Detecting Emotion in Text

Kaitlyn Mulcrone
University of Minnesota, Morris
mulcr002@morris.umn.edu

ABSTRACT

Advancements in textual analysis have allowed the area of emotion detection to become a recent interest in computational linguistics. This paper presents a general overview of the many diverse approaches that recognize emotion in text. It goes over the components the approaches have in common and then goes on to describe the different types of algorithms and results that been developed to date. Additionally, the paper covers the challenges and concerns of the field and what will come of emotion detection in the future.

Categories and Subject Descriptors

I.27 [**Natural Language Processing**]: Text analysis; H.3.1 [**Content Analysis and Indexing**]: Linguistic processing

General Terms

Algorithms

Keywords

subjective language, sentiment analysis, emotional lexicon, annotation, emotion labeled dataset, categorical classification, dimensional classification, precision, recall, f-score

1. INTRODUCTION

The Internet is a place of textual shared knowledge that contains large collections of documents that can provide various information [13]. Fields such as Computational Linguistics, Natural Language Processing (NLP), and Affective Computing have taken an interest in these collections, or corpuses, of texts. Out of the many textual analysis tasks pursued, a growing area of interest is the automatic detection of emotion. Being an important element in understanding human experience and communication, emotions have been studied in the psychological and behavioral sciences disciplines [9]. Methods of emotion detection are now possible with the foundations of textual analysis; however, the lack of consistency in approaches creates challenges when trying to compare methods.

This paper gives a general overview of the different approaches of detecting emotions. It provides background on foundational fields of textual analysis in Section 2, and describes similar elements that appear in many emotion recognition methods in Section 3. Section 4 presents an overview of the more common approaches and their results. The paper concludes in Section 5, which focuses on some of the ongoing challenges in the field and what the field will have to accommodate in the future.

2. BACKGROUND

Emotion detection is a newer area of textual analysis, and therefore, has weaker standard methods. However, there are other areas of textual analysis that have established standard methods and are also beneficial to emotion detection studies. The areas of subjectivity and sentiment analysis have created foundational methodologies that benefit many natural language processing (NLP) applications including multi-document summary systems, flame recognition, and question-answering systems. Further information on these applications can be found in (Wiebe, Wilson et al. 2004)[12].

2.1 Subjectivity

Emotion detection is a NLP application that benefits from being able to distinguish subjective from objective language. *Subjective language* is language used to express opinions, evaluations, emotions, or speculations [12]. Objective language is unbiased and not influenced by the writer's opinions or tastes. Both types of language are useful in text analysis: Subjective language is useful for automatic subjectivity analysis and objective language is useful for information extraction. Emotions are expressed in subjective language so it would appear that subjectivity analysis is the only area beneficial in emotion detection. However, recent studies conclude that a combination of these two areas is effective in producing accurate results in subjectivity analysis [11].

Automatic subjectivity analysis classifies sentences or documents by looking for subjective elements or expressions [11, 12]. Subjective elements are the linguistic expression of opinions, emotions, and speculation and are either single words or longer complex expressions. Although there are subjectivity lexicons and dictionaries that provide established subjective elements, the list is nowhere near complete. Additionally, some elements can be used subjectively or objectively, leading to ambiguity.

UMM CSci Senior Seminar Conference, April 2012 Morris, MN.

Lists of Basic Emotions	
Ekman	anger, disgust, fear, joy, sadness, and surprise
Izard	anger, contempt, disgust, distress, fear, guilt, interest, joy, shame, and surprise
Plutchik	anger, anticipation, disgust, fear, joy, sadