and well-known sources for emotionally labeled text is the Swiss Center for Affective Sciences [38]. The most used resource they provide is ISEAR, International Survey On Emotion Antecedents And Reactions. It consists of responses from about 3000 people around the world who were asked to report situations in which they experienced each of the seven major emotions (joy, fear, anger, sadness, disgust, shame, and guilt) and how they reacted to them. The result was a promising dataset to be used to test many methods for emotion extraction and classification. This dataset consists of about 7600 records of emotion-provoking text.

SCAS has many more resources that can be useful, especially in languages other than English.

EmotiNet knowledge base [39] tackled the emotion detection problem from another perspective. Balahur et al. showed that attempts to detect emotion based on Bag of Words models would lead to a low-performance methodology because "expressions of emotions are most of the time not presented in text in specific words" rather from the "interpretation of the situation presented" in the text. Their insight was based on *Appraisal Theory* in psychology [40] . The authors create a new knowledge base containing action chains and their corresponding emotional label. They started from around a thousand samples from the ISEAR database and clustered the examples within each emotion category based on language similarity. Then they extracted the agent, the verb, and the object from a selected subset of examples. Furthermore, they expanded the ontology created using VerbOcean [41] to increase the number of actions in their knowledge base and reduce the degree of dependency between the initial set of examples and the resources. Although their methodology showed promise, especially their attempt at concept extraction from text, it could not present itself as a viable and generalizable approach in its current form, because of the small size of the knowledge-base and the limited structure of information they used (the four-tuple of actor, action, object, and emotion).

On the other hand, Vu et al. [42] focused on the discovery and aggregation of emotion-provoking events. The authors created a dataset of emotional events by surveying 30 subjects and utilized that to aggregate events similar to those from the web by applying bootstrapping algorithms and Espresso pattern expansion [43]. One dataset that has been used frequently is the SemEval-2007 [44], which consists of 1250 news headlines extracted from news websites and annotated with six Ekman's emotions. The other example is Alm's annotated fairy tale dataset [45], consisting of

Table 2.1: Available emotion annotated datasets. [V]:valence, [A]: arousal, [D]: dominance [No]: no emotions, [Mi]: mixed emotions, [CF]: enthusiasm, fun, hate, neutral, love, boredom, relief, empty, [D]: disgust, [LT]: love, thankfulness, [DS]: disgust, surprise.

Datset	Topic	Annotation	Size	Source
AffectiveText	News (headlines)	Ekman + V	1,250	Strapparava (2007)[47]
SemEval	Conversational	Happy,Sad,Anger	30160	Chatterjee (2019)[48]
Blogs	Blogs (sentences)	Ekman + No + Mi	5,025	Aman (2007)[49]
CrowdFlower	General (tweets)	Ekman + CF	40,000	CrowdFlower (2016)[1]
EmotionalTweets	General (tweets)	Ekman - D + LT	2,488,982	Wang et al. (2012)[50]
DailyDialogs	Multiple (dialogues)	Ekman	13,118	Li et al. (2017)[51]
Electoral-Tweets	Elections (tweets)	Plutchik	4,058	Mohammad (2015)[52]
EmoBank	Multiple (sentences)	V,A,D	10,548	Buechel (2017)[53]
EmoInt	General (tweets)	Ekman - DS	7,097	Mohammad (2017)[54]
Emotion-Stimulus	General (sentences)	Ekman + shame	2,414	Ghazi et al. (2015)[55]
fb-VA	Questionnaire (posts)	V,A	2,895	Preotiuc (2016)[56]
Grounded-Emotions	Weather/event (sentences)	Happy,Sad	2,585	Liu et al. (2017)[57]
ISEAR	Event (descriptions)	Ekman + shame+ Guilt	7,665	Scherer (1994)[58]
Tales	Fairytales (sentences)	Ekman	15,302	Alm et al. (2005) [45]
SSEC	General (tweets)	Plutchik	4,868	Schuff et al. (2017)[59]
TEC	General (tweets)	Ekman + ± Surprise	21,051	Mohammad (2012)[60]

1580 sentences from children fairy tales, also annotated with six of Ekman's emotions. These datasets have been mostly used as a benchmark in the literature. As emotion detection gets more attention, there will be a need for more datasets that could be used in different tests of models and methods for emotion detection. A list of all available datasets can be seen in Table 2.1 [46].

The lack of standard benchmark datasets with proper generalized language and accepted annotations, especially ones which are large enough to train more modern machine learning models, pushes the community of researchers to utilize text from micro-blogs, like Twitter, in which self-expression using methods like *hashtags, and emoticons* are used as labels. An attempt to create an emotionally labeled dataset was presented by Wang et al. [50], which consisted of 2.5 million tweets annotated with hash-tags presented at the end of each tweet.

2.2 Emotional Lexicons

Although having access to expressive emotional text such as ISEAR is of utmost importance, especially for comparison among various emotion detection methodolo-

gies, there are many instances in which having access to an annotated emotional lexicon could be useful, especially for cases in which a more word-based analysis is required. And although considerable datasets are available on words sentiment polarity going back a few years [61], the lack of a large emotional lexicon led Mohammad et al. [62] to create an emotion word lexicon in 2010. In that paper, and later on, in [63] the authors utilized Amazon Mechanical Turk to annotate about 14000 English words along with lexicons in various other languages. (These are available on their website[2]).

Another popular lexicon for emotions that has been used a lot in literature is WordNet-Affect. Strapparava et al. [64] started from WordNet [65] (a well known lexical database) as their base and created a lexical representation of affective knowledge. The authors used the selection and tagging of a subset of synsets, which represented affective concepts, to introduce "affective domain labels" to WordNet's hierarchical structure. WordNet-Affect, despite its small size (containing 2874 synsets and 4787 words), was a great attempt to extract emotional relations of words from WordNet and was used in many early applications of sentiment analysis, opinion mining [66], and detecting emotions, specifically in order to extend affective word sets from the basic set of emotions.

DepecheMood, created by Staiano et al. [67], is another attempt to curate an emotional lexicon. In their work, the authors utilized crowd-sourcing to annotate thirty-five thousand English words. They showed that lexicons could be used in several approaches in sentiment analysis, as features for classification in machine learning methods [68], or to generate an affect score for each sentence, based on the scores of the words which are higher in the parse tree [69]. Other emotional

[2]NRC word-emotion association lexicon: http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm

lexicons frequently used in the literature are LIWC lexicon [70] consisting of 6400 English words annotated for various emotions, and also ANEW (Affective Norm for English Words) developed by Bradley et al. [71]. This dataset consists of nearly two thousand words annotated for emotions based on dimensional models of emotions, for three dimensions of arousal, valance, and dominance.

2.3 Word Embedding

Word embeddings is a technique based on distributional semantic modeling. It is rooted in the idea that words that frequently co-occur in a relatively large corpus are similar in some semantic criteria. In these methods, each word is represented as a vector in an n-dimensional space, called the vector space, and in a way, the distance between vectors corresponds to the semantic similarity of the words they represent. These vector space models have been shown to be useful in many natural language processing tasks, such as named entity recognition [72], machine translation [73], and parsing [74]. Many embedding models have been created in the past decade or so, all of which show similar performances, as demonstrated by Levy et al. [75]. Some of most frequently used and well-established vector space models in the literature are latent semantic analysis or LSA (an older methodology based on matrix factorization), Word2Vec [76, 77], GloVe [78], and ConceptNet [79]. It has been shown that these models, just by utilizing the statistical information of word co-occurrences, can incorporate a variety of information about words [78] such as closeness in meaning, gender, types, capital of countries, etc., and in the arithmetic of word vectors shown in such overused examples as $v(king) - v(queen) \approx v(man) - v(woman)$.

There also have been many attempts to increase their performance, and incorporate more information in these models such as retrofitting [80], and counter-fitting [81] external word ontologies or lexicons [79, 82]. Some work is available in the lit-

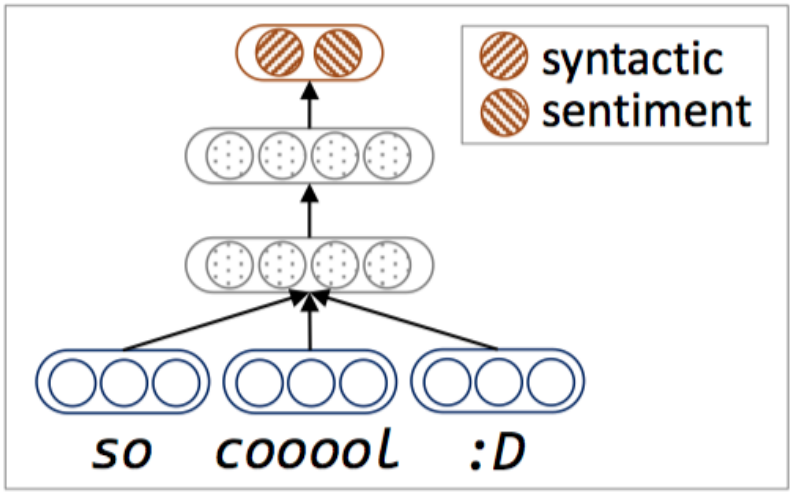

Figure 2.1: Neural network design used for sentiment specific word embedding (Tang et al. 2014)

erature on creating word embeddings for sentiment analysis. For instance, Tang et al. [83] who created a sentiment-specific word embeddings using neural networks, to classify tweets based on sentiments [84] Figure 2.1. Such approaches, designed specifically for creating emotional word embeddings from scratch, or incorporating emotional information into pre-trained word vectors after the fact, might lead to better performances in emotion detection tasks, either in unsupervised methods or as features for classification tasks using conventional machine learning, or deep learning [69].

2.4 Supervised Approaches

Due to the lack of large datasets labeled for emotion, many supervised approaches for detecting emotions have been performed on data gathered from microblogs (e.g., Twitter), using emoticons or hashtags as the emotion labels for the data, with the assumption that hashtags and emoticons show the writer's emotional state. For instance, such an attempt can be seen in Suttles et al. [85], in which the authors used the four pairs of opposite emotions in the Plutchik's wheel to build four binary clas-

sification models. They used hashtags, emoticons, and emoji as labels for their text and could reach between 75% to 91% accuracy on a secondary hand-labeled dataset.

Purver et al. [86] also used Twitter data and an SVM classifier to reached 82% accuracy for classifying the emotion of *Happiness* in 10-fold cross-validation, and 67% accuracy in classifying over the whole dataset for the same emotion. The authors used emoticons as labels for their training set, and hashtags as labels on the test dataset. They then tested their trained models for each emotion to check if they can distinguish emotion classes from each other rather than just discriminating one class from a general class of *Other*. The results varied from as low as 13% up to 76% accuracy for various emotions. They also created a human-annotated dataset of 1000 tweets and used it to evaluate the quality of using emoticons and hashtags as labels for the data. The reported F-score for various emotions varied from 0.10 up to 0.77. This study showed that the conventional classifiers performed well on emotions like happiness, sadness, and anger, but not well for less clear-cut emotions. They concluded that utilizing emoticons and hashtags as labels is a promising strategy and can be used as an alternative to manual human labeling.

Saif Mohammad [60] also utilized hashtags as labels for tweets and trained support vector machines as binary classifiers for the emotions in Ekman's model of basic emotions. He first showed that hashtags as labels perform at least better than random classification. He utilized Daumé's domain adaptation method [87] to assess the classification power of his data in a new domain. Roberts et al. [88] recognized 14 topics that "would frequently evoke emotion" and collected tweets representing those topics and created a dataset in which seven emotions (Ekman + Love) were represented. The authors then used seven SVM binary classifiers to classify emotions in the dataset, with an average of 0.66 in F1-score. The design can be seen in Figure 2.2

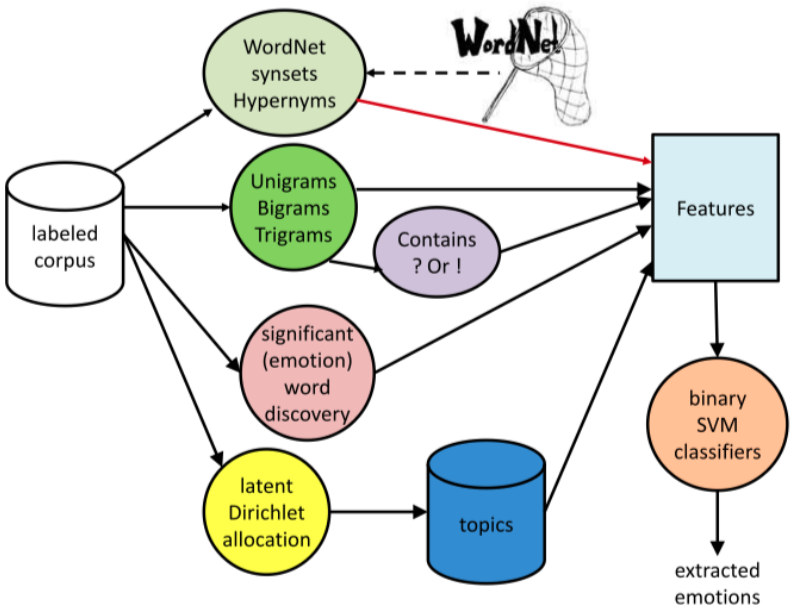

Figure 2.2: Roberts et al. 2012, classifier design for detecting emotions.

Hasan et al. [89] also utilized hashtags as the labels for their data and generated their features with the unigram model. The authors removed any word from the tweets which were not present in their emotion lexicon (created based on 28 basic emotion words presented in the Circumplex model and then extended using WordNet synsets). They used four classifiers (SVM, Decision Tree, Naive Bayes, and KNN), achieving accuracy close to 90% for four main emotion category classes in the Circumplex model. In another paper[90], they created an automatic emotion classification system to detect emotions in streams of tweets. This approach consisted of two tasks: training an offline emotion classifier based on the model presented in the 2014 paper, and a two-step classification method that first identified tweets containing emotions and then classified these tweets using soft classification techniques into more fine-grained labels.

Faced with the lack of emotionally labeled text, Wang et al. [50] gathered a large dataset (around 2.5 million tweets) using emotional hashtags and compared two ma-

Table 2.2: Results of Wang et al., for different emotions.

Emotion	Precision(%)	Recall(%)	F-measure(%)
joy (28.5%)	67.6	77.3	72.1
sadness (24.6%)	62.6	66.8	64.7
anger (23.0%)	69.8	73.3	71.5
love (12.1%)	58.1	46.2	51.5
fear (5.6%)	59.7	34.7	43.9
thankfulness (5.3%)	66.6	50.0	57.1
surprise (1.0%)	44.7	8.2	13.9

chine learning models for emotion identification. The authors used Shaver et al.'s model [1] to map hashtags to emotions. They extended the number of hashtag words to 131 in total for the seven basic emotions. They then tried to increase the quality of the dataset by keeping more informative tweets (i.e., tweets with the emotional hashtags at the end, ones with at least 5 word, and ones which contain no quotations or URLs, in English, and have three hashtags or less), and tried different combinations of features (e.g., different n-grams, the position of n-grams, multiple lexica, POS) with 250k of the training data to find the best set of features, with the best result for the combination of n-gram(n=1,2), LIWC lexicon, MPQA lexicon, WordNet-Affect, and POS. After finding the best feature set, they increased the size of training data from 1000 tweets to full training set to see the effect of training size in the classification (Figure 2.3). The final classifier reached the F-Measure as high as 0.72 for joy, and as low as 0.13 for surprise (See Table 2.2). They justified the varying result for different emotions because the training dataset had an unbalanced distribution. Also, based on the confusion matrix, they reported that a high number of misclassified tweets between class pairs like anger and sadness, or joy and love, were because these emotions are "naturally related," and "different people might have different emotions when facing similar events."

In another emotion classification task on tweets done by Balabantaray et al. [91],

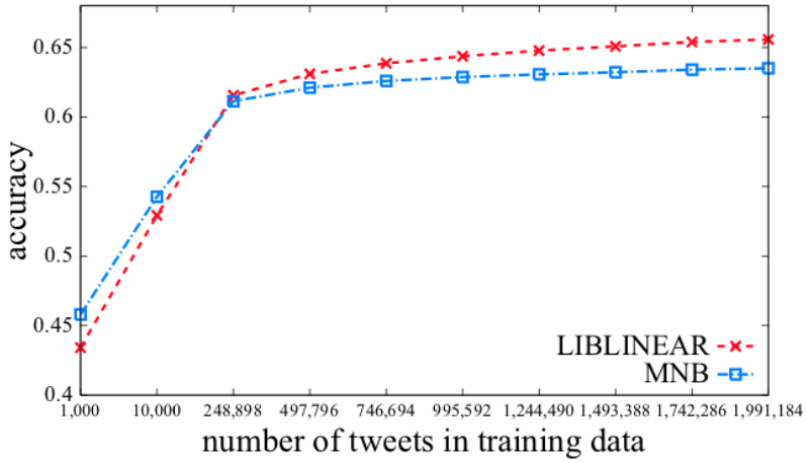

Figure 2.3: Accuracies of LIBNEAR and Multinomial NB with varied sizes of training data. (Wang et al.)

the authors labeled around 8000 tweets manually for Ekman's six basic emotions. They used a SVM multi-class classifier and extracted 11 features: *Unigrams, Bigrams, Dependency-Parsing, and Emoticons, Word-net Affect lexicon, Word-net Affect lexicon with left/right context, Word-net Affect emotion POS, POS, POS-bigrams, Personal-pronouns, Adjectives,* which resulted in an accuracy of 73.24%.

We can see a combination of two methods for detecting emotion in work by Wen et al. [92]. In their paper, the authors utilized a combination of machine learning (SVM) and lexicon-based methods to generate two emotional labels for each microblog text. They then use CRS mining (Class Sequential Rules)[93] to generate sequences for each post based on the two labeling for each sentence in the text and the conjunctions between them. You can see an example in Figures 2.4 and 2.5. Using the resulting dataset and adding extra features of lexicon counts and punctuation and using an SVM model, they could reach 0.44 in F-measure, which showed significant improvement over other methods based solely on emotion lexicons or simple SVM.

Li et al. [94] proposed an "emotion cause detection technique" to extract features that are "meaningful" to emotions instead of choosing words with a high co-occurrence

	Sentence	Emotion
1	今天下雨。 (Today is rainy.)	*none*
2	我有点郁闷 [流泪]！ (I am a little depressed [tears]!)	*sadness*
3	但是在家里看书也不错 [嘻嘻]。 (But staying at home to read some books is also not so bad [hee hee].)	*happiness*

Figure 2.4: CSR sample microblog post. (Wen et al. 2014)

(<{*none*, *sadness*}{*sadness*}{但是(*but*)}{*happiness*}>, *happiness*)

Figure 2.5: Sample sequence generated. (Wen et al. 2014)

degree. Their method is based on Lee et al.'s work on rule-based emotion cause detection [95]. After utilizing predefined linguistic patterns to extract causes for emotions and adding it as extra features, they used Support Vector Regression (SVR) to build the classifier and reach a higher F-score than previous works for some emotions such as anger, happiness, and disgust. Overall, their approach had higher precision but low recall.

In their work, Li et al. [96] tried sentence level emotion classification instead of document level. The authors recognized the two biggest problems in sentence-level emotion detection: firstly, it is a multi-label classification, understanding that each sentence could have more than one emotional label, and secondly, at the sentence level, the short length of the text provides less content for feature extraction. Considering these two challenges, they built a Dependence Factor Graph (DFG) based on two pieces of information, *label dependence*, i.e., multiple labels assigned to one sentence would be correlated to each other, like Love and Joy compared to Joy and Hate, and *context dependence*, i.e., two consecutive sentences, or sentences presented in the same document (paragraph) might share one or more emotion labels. After training, using the DFG model, they reached 63.4% accuracy with an F1-measure of 0.37, showing a significant improvement over previous methods [97, 98].

In a work done by Seyeditabari et al. [99], the authors classified social media comments posted about a social crisis event for the emotion of anger. They used a short

survey, in which the participants were asked to "comment under a news headline as though they are commenting on social media", and gathered 1192 response. They used these 1192 data points as their training dataset. They then used logistic regression coefficients for feature selection (words) and random forest as the main classifier, and could reached 90% accuracy in detecting anger in a dataset created by the same survey from a different population.

The current state of the art methodologies for emotion detection are mostly based on conventional supervised approaches. Small and imbalanced datasets, especially for emotion classification as a multi-class problem, are obstacles to overcome for supervised learning. This imbalance leads to an increasing number of misclassifications for underrepresented classes [100, 101, 50]. There are various approaches proposed in the literature [100] to overcome this issue. There are three ways to tackle this problem; First, by manipulating the learning algorithm so that it could adapt to this imbalance [102], second by adding some cost to the class with the majority during training [103], or third, by sampling from the training dataset to make the classes balanced [104]. Xu et al. [105] proposed an over-sampling methodology that is based on word vector spaces [77] and recursive neural tensor network [69] which showed a substantial improvement over previous sampling methods, especially for emotion detection as a multi-class multi-label task.

The principal question is if building emotion detection models based on conventional machine learning methods can move past the mediocre results presented throughout the literature. To emphasize the value of a deeper analysis, compared to what conventional machine learning methods can accomplish, we can point to the result of the comparative analysis done by Balahur et al. [106]. The authors assessed multiple feature sets and compared them to EmotiNet. They concluded that the task of emotion classification could be best tackled using methodologies based on common-

sense knowledge. Their results showed that EmotiNet, even with the small size of its knowledge base, could produce a comparative outcome to conventional supervised learning methodologies with a large training dataset.

2.5 Unsupervised Approaches

Kim et al. [107] used a lexicon-based unsupervised method to detect emotions in text. The authors used both categorical (sadness, anger, joy, and fear) and dimensional models of emotions. They utilized three datasets: ISEAR, children's fairy tales, and SemEval-2007 "Affective Text." For the categorical emotion model, they adopt WordNet-Affect as their lexicon and evaluated three dimensionality reduction methods: Non-negative Matrix Factorization (NMF), Latent Semantic Analysis (LSA), and Probabilistic Latent Semantic Analysis (PLSA). Moreover, for the dimensional emotion models, they utilized ANEW (Affective Norm for English Words) and used WordNet-Affect to extend the ANEW lexicon. They assigned the target text's emotional label based on its vector's similarity (cosine distance) to the corresponding vectors for each emotion category or emotional dimension. Their results showed that NMF-based categorical classification outperformed other categorical approaches, and the dimensional model could produce the second-best results with F-measure as high as 0.73.

Another approach based on unsupervised learning for emotion detection is presented in the paper by Agrawal et al. [108]. The authors start by extracting what they call NAVA terms (i.e., Nouns, Adjectives, Verbs, and Adverbs) from the sentences. They then extract syntactic dependencies among extracted terms in each sentence to represent the contextual information in their model. They then used semantic closeness to generate emotion vectors for words based on the assumption that the NAVA words (affect words), which co-occur in the same context more often,

tend to be more related semantically. They utilized Point-wise Mutual Information (PMI) as the measure of semantic relatedness of two terms (Equation 2.1) and generated a vector for each term using the PMI of the term with all terms related to each emotion. They then adjust the generated vectors by considering the contextual information presented in syntactic dependency of terms. After generating vectors for each term, they compute a vector for each sentence by averaging all the NAVA terms' emotional vectors. They evaluated their method on multiple data sources and showed that compared to other unsupervised approaches, this method performed more accurately and even performed on par with some supervised methods.

$$PMI(x, y) = \frac{cooccurrence(x, y)}{occurrence(x) * occurrence(y)} \tag{2.1}$$

Mohammad, in another lexicon base approach, [109] showed how emotion detection in text can be utilized to organize a collection of documents for emotion-based search, and how books portray various characters through co-occurring affect terms by analyzing books and emails. The author utilized the NRC lexicon to analyze which of the emotion terms occur in the available text by counting, and calculated ratios such as the number of terms associated with a particular emotion as opposed to another emotion, to see if a document has a more pronounced expressed emotions compared to others in the same corpus. He gathered three datasets for emotional emails: *hate mails, love letters, and suicide notes*. Figure 2.6 shows the various emotions which are present in these categories. The higher presence of emotions such as Trust and Joy compared to others can be seen in these diagrams. Furthermore, he analyzed and compared various emotions in different corpora, such as emails written by men to women vs. men, workplace emails, and emails written by women/men. Using the same lexical approach, he also performed some fascinating analysis of books and

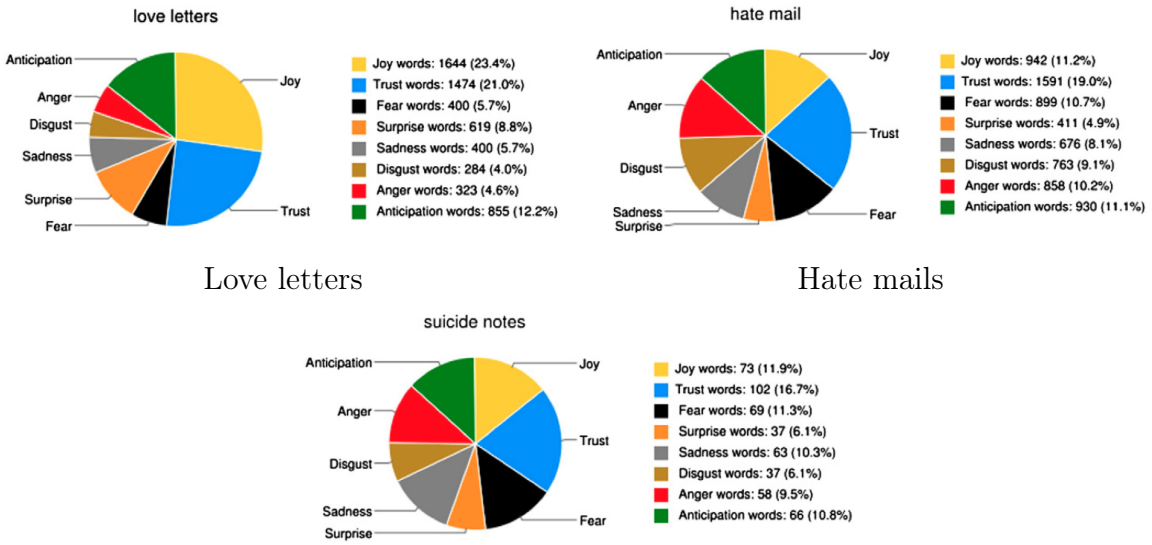

Figure 2.6: Percentage of emotions in various text (Mohammad 2012).

works of literature.

In another example of unsupervised approaches, Rey-Villamizar et al. [110] used a lexicon-based method to recognize language patterns associated with anxiety in online health-oriented forums. The authors define user behavioral dimension (BD) based on the anxious word list available in the LIWC lexicon. They define BD for each user as a measure of the average fraction of terms from the anxious list BD_i across all posts from that user, Equation 2.2:

$$BD_i(u) = log\left(\frac{1}{|posts(u)|}\sum_{p \in posts(u)}\frac{|words_{BD_i}(p)|}{|words(p)|}\right) \tag{2.2}$$

Then they tracked these values for each user (or groups of users) through time, and the correlation of this BD values with other behavioral dimensions, and showed that the anxiety level in patients participating in a support group lowers over time. Tromp et al. [111] introduced the RBEM-Emo method for emotion detection in text as a rule-based approach and an extension of their previous study in polarity detection [112]

called Rule-Based Emission Model. They showed that rule-based emotion detection techniques could perform on par with the current state of the art conventional machine learning methods, such as recursive auto-encoder and SVM classifier.

Bandhakavi et al. [113] utilized a domain-specific lexicon that they previously created based on unigram mixture models [114, 115] to extract features from text and showed that their lexicon performs better than other methods such as supervised Latent Dirichlet Allocation and Point-wise Mutual Information.

2.6 Conclusion

Going through the literature, we can see the hard task of detecting expressed emotions. The difficulties can be attributed to many problems from complex nature of emotion expression in text, to inefficiency of current emotion detection models, and lack of high quality data to be utilized by those models.

On one hand, complex nature of emotion in human, and on the other hand, the intricacy of emotional language, resulting from the context dependency of emotion expression, and implicit nature of such expressions, makes emotion detection a hard task. In order to address this issues, it is important to pay attention to the complexity of emotional language when building emotion detection systems. These systems should be designed based on the linguistic complexities of emotion expression to be able to grasp the implicit expression of emotions. It is also crucial to consider the contextual information in which the expression is occurring.

In addition, in any machine learning task, the quality and quantity of data has a huge effect on the performance of classification algorithms. Although huge amount of textual data is currently available, for any supervised model, a large amount of annotated data is required. A great body of work has already been dedicated to overcome this problem by using self annotated microblog data, but it has not yet

possesses qualities which are required for an applicable system. Additionally, the niche nature of the language used in microblog text, prevents the systems trained on these texts to be used to classify other types of text (e.g. tweets vs. news comments). Therefore, any attempt to create a large, balanced dataset, with high quality labels could provide a brighter future for the field.

Furthermore, creating a multi-class classification methodology based on the nature of the data and the task at hand, is another front that could be considered to increase the performance of emotion detection systems. Some suggestions that were less present in the literature, are to develop methods that go above BOW representations and consider the flow and composition of language. As we will see in this work, recurrent neural networks, along with attention mechanism, perform very well by capturing sequential nature of the text and preserving contextual information. or their future decision making processes.

CHAPTER 3: METHODOLOGY

3.1 Creating Emotional Word Embeddings

Word embeddings are one of the most useful tools in any modern natural language processing expert's toolkit. They contain various types of information such as semantic relatedness and similarity about each word. Although they are one of the best ways to represent words in many NLP tasks, there are some types of information that cannot be learned by these models. Emotional information of words is one of those. In this paper, we present an approach to incorporate emotional information of words into these models. We accomplish this by adding a second training stage, which uses an emotional lexicon and a psychological model of basic emotions. We show that fitting an emotional model into pre-trained word vectors can increase the performance of these models and emotion classification task using neural networks. We measure performance of these models by the cosine similarity of emotions in the same category and in opposite categories. This is the first such model presented in the literature, and they can open the way to increase performance in a variety of emotion detection techniques.

3.1.1 Introduction

There is an abundant volume of textual data available online about a variety of subjects through social media. This availability of a large amount of data led to fast growth in information extraction using natural language processing. One of the most important types of information that can be captured is the emotional reaction of the population to a specific event, product, etc. We have seen a vast improvement in

extracting the sentiment from text to the point that sentiment analysis has become one of the standard tools in any NLP expert's toolkit and has been used in various applications [116].

As discussed in section 1.1, emotion detection, as a more fine-grained affective information extraction technique, is just recently making a larger appearance in the literature. The amount of useful information that can be gained by moving past the negative and positive sentiments and towards identifying discrete emotions can help improve many applications. For example, the two emotions *Fear* and *Anger* both express a person's negative sentiment. However, it has been shown that fearful people tend to have a pessimistic view of the future, while angry people tend to have a more optimistic view [10]. Moreover, fear generally is a passive emotion, while anger is more likely to lead to action [11]. The usefulness of understanding emotions in political science [12], psychology, marketing [15], human-computer interaction [16], and many more, gave the field of emotion detection in natural language processing life of its own, resulting in a surge of research papers in recent years.

Word embeddings, as one of the best methods to create representation for each word in the corpus, are mostly used as features in any neural network-based classifiers. These word vectors are created so that various types of information are coded in the shape of the vector space and the angular distance between them. For example, the distance between the two words *cat* and *feline* should be less than the distance between *cat* and *book*, as cat is semantically closer to feline.

You can find a verity of similarity or categorical information in the shape of these vector spaces that make them one of the best tools we have in natural language processing. However, due to the nature of their training methods, these embeddings do not contain the emotional similarity information[117]. In this work, we present and analyze a methodology to incorporate emotional information into these models

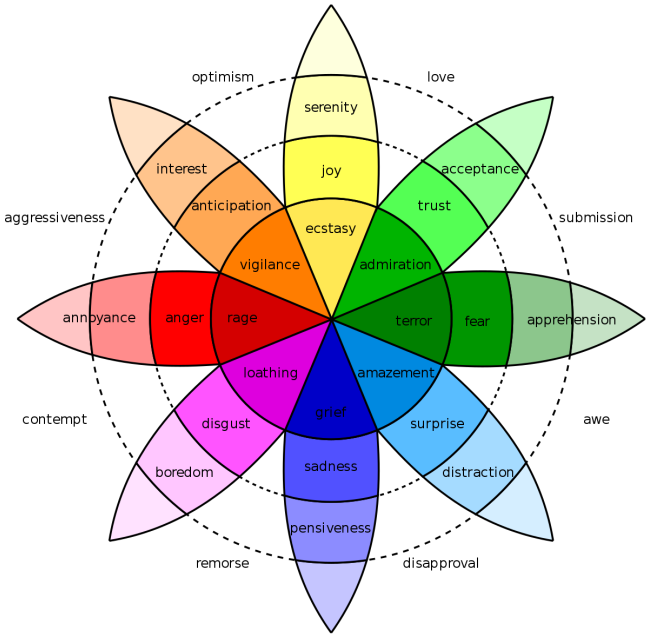

Figure 3.1: Plutchik's Wheel of Emotions. Opposite emotions are placed on opposite petals. (Source Wikipedia)

after the fact. We accomplish this by utilizing an emotion model and an emotion lexicon, in this case, Plutchik's wheel of emotions [118], and NRC emotion lexicon [119]. We have also used a secondary emotion model to create an emotional similarity metric to compare the models' performance before and after training.

This result is an important step to show the potential that these models can improve emotion detection systems in different ways. Emotion sensitive embeddings can be used in various emotion detection methodologies, such as recurrent neural network classifiers, to improve the model performance in learning, and classifying emotions. It can also be used in attention networks [120] to calculate feature weights for each term in the corpus to potentially improve the classification accuracy by giving more weights to the emotionally charged terms.

3.1.2 Related Works

In the past decade, word embeddings have been one of the most useful tools in natural language processing, especially by increasing neural network usage. Word2Vec, created by Mikolov et al., and presented in two papers [77, 76] has shown that these vectors could perform reliably in a variety of tasks. GloVe [78] took a different approach for creating word embeddings, which performed on par with Word2Vec.

After these models' success, many studies have been done to figure out the shortcomings of these models and try to make them better. Speer et al. used an ensemble method to integrate Word2Vec and GloVe with ConceptNet knowledge base [79] and created ConceptNet NumberBatch model [82] and showed that their model outperforms either of those models in a verity of tasks. [80] also presented a method to refine these vectors base on an external semantic lexicon by encouraging vectors for similar words to move closer to each other.

Understanding that these embedding models do not perform well for semantically opposite words, Nikola et al. in their paper [81], created a methodology that not only brought the vectors for similar words close to each other but also moved vectors for opposite words farther apart. Mikolov et al. [121] created an improved model fastText in which they used a combination of known tricks to make the vectors perform better in different tasks. Nevertheless, as shown in [117], these models do not perform well in emotional similarity tasks.

There have also been various attempts to create sentiment embeddings that would perform better in sentiment analysis tasks than standard vector spaces [83, 122, 123]. Moving past sentiments, in this paper, we present a method to incorporate emotional information of words into some of these models mentioned above.

Table 3.1: Three layered emotion classification [1].

Primary Emotion	Secondary Emotion	Tertiary Emotion
Liking	Affection	Adoration · Fondness · Liking · Attractiveness · Caring · Tenderness · Compassion · Sentimentality
	Lust/Sexual desire	Desire · Passion · Infatuation
	Longing	Longing
Joy	Cheerfulness	Amusement · Bliss · Gaiety · Glee · Jolliness · Joviality · Joy · De-light · Enjoyment · Gladness · Happiness · Jubilation · Elation · Satisfaction · Ecstasy · Euphoria
	Zest	Enthusiasm · Zeal · Excitement · Thrill · Exhilaration
	Contentment	Pleasure
	Pride	Triumph
	Optimism	Eagerness · Hope
	Enthrallment	Enthrallment · Rapture
	Relief	Relief
Surprise	Surprise	Amazement · Astonishment
Anger	Irritability	Aggravation · Agitation · Annoyance · Grouchy · Grumpy · Crosspatch
	Exasperation	Frustration
	Rage	Anger · Outrage · Fury · Wrath · Hostility · Ferocity · Bitter · Hatred · Scorn · Spite · Vengefulness · Dislike · Resentment
	Disgust	Revulsion · Contempt · Loathing
	Envy	Jealousy
	Torment	Torment
Sadness	Suffering	Agony · Anguish · Hurt
	Sadness	Depression · Despair · Gloom · Glumness · Unhappy · Grief · Sor-row · Woe · Misery · Melancholy
	Disappointment	Dismay · Displeasure
	Shame	Guilt · Regret · Remorse
	Neglect	Alienation · Defeatism · Dejection · Embarrassment · Homesickness · Humiliation · Insecurity · Insult · Isolation · Loneliness · Rejection
	Sympathy	Pity · Mono no aware · Sympathy
Fear	Horror	Alarm · Shock · Fear · Fright · Horror · Terror · Panic · Hysteria · Mortification
	Nervousness	Anxiety · Suspense · Uneasiness · Apprehension (fear) · Worry · Distress · Dread

3.1.3 Fitting Emotional Constraints in Word Vectors

We use a methodology similar to what [81] used to incorporate additional linguistic constraints in word vector spaces for fitting emotional information into pre-trained word vectors. Our goal here, is to change the vector space $V = \{v_1, v_2, \ldots, v_n\}$ to $V' = \{v'_1, v'_2, \ldots, v'_n\}$ in a careful manner to add emotional constraints to the vector space without losing too much information already present during the original learning step. To perform this task, we create two sets of constraints based on the NRC emotion lexicon, one for words which have a positive relation to an emotion such as (abduction, sadness), and one to keep track of each word related to the opposite of that emotion (abduction, joy), joy being the opposite of sadness. In the NRC lexicon, Mohammad et al. annotated over 14k English words for eight emotions from Plutchik's model of basic emotions(See Figure 3.1).

To create our two constraint sets, we extract all (word,emotion) relations indicated in the lexicon so that in our first set, we have ordered pairs, each indicating a word

and the emotion it is associated with.

$S = \{(w_1, e_1), (w_1, e_3), (w_2, e_2), \ldots\}$

And for each emotion e_i, we add its opposite e'_i to our second set in which e'_i is the opposite emotion to e_i based on Plutchik's model.

$O = \{(w_1, e'_1), (w_1, e'_3), (w_2, e'_2), \ldots\}$

For example, in the NRC lexicon, the word *abandon* is related to the emotion *sadness*. In this case we add the pair (abandon, sadness) to our positive relation set S and also add the order pair (abandon, joy) to our negative relation set O. We have extracted over 8k such pairs of (word, emotion) constraints from NRC lexicon for each of the positive and negative relation sets.

We define our objective functions so that we decrease the angular distance between words with their associated emotion in the set S, and at the same time, increase their distance with their opposite emotions in the set O. We want the pairs of words in positive relation set to get closer together, so the objective function for positive relations would be:

$$PR(V') = \sum_{(u,w) \in S} max(0, d(v'_u, v'_w)) \tag{3.1}$$

c

where $d(v'_u, v'_w)$ is the cosine distance between the two vectors. Moreover, we want to increase the distance between pairs of words in our negative relation set. As value 1.0 represents dissimilarity we want to minimize 1.0 minus the distance, so the objective function for the negative relations would be:

$$NR(V') = \sum_{(u,w) \in O} max(0, 1 - d(v'_u, v'_w)) \tag{3.2}$$

We also need to make sure we lose as little information as possible by preserving the

Table 3.2: **Left**: Average similarity between opposite emotion groups. We want the similarity of opposite emotions to be as close to zero as possible. After training, the average similarities decrease for all models. **Right**: Average of in-category mutual similarity in three layered categorization of emotions before and after emotional fitting. We want the similarity of close emotions to be as close to one as possible. After training, the average similarity of in-category emotions increases for all models.

	Sadness vs. Joy		**Anger vs. Fear**		**In-category Similarity**	
	Before	**After**	**Before**	**After**	**Before**	**After**
Word2Vec	0.32	0.16	0.31	**0.09**	0.45	0.51
GloVe	0.23	0.11	0.19	0.04	0.38	**0.49**
fastText	0.38	0.17	0.33	0.12	0.44	0.50
Numberbatch	0.23	**0.10**	0.19	0.05	0.47	0.57

shape of our original vector space. In order to do this, we add a third part to our objective function to make sure we are not changing the overall shape of the space by much:

$$VSP(V, V') = \sum_{u=1}^{N} \sum_{w \in N(u)} |d(v'_u, v'_w) - d(v_u, v_w)| \tag{3.3}$$

For efficiency purposes, we only calculate the distance for a neighborhood of each word $N(i)$, which includes all words within the radius distance $r = 0.2$ of the word. So our final objective function is the sum of all three together:

$$Obj(V') = PR(V') + NR(V') + VSP(V, V') \tag{3.4}$$

Stochastic gradient descent was used for 20 epochs to train the vector space V and generate the new space V'.

3.1.4 Emotion Similarity Experiments

In our experiment, we compared various word embeddings with their emotionally fitted counterparts for various metrics based on emotional models. As we trained the model on Plutchik's model, we decided to use another emotion model for testing. In

the first experiment, we assess the average in-category mutual similarity of secondary and tertiary emotions in the three-level categorization of emotions described by [1]. In this model, Shaver et al. defined six basic emotions of *Liking, Joy, Surprise, Anger, Sadness, and Fear*, and categorized around 140 sub-emotions under these six emotions in two layers (See Table 3.1). The reported numbers are the average cosine similarity of all mutual in-category emotions words and can be seen in Table 3.2. The vector spaces used here are:

- Word2Vec trained full English Wikipedia dump
- GloVe from their own website
- fastText trained with subword information on Common Crawl
- ConceptNet Numberbatch

It is clear that each emotionally fitted vector space is preforming much better than its original counterpart from 13% improvement for the Word2Vec model to 29% for GloVe vectors. Overall best performance belongs to emotionally fitted ConceptNet Numberbatch by average similarity score of 0.57 up from 0.47 (See Table 3.2).

In the second experiment, we assessed the performance of the model for the similarity between opposite emotions. Again we used Shaver et al.'s categorization [1] as our testing emotion model and calculated the mutual similarity between opposite emotion groups. In this test we chose two pairs of opposite emotions, *Joy* vs. *Sadness* and *Anger* vs. *Fear*. The reported numbers are average cosine similarity between each member of the opposite emotion categories.

As shown in Table 3.2 the models perform significantly better after training with best performance for the retrained Numberbatch in distinguishing between *Anger* vs. *Fear* and retrained Glove for *Joy* vs. *Sadness* (with Numberbatch following closely).

3.1.5 Emotion Detection Experiments

To analyze emotional embeddings' performance in the emotion detection task, we have decided to use a simple neural network architecture to classify emotion *anger* in a large dataset. Here we compare the performance of emotional embedding with various training parameters with standard embeddings.

3.1.5.1 Distribution of Our Dataset

Not many datasets are labeled for emotions and large enough to train deep neural networks. For our classification experiment, a dataset of emotionally labeled tweets, created by [50], was used. This dataset consists of over 1.3 million tweets annotated for seven emotions. More detail about the data can be seen in Table 3.3.

Table 3.3: Number of tweets for each emotion.

Emotion	**Count**
joy	393,631
sadness	338,015
anger	298,480
love	169,267
fear	73,575
thankfulness	79,341
surprise	13,535
Total	**1,387,787**

3.1.5.2 The Model Used for the Experiment

To understand and compare different embeddings' impact on emotion detection classifiers, we decided to choose a simpler neural network model because simple models would be more sensitive to the embedding layer's changes. Our goal was not to build the best model for detecting emotion but to use a basic architecture to show the difference of using emotional vs. non-emotional word vectors in our embedding layer.

The model consists of an embedding layer, loaded with one of the embedding

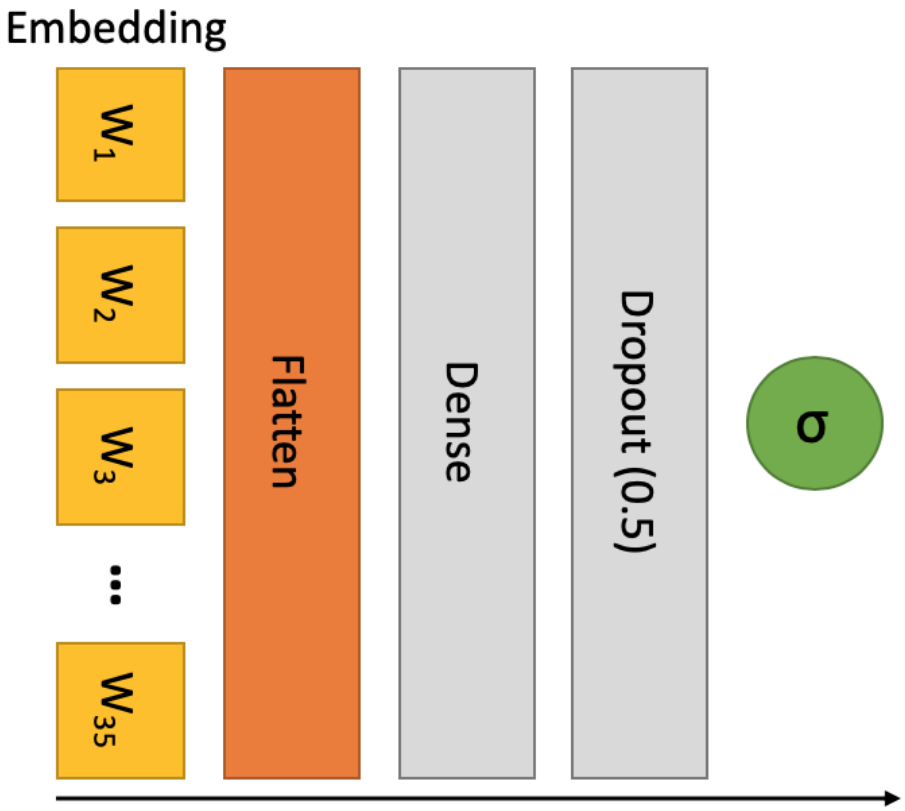

Figure 3.2: The architecture of the model used for all experiments.

models, with 35 inputs. Then, we used a dense layer of length 35 with a dropout layer with a rate of 0.5. The output layer is a sigmoid node for classification. We used the same model architecture for all of the experiments. Binary cross-entropy and Adam optimizer were used as our loss function and learning algorithm. The model was run using early stopping with a batch size of 350 for 10 epochs.

3.1.5.3 Experiment and Results

To analyze the effect retraining has on the emotion classification task's performance, we used our retrained embeddings to initialize the embedding layer in our detection model. We also trained various versions of the emotional embeddings and compared their performances using two different approaches:

First, we increased the effect of the two parts of our objective function that incorporate emotional relations, PR (positive relations), and NR (negative relations). At the beginning the value of k_1 and k_2 are equal in equation 3.5. Then we increase the value of the k_1 from 2 to 10 times the value of k_2. This is to see how much we can change the vector space's shape without losing the consistency of the original embeddings.

$$Obj(V') = k_1 PR(V') + k_1 NR(V') + k_2 VSP(V, V') \tag{3.5}$$

Second, we extend the NRC lexicon using WordNet [65]. We added the synonyms and derivatives of the word to our dictionary for each word-emotion pair and used the resulting lexicon during retraining. Two embedding models, FastText and NumberBatch, were used for these experiments. The results can be seen in Figure 3.3.

By looking at the results in Figure 3.3, we can see that the performance of all embedding models increases as the value of k_1 increases up to when it is two or three times k_2 and after that, the F-measures show no consistent trend. This shows that as the strength of retraining increases, the embedding model's emotional information increases, resulting in better performance in our classification model up to a point. After that, the vector space model changes become too severe, and the structure of the model changes too much to get any consistent results. We can also see that extending the lexicon, which results in more word vectors being effected during retraining, does not make any meaningful difference in our classification task. Both of these takeaways show us that when we are retraining the embedding models, subtlety is an important factor, meaning that we should find the minimal changes we can perform on the model toward our retraining goal that does not change the structure of the vector space.

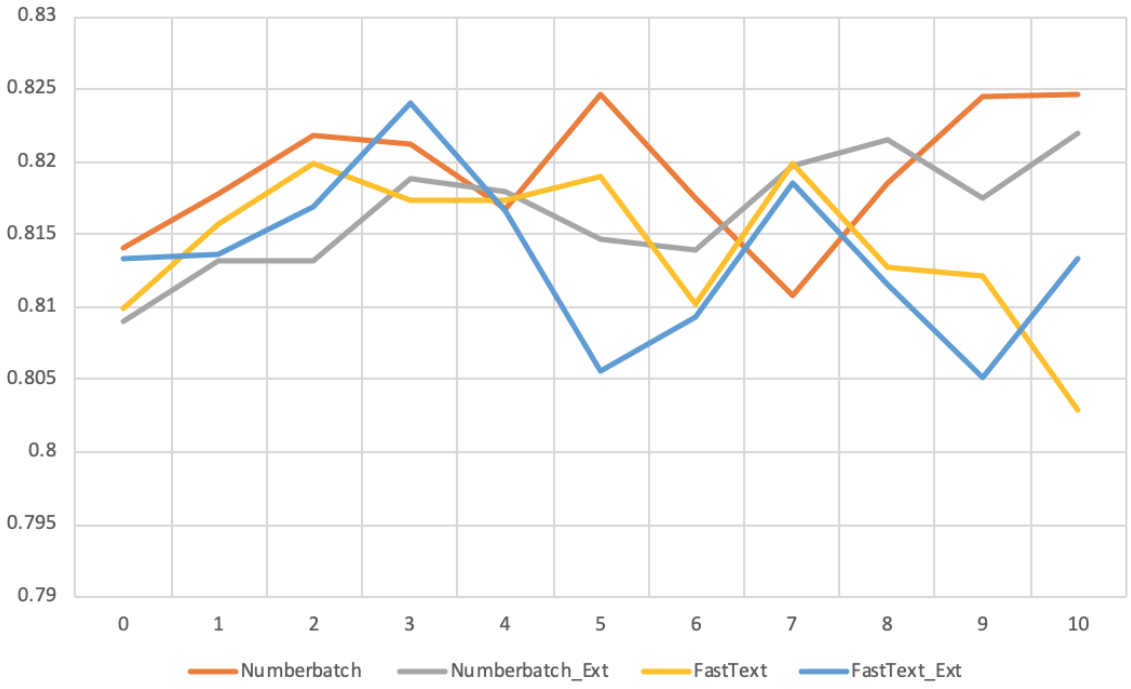

Figure 3.3: F-measure values for classifying *anger* for all embedding models. This chart shows how the F-measure reacts to an increase of k_2. On the X-axis shows the the relation between k_1 and $k_@$ (e.g. 2 means k_2 is double the k_1 value). Zero value shows the results for the original embedding model without any retraining. 'Ext' indicates that the embedding was retrained using the extended lexicon.

3.1.6 Conclusion

Embedding models are important for word representation in various natural language processing tasks. Although information such as semantic similarity and relatedness can be captured from the statistical information in the corpus, emotional information cannot. In this chapter, we described an approach that incorporates the emotional information of the words during the second stage of model training. We showed that this approach increases the performance of the embedding models, as measured by the cosine similarity of emotions in the same category and in opposite categories. For cosine similarity within the same category, Glove improved the most (0.11) and Word2Vec the least (0.06); for the cosine similarity between opposite

categories, Word2Vec improved the most (-0.22) and Glove the least (-0.12). The model with the best average performance across both metrics was Glove. In addition, we showed that these emotional embeddings can improve the performance of a simple neural network for emotion detection. Next, we investigated how increasing the strength of emotional retraining influences the emotion detection model. We found that by increasing the magnitude of the objective function's emotional relation segments, we could increase the performance of the model to a stable point, but beyond that, the performance became too erratic, likely because making much change in the structure of the vector space leads to loss of too much information. Therefore, by carefully retraining the word embedding models, we showed that we could successfully refine them to contain emotional information about words, leading to better emotion detection classifiers.

3.2 Emotion Detection In Text

In recent years, emotion detection in text has become more popular due to its vast potential applications in marketing, political science, psychology, human-computer interaction, artificial intelligence, etc. In this work, we argue that current methods based on conventional machine learning models cannot grasp the intricacy of emotional language by ignoring the text's sequential nature and context. These methods, therefore, are not sufficient to create an applicable and generalizable emotion detection methodology. Understanding these limitations, we move toward methodologies that can utilize the information in the text's sequential nature and context. We present results from a network based on a bidirectional GRU model and attention mechanism along with the fine-tuned transformer model (BERT) to show that capturing more meaningful information from the text can significantly improve the performance of emotion detection models. The results show significant improvement over the conventional method on the same dataset with an average of 26.8 point increase in F-measure on our test data and 38.6 increase on the totally new dataset. We show that a bidirectional-GRU with attention could perform better than BERT.

3.2.1 Introduction

There have been many advances in machine learning methods to help machines understand human behavior better than ever. One of the most important aspects of human behavior is emotion. If machines could detect human emotional expressions, it could be used to improve verity of applications in marketing [15], human-computer interactions [16], political science [12] etc. For example, understanding the emotional response of a customer to a product (rather than just positive/negative feedback) can be used to plan the marketing strategies in response to those emotions. This will then result in better customer satisfaction and more useful products. Also, customer

service efficiency in a company can be assessed based on the emotional reactions of customers during their contact with support staff.

Emotion in humans is complex and hard to distinguish. There have been many emotional models in psychology which tried to classify and point out basic human emotions such as Ekman's 6 basic emotions [24], Plutchik's wheel of emotions [118], or Parrott's three-level categorization of emotions [26]. These varieties show that emotions are hard to define, distinguish, and categorize even for human experts.

Adding the complexity of language and the fact that emotion expressions are very complex and context dependant [29, 30, 31], we can see why detecting emotions in textual data is a challenging task. This difficulty can be seen when human annotators[124]. Tafreshi et al. observed that, in the annotation process, the annotators tend to get confused when they have to select one emotion in each of these categories: anger, disgust, fear or trust, joy, anticipation.

Until a few years ago, most emotion detection techniques used conventional machine learning to classify a given text into one or more emotional categories [125, 126]. However, these methods were not very successful in detecting and classifying emotions in text because emotion detection requires context and understanding the sequence of words in a sentence. For these reasons, recurrent neural networks (RNNs) seem to be a great candidate to tackle such problems.

In this paper, we propose and compare three different methodologies based on recurrent neural networks and compare them to the previous state of the art model PAPER that is now our baseline. We first proposed a bidirectional GRU model, which has improved the baseline by 26.8%. In the second section, we added an attention mechanism to improve the contextual understanding of our model. This improved our results by almost 1%. In the last approach, we fine-tuned BERT to classify the emotions. Our GRU with attention mechanism performed on par with the BERT

model with a higher average performance of 1% in F-measure.

In the rest of this paper, we first start with a brief literature review of emotion detection methodologies and research. In Section 3, we explain our choice of data and its preparation process. Section 4 splits into four subsections. In each subsection, we explain a different methodology, model specification, and results compared to the previous state of the art model. Section 5 is the conclusion.

3.2.2 Related Works

Much work has been done on detecting emotion in speech or visual data [127, 128, 129, 130]. However, detecting emotions in textual data is a relatively new area that demands more research. There have been many attempts to detect emotions in text using conventional machine learning techniques and handcrafted features [85, 86, 60, 87, 88, 89, 90, 50, 91, 92, 94, 96, 99]. During the process of creating the feature set, in these methods, some of the most important information in the text, such as the sequential nature of the data, and the context will be lost. These attempts lead to methods that are not scalable and generalizable.

Due to the sequential nature of textual data, recurrent and convolutional neural networks have been used in many NLP tasks and improved the performance in a variety of classification tasks [131, 132, 133, 134]. However, emotion detection has gotten less attention. In recent years there have been some works in using deep neural network for emotion detection in text [135, 136, 137, 138, 139, 140, 141, 142, 143, 144].

Although achieving good results, all these papers have been trained on relatively small datasets (SemEval dataset has about 30k of records in training data for 4 labels). The small training data leads to models that are not scalable or generalizable, so they tend not to be useful for real-world applications—besides, the small size of data limits model choices. Our dataset, compared to most of the datasets used, has over 1.3M

number of data (for seven labels), which allows us to use state-of-the-art models used in other NLP tasks, such as BERT [145], and attention models [146, 120]. Complex models trained and fine-tuned on larger datasets achieve more valuable results than when fine-tuned on smaller datasets.

These models can better capture the complexity and context of the language by keeping the sequential information of text and creating latent representation for the text as a whole and by learning the important features without any additional (and often incomplete) human-designed features. In [48], as part of a summary of the SemEval 2019 competition, the authors show that the winning teams have used contextual embeddings or models to achieve the best performances.

In this work, we use a larger dataset compared to most research in this area and compare different deep neural network architectures. To the best of our knowledge, our proposed GRU with attention model achieved the state of the art result and performed slightly better than BERT on the task of emotion detection.

3.2.3 Data and Preparation

There are not many free datasets available for emotion classification. Most datasets are subject-specific (i.e., news headlines, fairy tails, etc.) and not big enough to train deep neural networks. Here we use the tweet dataset created by Wang et al. mentioned in the previous section (See Table 3.4).

Wang et al. [50] downloaded over 5M tweets, which included one of 131 emotional hashtags based on Parrott's three-level categorization of emotions in seven categories: *joy, sadness, anger, love, fear, thankfulness, surprise*. After comparing human annotations by hashtag labels on a sample of the data, the authors came up with simple heuristics to increase labeling quality by ignoring tweets with quotations and URLs and only keeping tweets with 5 terms or more that have the emotional hashtags at

Table 3.4: Statistics in the original dataset from Wang et al. (2012)

Emotion	**Hashtags**	**Original**	**Ours**
joy	36	706,182	393,631
sadness	36	616,471	338,015
anger	23	574,170	298,480
love	7	301,759	169,267
fear	22	135,154	73,575
thankfulness	2	131,340	79,341
surprise	5	23,906	13,535
Total	**131**	**2,488,982**	**1,387,787**

the end of the tweets. Using these rules, they extracted around 2.5M tweets. After sampling another 400 random tweets and comparing it to human annotation, they saw that hashtags could classify the tweets with 95% precision.

As Twitter is against publishing this many tweets, Wang et al. provided the tweet ids along with their label. For our experiment, we retrieved the tweets in Wang et al.'s dataset by tweet IDs. We could only download over 1.3 million tweets from around 2.5M tweet IDs in the dataset, as not all the tweets on the list of tweet IDs were available. The distribution of the data can be seen in Table 3.4.

In our experiment, we used simpler pre-processing steps, which will be explained later on in section 3.2.4.

3.2.4 Experiments

In this section, we have tried various models with different complexities. In 3.2.4.1, we explain the baseline approach published in [50]. We use their published results as the baseline to compare our model improvements. Our first proposed model is explained in 3.2.4.2 is a bidirectional GRU model trained on the same dataset. This model had significant improvement compared to baseline, with an average increase of 26.8% in F-measure. We also showed that this model works great on a new unseen dataset. The next model, in 3.2.4.3, is a bidirectional GRU with an additional at-

tention layer with various architectural designs. Using only attention on top of GRU improved our previous results in almost every emotion category except for *Surprise*. In our last model, in 3.2.4.5 we fine tuned BERT on our dataset. Compared to our model, BERT has performed better on some of the emotion categories and performed worst on others. On average, our GRU with attention model performed slightly better compared to BERT.

Bellow, we will explain all the experiments in more detail.

3.2.4.1 Baseline Approaches

We used the Wang et al. model all through this paper as our baseline model. All of our training was performed on their dataset. Wang et al. used two different classifiers (multinomial Naive Bayes and LIBLINEAR) and 12 different feature sets. Their best results were achieved using LIBLINEAR classifier and a feature set containing n-gram (n=1,2), LIWC and MPQA lexicons, WordNet-Affect, and POS tags can be seen in Table 3.5. It can be seen that their best results were for high count emotions like *joy* and *sadness* for as high as 72.1 in F-measure. Their model did not perform well for low count emotion labels (*surprise* with F-measure of 13.9).

To understand our trained models' generalization capability, we used the Crowd-Flower dataset as a secondary test set. We downloaded this dataset from a paper by [46]. In the paper, the authors used a maximum entropy classifier with the bag of words model to classify various emotional datasets. The CrowdFlower dataset had 12 different emotion labels. We only used the part of the dataset that the emotions could be mapped to one of our seven labels.

The result of the baseline is presented in the following sections when compared to our different models.

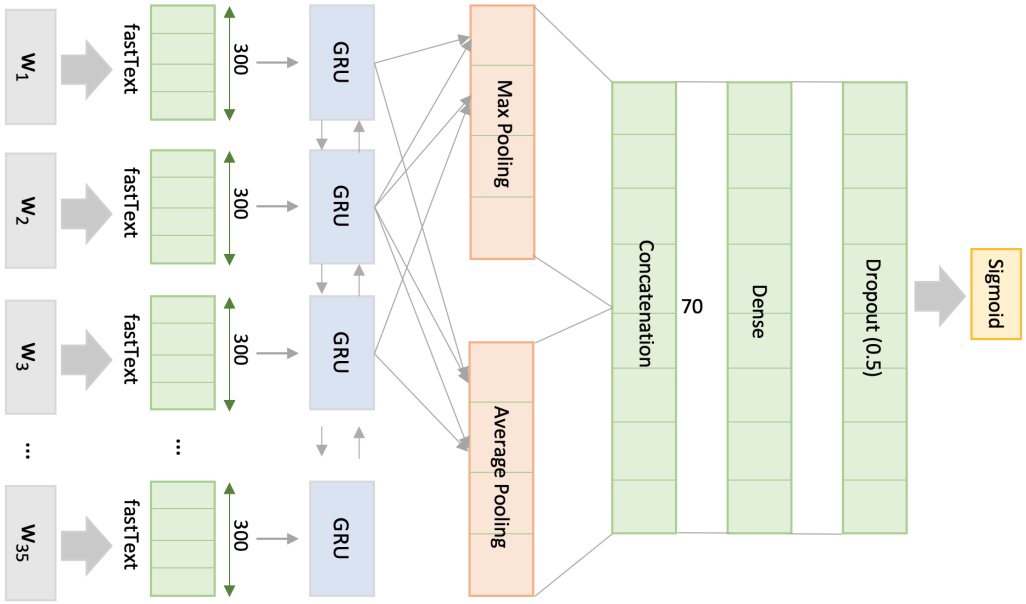

Figure 3.4: Bidirectional GRU architecture used in our experiment.

3.2.4.2 Experiment 1: Bidirectional GRU

Model Architecture This section introduces the deep neural network architecture that we used to classify emotions in the tweets dataset. Emotional expressions are more complex and context-dependent even compared to other forms of expressions based mostly on the complexity and ambiguity of human emotions and emotional expressions and the huge impact of context on understanding the expressed emotion. These complexities lead us to believe lexicon-based features normally used in conventional machine learning approaches are unable to capture the intricacy of emotional expressions.

Our first proposed architecture was designed to show that using a model that captures better information about the text's context and sequential nature can outperform lexicon-based methods commonly used in the literature. As mentioned in the **??**, Recurrent Neural Networks (RNNs) have been shown to perform well for the verity of tasks in NLP, especially classification tasks. We decided to use a model

based on bidirectional RNN, specifically a bidirectional GRU network, to analyze the tweets. GRU has one hidden state less than LSTM, which results in faster performance. LSTMs are better for longer sequences as they can contain more sequential information; however, as our tweets are fairly short, we opted for GRU for computational performance reasons. [1]

For building the emotion classifier, we have decided to use 7 binary classifiers -one for each emotion- each of which uses the same architecture for detecting a specific emotion. You can see the plot diagram of the model in Figure 3.4. The first layer consists of an embedding lookup layer initialized with a specific word embedding model and will be used to convert each term to its corresponding embedding vector. In our experiments, we tried various word embedding models but saw little difference in their performance. Here we report the results for two, which had the best performance among all, ConceptNet Numberbatch [147] and fastText [148] both had 300 dimensions. All of our models used fastText.

As none of our tweets had more than 35 terms, we set the embedding layer's size to 35 and added padding to shorter tweets. This layer's output goes to a bidirectional GRU layer selected to capture the entirety of each tweet before passing its output forward. The goal is to create an intermediate representation for the tweets that capture the data's sequential nature. We use a concatenation of global max-pooling and average-pooling layers (with a window size of two) for the next step. Then a max-pooling was used to extract the most important features form the GRU output, and an average-pooling layer was used to considers all features to create a representation for the text as a whole. These partial representations are then were concatenated to create our final latent representation. For classification, the output of the concatenation is passed to a dense classification layer with 70 nodes along with a dropout

[1] Code available at: https://github.com/armintabari/Emotion-Detection-RNN

layer with a rate of 50% to prevent over-fitting. The final layer is a Sigmoid layer that generates the classifier's final output, returning the class probability.

Table 3.5: Results of classification using bidirectional GRU. Reported numbers are F1-measures.

Emotion	**Wang%.**	**Ours%**	**Diff%**
joy	72.1	82.1	10.0
sadness	64.7	79.2	14.5
anger	71.5	83.7	12.2
love	51.5	80.3	28.8
fear	43.9	78.1	34.2
thankfulness	57.1	83.6	26.5
surprise	13.9	75.6	61.7
Average	53.5	80.4	26.8

Results Minimal pre-processing was done by converting text to lower case, removing the hashtags at the end of tweets, and separating each punctuation from the connected token (e.g., awesome!! $\rightarrow$ awesome !!) and replacing comma and new-line characters with white space. The text, then, was tokenized using the TensorFlow-Keras tokenizer. Top N terms were selected and added to our dictionary where N=100k for higher count emotions *joy, sadness, anger, love* and N=50k for *thankfulness and fear* and N=25k for *surprise*. Seven binary classifiers were trained for the seven emotions with a batch size of 250 and 20 epochs with binary cross-entropy as the objective function and Adam optimizer. The architecture of the model can be seen in Figure 3.4. For training each classifier, a balanced dataset was created with selecting all tweets from the target set as class 1 and a random sample of the same size from other classes as class 0. For each classifier, 80% of the data was randomly selected as the training set, and 10% for the validation set, and 10% as the test set. As mentioned before, we used the two embedding models, ConceptNet Numberbatch

and fastText, as the two more modern pre-trained word vector spaces to see how changing the embedding layer can affect the performance. The result of comparison among different embeddings can be seen in Table 3.6. It can be seen that the best performance was divided between the two embedding models with minor performance variations.

The comparison of our result with Wang et al. can be seen in Table 3.5. Our model's results show significant improvement from 10% increase in F-measure for a high count emotion *joy* to up to 61.7 point increase in F-measure for a low count emotion *surprise*. On average, we see a 26.8 point increase in F-measure for all categories, and more interestingly, our result shows a minimal variance between different emotions compare to results reported by Wang et al.

Model Performances on CrowdFlower To assess these models' performance on a totally unseen data, we tried to classify the CrowdFlower emotional tweets dataset. The CrowdFlower dataset consists of 40k tweets annotated via crowd-sourcing, each with a single emotional label. This dataset is considered a hard dataset to classify with many noise [46]. The labeling on this dataset is non-standard, so we used the following mapping for labels:

- sadness $\rightarrow$ sadness
- worry $\rightarrow$ fear
- happiness $\rightarrow$ joy
- love $\rightarrow$ love
- surprise $\rightarrow$ surprise

- anger → anger

We then classified emotions using the pre-trained models and fastText embedding. The result can be seen in Table 3.7. The baseline results are from [46] done using the BOW model and maximum entropy classifier. We saw a huge improvement from a 26 point increase in F-measure for the emotion *joy (happiness)* up to 57 point increase for *surprise* with a total average increase of 38.6 points. Bostan and Klinger did not report classification results for the emotion *love*, so we did not include it in the average. These results show that our trained models perform exceptionally on a totally new dataset with different annotation methods.

Table 3.6: Results of classification using two embedding models and bidirectional GRU. No meaningful differences were seen between the two models. Reported numbers are F1-measures.

Emotion	**Numberbatch**	**fastText**
joy	**82.11**	81.90
sadness	**79.17**	78.71
anger	83.44	**83.74**
love	79.83	**80.29**
fear	77.61	**78.11**
thankfulness	**83.64**	83.58
surprise	75.40	**75.58**

Table 3.7: Results from classifying CrowdFlower data using pre-trained model. Reported numbers are F1-measure.

Emotion	**Baseline%**	**Ours%**	**Difference**
joy (happiness)	38	64	26
sadness	27	65	38
anger	24	62	38
love	-	66	-
fear (worry)	31	65	34
surprise	9	66	57
Average	**25.8**	**63.2**	**38.6**

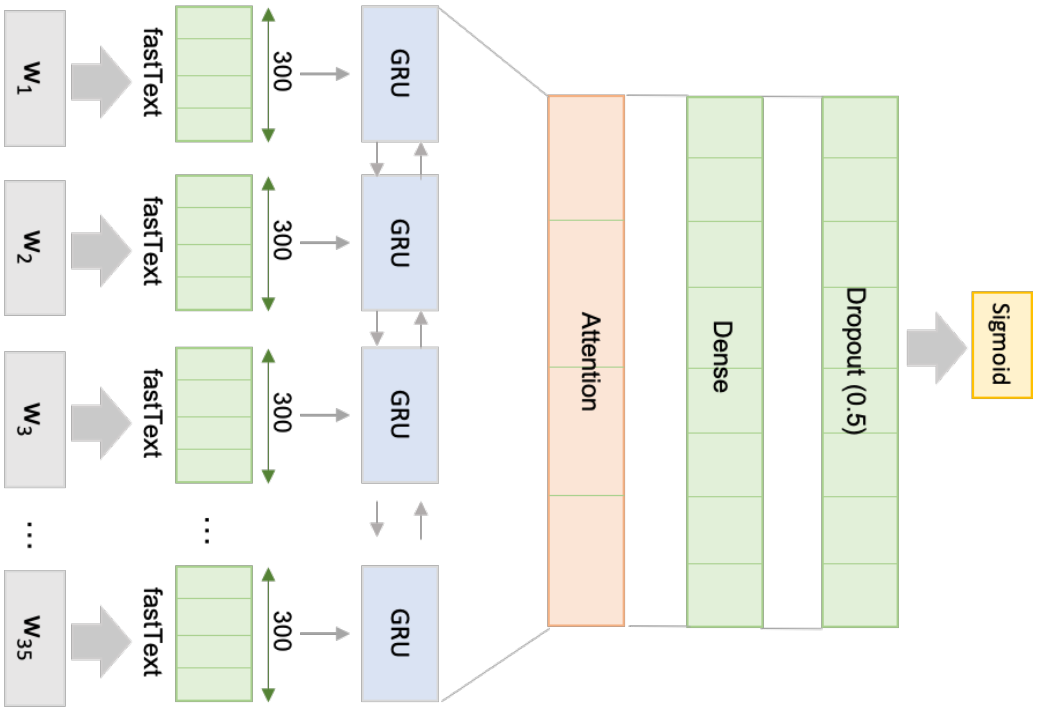

Figure 3.5: An attention Layer has been used to generate the latent representations.

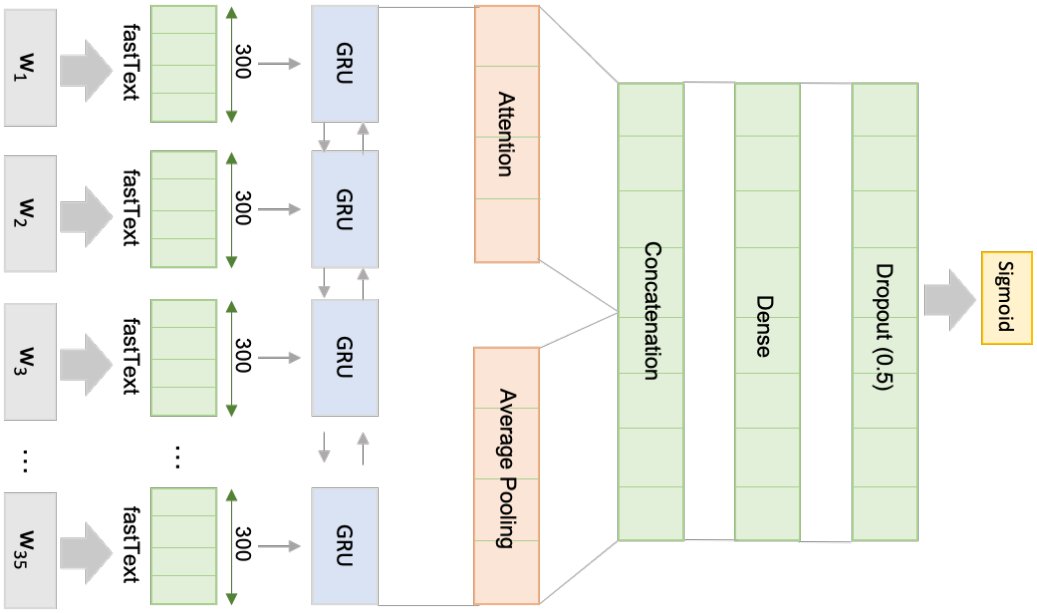

Figure 3.6: Concatenation of outputs from attention layer and average-pooling has been used to generate the latent representation.

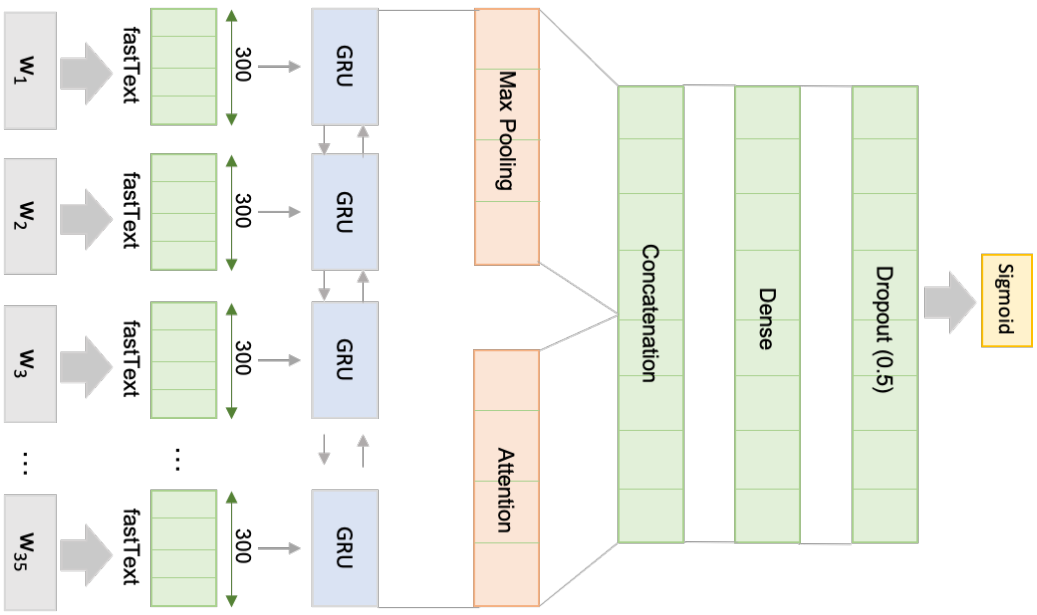

Figure 3.7: Concatenation of attention and max-pooling has been used to build the latent representation.

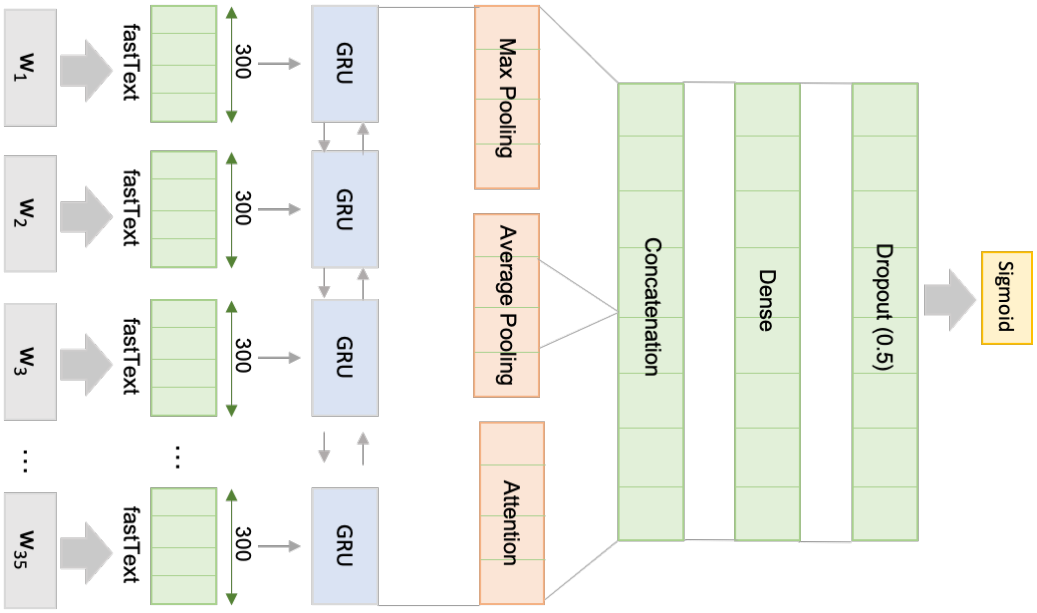

Figure 3.8: Concatenation of attention layer with both average and max-pooling layers has been used to generate the representation.

3.2.4.3 Experiment 2: Using Attention Mechanism

Attention mechanism has been shown to improve the performance of various NLP tasks by allowing the network to create a more context dependant latent representation for the terms in the text. In this section, we assess the effect of adding an attention layer for generating our latent representation.

The attention mechanism is responsible for understanding the interdependence among elements in the model (inputs and output words). The attention layer learns to add extra weight to words that carry important and relevant information from input words, enhancing the quality of the representation and, as a result, the model's performance.

Model Architeture The model used in this experiment incorporates attention mechanism in various combinations with max-pooling and average-pooling. We have used the base model from experiment 3.2.4.2 and added an attention layer either alone or along with max-pooling and average-pooling layers (Figure 3.5).

Four different models were used to assess the best use of the attention layer in our network. In the first model, we passed the output of the bidirectional GRU to an attention layer and used the output from the attention layer as our latent representations. In the second and third models, we used concatenation of the output from the attention layer with average-pooling and max-pooling, respectively, to build our latent representations. Furthermore, in the last model, we concatenated the output from all three (attention, max-pooling, and average-pooling) as the latent representations.

Other model parameters were set the same as the experiment 3.2.4.2. The dataset was partitioned to train, validation, and test set randomly with the same ratio, and

Table 3.8: Results of classification using two embedding models and bidirectional GRU with attention layer to generate latent representations. Reported numbers are F1-measures. FT: fastText, NB: NumberBatch, max: Max-pooling, avg: Average-pooling, att: Attention layer

Model	**joy**		**sadness**		**anger**		**love**		**fear**		**thankfulness**		**surprise**	
	NB	**FT**	**NB**	**FT**	**NB**	**FT**	**NB**	**FT**	**NB**	**FT**	**NB**	**FT**	**NB**	**FT**
GRU(max+avg)	78.99	80.94	75.56	78.23	81.56	82.76	74.96	79.3	75.71	81.32	83.02	84.71	64.11	**76.75**
GRU(att)	79.08	**81.67**	76.28	**78.88**	80.75	**83.25**	74.72	**79.71**	75.94	**81.68**	83	**85.17**	67.01	75.76
GRU(att+avg)	79.17	81.36	75.96	77.71	81.5	82.78	75.39	79.34	76.23	80.1	82.97	85.16	65.35	75.95
GRU(att+max)	78.68	81.08	75.6	78.34	81.5	83.06	75.49	78.87	75.58	81.47	82.87	84.73	64.12	75.64
GRU(att+both)	78.89	81.31	75.44	78.67	81.35	83.04	75.36	79.47	76.72	81.33	83.07	84.71	66.29	76.38

the same partitioning has been used through all experiments.

Two different embedding models were used to compare a standard embedding model, fastText, and a model that has gone through an extra stage of training, numberBatch.

Results The full results can be seen in Table 3.8. As shown in Table 3.8, fastText outperformed numberBatch through all four detection models significantly. We believe the extra training stage in creating Numberbatch representation caused some loss in information learned in the first stage of training.

For six emotions of *joy, sadness, anger, love, fear* and *thankfulness*, the model that only used the attention layer to generate the representation out-performed the other models. The only exception was the emotion *surprise* for which the original model in experiment 3.2.4.2 out-performed all others. This can be due to the significant decrease in the size of the training data for this emotion, making learning harder for the more complex models.

3.2.4.4 Experiment 3: Using Emotional Embeddings for Emotion Detection

In section 3.1 we introduced emotional word embeddings, and showed that they can improve the performance in emotion detection using a simpler neural network model. In this experiment, we intend to assess the performance of these embedings

Table 3.9: Results of classification using emotional embedding models and bidirectional GRU with attention mechanism. Reported numbers are F1-measures.

Emotion	**fastText**	**Emotional fT**
joy	81.67	**83.62**
sadness	78.88	**80.81**
anger	83.25	**84.16**
love	79.71	**82.22**
fear	81.68	**83.23**
thankfulness	85.17	**86.29**
Surprise	75.76	**76.83**

in a more complex emotion detection model.

Here we use the best performing emotional embedings, generated by retraining a fastText model as described in section 3.1.5.3. We also choose our best performing emotion detection model that only uses attention mechanism to create the text representation. By keeping all the training parameters the same as before, and just changing the embedding layer to emotional embedding, we want to assess the effect of using emotional embeddings. You can see the results in Table 3.9. These results show that using emotional embeddings can improve the performance of our emotion detection model significantly.

3.2.4.5 Experiment 4: Using Transformers

Transformers are based on encoder/decoder models and attention mechanisms trained as a language model on a huge corpus. These models have outperformed many previous deep neural network models in various NLP tasks.

In this section, we fine-tune a transformer model, BERT, with our dataset and compare it with our previous models. BERT (Bidirectional Encoder Representations from Transformers) is a multi-layer multi-head Transformer model with attention and self-attention mechanism. The key innovation of BERT is to apply bidirectional training of Transformers with the Masked Language Model mechanism. This has resulted

in BERT to outperform all previous models in various NLP tasks.

Model Architecture BERT (Bidirectional Encoder Representations from Transformers) is a transformer model developed by Google and achieves state-of-the-art performance in various NLP tasks. Standard embeddings are context-free and generate a representation for each term in the corpus, while BERT, like attention models, considers the context in which the term has appeared. As one of the most used transformer models in the literature, we decided to use it to assess its performance in our emotion detection task.

For this experiment, we have used HuggingFace[2] to fine-tune and run the BERT model. We have chosen to use the BERT-base model and fine-tune it with our data. The same proportions of 80%, 10%, 10% for the train, validation, and test datasets were used, and the tweets were undergone the same minimal pre-processing (e.g., making the text lowercased, and separating punctuation from words). Due to a lack of memory, a smaller batch size of 32 was used, and the model was run for 5 epochs. We fine-tuned on a single GPU (Nvidia Tesla K80) with 2GB memory.

Results A comparison of the GRU model's best results to the BERT outcomes can be seen in Table 3.10. The best performance for different emotions was split between the two models, with GRU having a slightly higher average F-measure. Also, it is important to mention that the GRU model has the advantage of being substantially faster to train over BERT. For comparison, the highest number of trainable parameter in any of our GRU models were around 30 million, while the BERT model had more than 108 million trainable parameters.

[2]https://github.com/huggingface

Table 3.10: Comparison of results between our model with standard (fT) and emotional (Emo-fT) embeddings with the fine-tuned BERT. Reported numbers are F1-measures.

Emotion	**GRU+fT**	**GRU+Emo-fT**	**BERT**
joy	81.67	**83.62**	81.69
sadness	78.88	**80.81**	77.86
anger	83.25	84.16	**84.76**
love	79.71	**82.22**	80.72
fear	81.68	**83.23**	78.48
thankfulness	85.17	86.29	**86.35**
surprise	76.75	**76.83**	74.53
Average	81.01	**82.45**	80.62

3.2.5 Conclusion

In this chapter, we designed a network based on a bidirectional GRU with attention mechanism and found that it dramatically improved classification performance. We showed that by taking advantage of the sequential nature of the data, this model can detect complex attributes of textual data such as emotions. We also showed that it creates a more informative latent representation of text by incorporating a recurrent network that capture the sequential nature of the text, a max-pooling layer that captures the most relevant features, and an average pooling layer that captures the text as a whole.

We also showed that the added attention mechanism improves the performance of the model by creating a more contextual representation of the text. This is best observed by comparing our model to a transformer model, BERT. We found that our results using standard embedding are on par with a fine-tuned BERT model, but requires only a fraction of the computational power. Also we could significantly increase the performance by using emotional embedding model, and, on average, could acheived 1.83 point increase in F-measure compared to the BERT model. These results show that with the availability of large enough labeled datasets for training,

the task of emotion detection can move from conventional machine learning toward deep neural networks with results that can encourage wide usage in many high-end applications or as subtasks in a variety of natural language processing activities.

CHAPTER 4: Summary and Future Direction

Emotion detection in text has been on the sideline for a long while, owing largely to the complexity of human emotions and emotional expression, and the inability of conventional methodologies to capture them. Most techniques developed were unreliable and dataset-specific, and this stymied research in this field. The last few years, however, has seen a small surge in published literature on this topic.

In this work, we first reviewed various emotional models found in the psychology literature, especially those also used in computer science literature. For example, discrete models of emotions, such as Ekman's or Plutchik's, and dimensional representation of emotions. We also argued that complexity of emotional expression, like vague language and context-dependency, makes it harder to capture the emotion present in the text.

Emotion detection is a relatively young field, and as such, the available resources and methodologies are mostly inadequate, and are mostly designed for use with conventional machine learning methodologies. For instance, the largest standard dataset labeled for emotions is SemEval-2019, which only contains around 30,000 data points has labels for only four categories of emotion. In Chapter 2, we discussed all the datasets that are available.

In Chapter 2.4, we presented the current methodologies from the literature that are used for classifying emotions. In general these techniques do not perform well, but when they do, it is because they are fine-tuned for a specific dataset, making it difficult or impossible that they can be transferred and scaled for use with a new dataset.

These models suffer from their inability to capture the general complexity of language and the context-dependency of emotional language. To address the shortcomings of current emotion detection models, we argued that a more complex model was necessary, and that it was essential that this model be able to capture the sequential nature of textual data and to understand the contextual meaning of words. Therefore, we decided that Deep Neural Networks were the best candidate to build such a model.

We first showed that word embeddings, as one of the essential tools in modern natural language processing, could not capture the emotional meanings of words. To address this shortcoming, we devised a secondary stage of training that incorporates emotional weights of words into these models after the fact and can be used for any word embedding model. These models showed significant improvement, as measured by emotional similarity metrics and by their performance in emotion classification tasks using Deep Neural Networks after retraining.

For our main task of emotion detection, we tried several architectures before finding the best model. We decided to use a Bidirectional-GRU (Bi-GRU) to capture the inter-dependencies of words to generate the first-level latent representation for our data, and then tried various layers on top of it to see which one (or ones) perform better in generating our final latent representation. We tried all combinations of max-pooling, average-pooling, and attention layers, and found that the model performed best when using only an attention layer. Attention layers are designed to capture the contextual meaning of words and can take care of various subtleties in the text, such as negations, words with multiple meanings, and contextual meaning of words, proving our hypothesis that understanding the context of words is important in emotion detection. Attention also takes takes into account the sequential information in long sequences, which is lost with conventional methods.

Because Transformer models have been shown to outperform other NLP models in

various tasks, we also fine-tuned a BERT model using our data and compared the results with our Bi-GRU with the attention model. The results of the BERT model were on par with our model, showing only a slight decrease in F-measure (less than 1%) averaged across all emotions.

In this work, we showed that emotion detection can be:

1. Reliable so long as a model can capture all the complexities of emotional language.

2. Transferable and generalizable, given a large and diverse dataset is used to train such a model.

From the advances presented in this work, two directions for future work present themselves. First, it is clear that large datasets, specifically annotated for emotions, are essential, and creating such datasets will help to move emotion detection in text forward and lead to more generalizable models. Secondly, having access to such datasets, we can expand these models to detect finer information such as the intensity of emotions.

REFERENCES

[1] P. Shaver, J. Schwartz, D. Kirson, and C. O'connor, "Emotion knowledge: Further exploration of a prototype approach.," *Journal of personality and social psychology*, vol. 52, no. 6, p. 1061, 1987.

[2] G. Qiu, X. He, F. Zhang, Y. Shi, J. Bu, and C. Chen, "DASA: Dissatisfaction-oriented Advertising based on Sentiment Analysis," *Expert Systems with Applications*, vol. 37, no. 9, pp. 6182–6191, 2010.

[3] X. Jin, Y. Li, T. Mah, and J. Tong, "Sensitive webpage classification for content advertising," in *Proceedings of the 1st International Workshop on Data Mining and Audience Intelligence for Advertising*, ADKDD '07, (New York, NY, USA), pp. 28–33, ACM, 2007.

[4] S. Somasundaran, T. Wilson, J. Wiebe, and V. Stoyanov, "Qa with attitude: Exploiting opinion type analysis for improving question answering in on-line discussions and the news.," in *ICWSM*, 2007.

[5] V. Stoyanov, C. Cardie, and J. Wiebe, "Multi-perspective question answering using the opqa corpus," in *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 923–930, Association for Computational Linguistics, 2005.

[6] L. V. Lita, A. H. Schlaikjer, W. Hong, and E. Nyberg, "Qualitative dimensions in question answering: Extending the definitional qa task," in *PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE*, vol. 20, p. 1616, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005.

[7] Y. Seki, K. Eguchi, N. Kando, and M. Aono, "Multi-document summarization with subjectivity analysis at duc 2005," in *Proceedings of the Document Understanding Conference (DUC)*, 2005.

[8] L. Terveen, W. Hill, B. Amento, D. McDonald, and J. Creter, "Phoaks: A system for sharing recommendations," *Commun. ACM*, vol. 40, pp. 59–62, Mar. 1997.

[9] E. Riloff, J. Wiebe, and W. Phillips, "Exploiting subjectivity classification to improve information extraction," *Proceedings of the 20th national conference on Artificial intelligence*, vol. 20, no. 3, pp. 1106–1111, 2005.

[10] J. S. Lerner and D. Keltner, "Beyond valence: Toward a model of emotion-specific influences on judgement and choice," *Cognition & emotion*, vol. 14, no. 4, pp. 473–493, 2000.

[11] D. A. Miller, T. Cronin, A. L. Garcia, and N. R. Branscombe, "The relative impact of anger and efficacy on collective action is affected by feelings of fear," *Group Processes & Intergroup Relations*, vol. 12, no. 4, pp. 445–462, 2009.

[12] J. N. Druckman and R. McDermott, "Emotion and the framing of risky choice," *Political Behavior*, vol. 30, pp. 297–321, Sep 2008.

[13] A. Bartsch, P. Vorderer, R. Mangold, and R. Viehoff, "Appraisal of emotions in media use: Toward a process model of meta-emotion and emotion regulation," *Media Psychology*, vol. 11, no. 1, pp. 7–27, 2008.

[14] B. Huebner, S. Dwyer, and M. Hauser, "The role of emotion in moral psychology," *Trends in cognitive sciences*, vol. 13, no. 1, pp. 1–6, 2009.

[15] R. P. Bagozzi, M. Gopinath, and P. U. Nyer, "The role of emotions in marketing," *Journal of the academy of marketing science*, vol. 27, no. 2, pp. 184–206, 1999.

[16] S. Brave and C. Nass, "Emotion in human–computer interaction," *Human-Computer Interaction*, p. 53, 2003.

[17] N. Gupta, M. Gilbert, and G. D. Fabbrizio, "Emotion detection in email customer care," *Computational Intelligence*, vol. 29, no. 3, pp. 489–505, 2013.

[18] C. Peter and R. Beale, *Affect and emotion in human-computer interaction: From theory to applications*, vol. 4868. Springer Science & Business Media, 2008.

[19] S. Voeffray, "Emotion-sensitive human-computer interaction (hci): State of the art-seminar paper," *Emotion Recognition*, pp. 1–4, 2011.

[20] F. Rangel and P. Rosso, "On the impact of emotions on author profiling," *Information processing & management*, vol. 52, no. 1, pp. 73–92, 2016.

[21] A. Bechara, "The role of emotion in decision-making: Evidence from neurological patients with orbitofrontal damage," *Brain and Cognition*, vol. 55, no. 1, pp. 30 – 40, 2004. Development of Orbitofrontal Function.

[22] E. Hudlicka, "Guidelines for designing computational models of emotions," *International Journal of Synthetic Emotions (IJSE)*, vol. 2, no. 1, pp. 26–79, 2011.

[23] G. Colombetti, "From affect programs to dynamical discrete emotions," *Philosophical Psychology*, vol. 22, no. 4, pp. 407–425, 2009.

[24] P. Ekman, "An argument for basic emotions," *Cognition & emotion*, vol. 6, no. 3-4, pp. 169–200, 1992.

[25] R. Plutchik, "Emotions: A general psychoevolutionary theory," *Approaches to emotion*, vol. 1984, pp. 197–219, 1984.

[26] W. G. Parrott, *Emotions in social psychology: Essential readings*. Psychology Press, 2001.

[27] J. A. Russell, "A circumplex model of affect.," *Journal of personality and social psychology*, vol. 39, no. 6, p. 1161, 1980.

[28] R. A. Calvo and S. Mac Kim, "Emotions in text: dimensional and categorical models," *Computational Intelligence*, vol. 29, no. 3, pp. 527–543, 2013.

[29] A. Ben-Ze'ev, *The subtlety of emotions*. MIT Press, 2000.

[30] C. Bazzanella, "Emotions, language and context," *Emotion in dialogic interaction: Advances in the Complex*, pp. 55–72, 2004.

[31] K. Oatley, D. Keltner, and J. M. Jenkins, *Understanding emotions*. Blackwell publishing, 2006.

[32] A. Balahur and A. Montoyo, "Applying a culture dependent emotion triggers database for text valence and emotion classification," *Procesamiento del lenguaje natural*, vol. 40, 2008.

[33] A. Pavlenko, "Emotion and emotion-laden words in the bilingual lexicon," *Bilingualism: Language and cognition*, vol. 11, no. 2, pp. 147–164, 2008.

[34] S. Y. M. Lee, "A linguistic analysis of implicit emotions," in *Workshop on Chinese Lexical Semantics*, pp. 185–194, Springer, 2015.

[35] S. Greene and P. Resnik, "More than words: Syntactic packaging and implicit sentiment," in *Proceedings of human language technologies: The 2009 annual conference of the north american chapter of the association for computational linguistics*, pp. 503–511, Association for Computational Linguistics, 2009.

[36] E. Cambria, A. Hussain, C. Havasi, and C. Eckl, "Affectivespace: Blending common sense and affective knowledge to perform emotive reasoning," *WOMSA at CAEPIA, Seville*, pp. 32–41, 2009.

[37] G. Lakoff, *Women, fire, and dangerous things*. University of Chicago press, 2008.

[38] "Swiss center for affective sciences - research material."

[39] A. Balahur, J. M. Hermida, A. Montoyo, and R. Muñoz, *EmotiNet: A Knowledge Base for Emotion Detection in Text Built on the Appraisal Theories*, pp. 27–39. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[40] T. Dalgleish and M. Power, *Handbook of cognition and emotion*. John Wiley & Sons, 2000.

[41] T. Chklovski and P. Pantel, "Verbocean: Mining the web for fine-grained semantic verb relations.," in *EMNLP*, vol. 4, pp. 33–40, 2004.

[42] H. T. Vu, G. Neubig, S. Sakti, T. Toda, and S. Nakamura, "Acquiring a dictionary of emotion-provoking events.," in *EACL*, pp. 128–132, 2014.

[43] P. Pantel and M. Pennacchiotti, "Espresso: Leveraging generic patterns for automatically harvesting semantic relations," in *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, (Stroudsburg, PA, USA), pp. 113–120, Association for Computational Linguistics, 2006.

[44] C. Strapparava and R. Mihalcea, "Semeval-2007 task 14: Affective text," in *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, (Stroudsburg, PA, USA), pp. 70–74, Association for Computational Linguistics, 2007.

[45] C. O. Alm, D. Roth, and R. Sproat, "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*, pp. 579–586, Association for Computational Linguistics, 2005.

[46] L. A. M. Bostan and R. Klinger, "An analysis of annotated corpora for emotion classification in text," in *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 2104–2119, 2018.

[47] C. Strapparava and R. Mihalcea, "Semeval-2007 task 14: Affective text," in *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pp. 70–74, 2007.

[48] A. Chatterjee, K. N. Narahari, M. Joshi, and P. Agrawal, "Semeval-2019 task 3: Emocontext contextual emotion detection in text," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 39–48, 2019.

[49] S. Aman and S. Szpakowicz, "Identifying expressions of emotion in text," in *International Conference on Text, Speech and Dialogue*, pp. 196–205, Springer, 2007.

[50] W. Wang, L. Chen, K. Thirunarayan, and A. P. Sheth, "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)*, pp. 587–592, IEEE, 2012.

[51] Y. Li, H. Su, X. Shen, W. Li, Z. Cao, and S. Niu, "Dailydialog: A manually labelled multi-turn dialogue dataset," *arXiv preprint arXiv:1710.03957*, 2017.

[52] S. M. Mohammad, "Sentiment analysis: Detecting valence, emotions, and other affectual states from text," *Emotion Measurement*, 2015.

[53] S. Buechel and U. Hahn, "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*, pp. 578–585, 2017.

[54] S. M. Mohammad and F. Bravo-Marquez, "Wassa-2017 shared task on emotion intensity," *arXiv preprint arXiv:1708.03700*, 2017.

[55] D. Ghazi, D. Inkpen, and S. Szpakowicz, "Detecting emotion stimuli in emotion-bearing sentences," in *International Conference on Intelligent Text Processing and Computational Linguistics*, pp. 152–165, Springer, 2015.

[56] D. Preoţiuc-Pietro, H. A. Schwartz, G. Park, J. Eichstaedt, M. Kern, L. Ungar, and E. Shulman, "Modelling valence and arousal in facebook posts," in *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pp. 9–15, 2016.

[57] V. Liu, C. Banea, and R. Mihalcea, "Grounded emotions," in *2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII)*, pp. 477–483, IEEE, 2017.

[58] K. R. Scherer and H. G. Wallbott, "Evidence for universality and cultural variation of differential emotion response patterning.," *Journal of personality and social psychology*, vol. 66, no. 2, p. 310, 1994.

[59] H. Schuff, J. Barnes, J. Mohme, S. Padó, and R. Klinger, "Annotation, modelling and analysis of fine-grained emotions on a stance and sentiment detection corpus," in *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pp. 13–23, 2017.

[60] S. M. Mohammad, "# emotional tweets," in *Proceedings of 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*, pp. 246–255, Association for Computational Linguistics, 2012.

[61] S. Baccianella, A. Esuli, and F. Sebastiani, "Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining.," in *LREC*, vol. 10, pp. 2200–2204, 2010.

[62] S. M. Mohammad and P. D. Turney, "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*, CAAGET '10, (Stroudsburg, PA, USA), pp. 26–34, Association for Computational Linguistics, 2010.

[63] S. M. Mohammad and P. D. Turney, "Crowdsourcing a word-emotion association lexicon," vol. 29, no. 3, pp. 436–465, 2013.

[64] C. Strapparava, A. Valitutti, *et al.*, "Wordnet affect: an affective extension of wordnet.," in *LREC*, vol. 4, pp. 1083–1086, 2004.

[65] G. Miller and C. Fellbaum, "Wordnet: An electronic lexical database," 1998.

[66] A. Balahur, R. Steinberger, M. Kabadjov, V. Zavarella, E. Van Der Goot, M. Halkia, B. Pouliquen, and J. Belyaeva, "Sentiment analysis in the news," *arXiv preprint arXiv:1309.6202*, 2013.

[67] J. Staiano and M. Guerini, "Depechemood: a lexicon for emotion analysis from crowd-annotated news," *arXiv preprint arXiv:1405.1605*, 2014.

[68] B. Liu and L. Zhang, "A survey of opinion mining and sentiment analysis," in *Mining text data*, pp. 415–463, Springer, 2012.

[69] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

[70] J. W. Pennebaker, M. E. Francis, and R. J. Booth, "Linguistic inquiry and word count: Liwc 2001," *Mahway: Lawrence Erlbaum Associates*, vol. 71, no. 2001, p. 2001, 2001.

[71] M. M. Bradley and P. J. Lang, "Affective norms for english words (anew): Instruction manual and affective ratings," tech. rep., Citeseer, 1999.

[72] J. Turian, L. Ratinov, and Y. Bengio, "Word representations: a simple and general method for semi-supervised learning," in *Proceedings of the 48th annual meeting of the association for computational linguistics*, pp. 384–394, Association for Computational Linguistics, 2010.

[73] W. Y. Zou, R. Socher, D. Cer, and C. D. Manning, "Bilingual word embeddings for phrase-based machine translation," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1393–1398, 2013.

[74] R. Socher, J. Bauer, C. D. Manning, *et al.*, "Parsing with compositional vector grammars," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, pp. 455–465, 2013.

[75] O. Levy, Y. Goldberg, and I. Dagan, "Improving distributional similarity with lessons learned from word embeddings," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 211–225, 2015.

[76] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[77] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, pp. 3111–3119, 2013.

[78] R. JeffreyPennington, C. C. Manning, J. Pennington, R. Socher, and C. C. Manning, "Glove: Global vectors for word representation," *Proceedings of the Empiricial Methods in ...*, vol. 12, pp. 1532–1543, 2014.

[79] R. Speer, J. Chin, and C. Havasi, "Conceptnet 5.5: An open multilingual graph of general knowledge," *arXiv preprint arXiv:1612.03975*, 2016.

[80] M. Faruqui, J. Dodge, S. K. Jauhar, C. Dyer, E. Hovy, and N. A. Smith, "Retrofitting word vectors to semantic lexicons," *arXiv preprint arXiv:1411.4166*, 2014.

[81] N. Mrkšić, D. O. Séaghdha, B. Thomson, M. Gašić, L. Rojas-Barahona, P.-H. Su, D. Vandyke, T.-H. Wen, and S. Young, "Counter-fitting word vectors to linguistic constraints," *arXiv preprint arXiv:1603.00892*, 2016.

[82] R. Speer and J. Chin, "An ensemble method to produce high-quality word embeddings," *arXiv preprint arXiv:1604.01692*, 2016.

[83] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin, "Learning sentiment-specific word embedding for twitter sentiment classification.," in *ACL (1)*, pp. 1555–1565, 2014.

[84] D. Tang, F. Wei, B. Qin, T. Liu, and M. Zhou, "Coooolll: A deep learning system for twitter sentiment classification," in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pp. 208–212, 2014.

[85] J. Suttles and N. Ide, "Distant supervision for emotion classification with discrete binary values," in *International Conference on Intelligent Text Processing and Computational Linguistics*, pp. 121–136, Springer, 2013.

[86] M. Purver and S. Battersby, "Experimenting with distant supervision for emotion classification," in *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 482–491, Association for Computational Linguistics, 2012.

[87] H. Daumé III, "Frustratingly easy domain adaptation," *arXiv preprint arXiv:0907.1815*, 2009.

[88] K. Roberts, M. A. Roach, J. Johnson, J. Guthrie, and S. M. Harabagiu, "Empatweet: Annotating and detecting emotions on twitter.," in *LREC*, vol. 12, pp. 3806–3813, Citeseer, 2012.

[89] M. Hasan, E. Rundensteiner, and E. Agu, "Emotex: Detecting emotions in twitter messages," 2014.

[90] M. Hasan, E. Rundensteiner, and E. Agu, "Automatic emotion detection in text streams by analyzing twitter data," *International Journal of Data Science and Analytics*, Feb 2018.

[91] R. C. Balabantaray, M. Mohammad, and N. Sharma, "Multi-class twitter emotion classification: A new approach," *International Journal of Applied Information Systems*, vol. 4, no. 1, pp. 48–53, 2012.

[92] S. Wen and X. Wan, "Emotion classification in microblog texts using class sequential rules.," in *AAAI*, pp. 187–193, 2014.

[93] B. Liu, *Web data mining: exploring hyperlinks, contents, and usage data*. Springer Science & Business Media, 2007.

[94] W. Li and H. Xu, "Text-based emotion classification using emotion cause extraction," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1742–1749, 2014.

[95] S. Y. M. Lee, Y. Chen, and C.-R. Huang, "A text-driven rule-based system for emotion cause detection," in *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pp. 45–53, Association for Computational Linguistics, 2010.

[96] S. Li, L. Huang, R. Wang, and G. Zhou, "Sentence-level emotion classification with label and context dependence," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, vol. 1, pp. 1045–1053, 2015.

[97] S. Wang, J. Wang, Z. Wang, and Q. Ji, "Enhancing multi-label classification by modeling dependencies among labels," *Pattern Recognition*, vol. 47, no. 10, pp. 3405–3413, 2014.

[98] J. Xu, R. Xu, Q. Lu, and X. Wang, "Coarse-to-fine sentence-level emotion classification based on the intra-sentence features and sentential context," in *Proceedings of the 21st ACM international conference on Information and knowledge management*, pp. 2455–2458, ACM, 2012.

[99] A. Seyeditabari, S. Levens, C. D. Maestas, S. Shaikh, J. I. Walsh, W. Zadrozny, C. Danis, and O. P. Thompson, "Cross corpus emotion classification using survey data," 2018.

[100] V. López, A. Fernández, S. García, V. Palade, and F. Herrera, "An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics," *Information Sciences*, vol. 250, pp. 113–141, 2013.

[101] Q. Yang and X. Wu, "10 challenging problems in data mining research," *International Journal of Information Technology & Decision Making*, vol. 5, no. 04, pp. 597–604, 2006.

[102] Y. Tang, Y.-Q. Zhang, N. V. Chawla, and S. Krasser, "Svms modeling for highly imbalanced classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 1, pp. 281–288, 2009.

[103] Y. Sun, M. S. Kamel, A. K. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recognition*, vol. 40, no. 12, pp. 3358–3378, 2007.

[104] N. V. Chawla, N. Japkowicz, and A. Kotcz, "Special issue on learning from imbalanced data sets," *ACM Sigkdd Explorations Newsletter*, vol. 6, no. 1, pp. 1–6, 2004.

[105] R. Xu, T. Chen, Y. Xia, Q. Lu, B. Liu, and X. Wang, "Word embedding composition for data imbalances in sentiment and emotion classification," *Cognitive Computation*, vol. 7, no. 2, pp. 226–240, 2015.

[106] A. Balahur, J. M. Hermida, and A. Montoyo, "Detecting implicit expressions of emotion in text: A comparative analysis," *Decision Support Systems*, vol. 53, no. 4, pp. 742–753, 2012.

[107] S. M. Kim, A. Valitutti, and R. A. Calvo, "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*, CAAGET '10, (Stroudsburg, PA, USA), pp. 62–70, Association for Computational Linguistics, 2010.

[108] A. Agrawal and A. An, "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*, pp. 346–353, IEEE Computer Society, 2012.

[109] S. M. Mohammad, "From once upon a time to happily ever after: Tracking emotions in mail and books," *Decision Support Systems*, vol. 53, no. 4, pp. 730–741, 2012.

[110] N. Rey-Villamizar, P. Shrestha, F. Sadeque, S. Bethard, T. Pedersen, A. Mukherjee, and T. Solorio, "Analysis of anxious word usage on online health forums," *EMNLP 2016*, p. 37, 2016.

[111] E. Tromp and M. Pechenizkiy, "Rule-based emotion detection on social media: putting tweets on plutchik's wheel," *arXiv preprint arXiv:1412.4682*, 2014.

[112] E. Tromp and M. Pechenizkiy, "Rbem: a rule based approach to polarity detection," in *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining*, p. 8, ACM, 2013.

[113] A. Bandhakavi, N. Wiratunga, D. Padmanabhan, and S. Massie, "Lexicon based feature extraction for emotion text classification," *Pattern Recognition Letters*, vol. 93, pp. 133 – 142, 2017. Pattern Recognition Techniques in Data Mining.

[114] A. Bandhakavi, N. Wiratunga, P. Deepak, and S. Massie, "Generating a word-emotion lexicon from# emotional tweets," in *Proceedings of the Third Joint Conference on Lexical and Computational Semantics (* SEM 2014)*, pp. 12–21, 2014.

[115] A. Bandhakavi, N. Wiratunga, S. Massie, and D. Padmanabhan, "Lexicon generation for emotion detection from text," *IEEE intelligent systems*, vol. 32, no. 1, pp. 102–108, 2017.

[116] K. Ravi and V. Ravi, "A survey on opinion mining and sentiment analysis: tasks, approaches and applications," *Knowledge-Based Systems*, vol. 89, pp. 14–46, 2015.

[117] A. Seyeditabari and W. Zadrozny, "Can word embeddings help find latent emotions in text? preliminary results," in *The Thirtieth International Flairs Conference*, 2017.

[118] R. Plutchik, *The emotions*. University Press of America, 1991.

[119] S. M. Mohammad and P. D. Turney, "Crowdsourcing a word–emotion association lexicon," *Computational Intelligence*, vol. 29, no. 3, pp. 436–465, 2013.

[120] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1480–1489, 2016.

[121] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," *arXiv preprint arXiv:1712.09405*, 2017.

[122] D. Tang, F. Wei, B. Qin, N. Yang, T. Liu, and M. Zhou, "Sentiment embeddings with applications to sentiment analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 2, pp. 496–509, 2016.

[123] L.-C. Yu, J. Wang, K. R. Lai, and X. Zhang, "Refining word embeddings for sentiment analysis," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 534–539, 2017.

[124] S. Tafreshi and M. Diab, "Sentence and clause level emotion annotation, detection, and classification in a multi-genre corpus," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, 2018.

[125] D. Seal, U. K. Roy, and R. Basak, "Sentence-level emotion detection from text based on semantic rules," in *Information and Communication Technology for Sustainable Development*, pp. 423–430, Springer, 2020.

[126] R. Cheng, J. Zhang, and P. Hu, "Document-level emotion detection using graph-based margin regularization," *Neurocomputing*, 2020.

[127] K. Han, D. Yu, and I. Tashev, "Speech emotion recognition using deep neural network and extreme learning machine," in *Fifteenth annual conference of the international speech communication association*, 2014.

[128] J. Lee and I. Tashev, "High-level feature representation using recurrent neural network for speech emotion recognition," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[129] S.-H. Wang, P. Phillips, Z.-C. Dong, and Y.-D. Zhang, "Intelligent facial emotion recognition based on stationary wavelet entropy and jaya algorithm," *Neurocomputing*, vol. 272, pp. 668–676, 2018.

[130] Y.-D. Zhang, Z.-J. Yang, H.-M. Lu, X.-X. Zhou, P. Phillips, Q.-M. Liu, and S.-H. Wang, "Facial emotion recognition based on biorthogonal wavelet entropy, fuzzy support vector machine, and stratified cross validation," *IEEE Access*, vol. 4, pp. 8375–8385, 2016.

[131] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[132] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Advances in neural information processing systems*, pp. 649–657, 2015.

[133] C. Zhou, C. Sun, Z. Liu, and F. Lau, "A c-lstm neural network for text classification," *arXiv preprint arXiv:1511.08630*, 2015.

[134] J. Y. Lee and F. Dernoncourt, "Sequential short-text classification with recurrent and convolutional neural networks," *arXiv preprint arXiv:1603.03827*, 2016.

[135] M. Abdul-Mageed and L. Ungar, "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)*, pp. 718–728, 2017.

[136] S. Mundra, A. Sen, M. Sinha, S. Mannarswamy, S. Dandapat, and S. Roy, "Fine-grained emotion detection in contact center chat utterances," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 337–349, Springer, 2017.

[137] H. Al-Omari, M. A. Abdullah, and S. Shaikh, "Emodet2: Emotion detection in english textual dialogue using bert and bilstm models," in *2020 11th International Conference on Information and Communication Systems (ICICS)*, pp. 226–232, IEEE, 2020.

[138] M. Polignano, P. Basile, M. de Gemmis, and G. Semeraro, "A comparison of word-embeddings in emotion detection from text using bilstm, cnn and self-attention," in *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, pp. 63–68, 2019.

[139] C. Huang, A. Trabelsi, and O. R. Zaïane, "Ana at semeval-2019 task 3: Contextual emotion detection in conversations through hierarchical lstms and bert," *arXiv preprint arXiv:1904.00132*, 2019.

[140] J.-Á. González, L.-F. Hurtado, and F. Pla, "Elirf-upv at semeval-2019 task 3: Snapshot ensemble of hierarchical convolutional neural networks for contextual emotion detection," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 195–199, 2019.

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

[142] Z. Wang, "Text emotion detection based on bi-lstm network," *Academic Journal of Computing & Information Science*, vol. 3, no. 3, pp. 129–137, 2020.

[143] J. Á. González, L. Hurtado, F. Pla, and J. Moncho, "Elirf-upv at tass 2020: Twilbert for sentiment analysis and emotion detection in spanish tweets," *Proceedings of TASS*, 2020.

[144] M. Karna, D. S. Juliet, and R. C. Joy, "Deep learning based text emotion recognition for chatbot applications," in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, pp. 988–993, IEEE, 2020.

[145] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[146] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[147] R. Speer, J. Chin, and C. Havasi, "ConceptNet 5.5: An open multilingual graph of general knowledge," pp. 4444–4451, 2017.

[148] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," in *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.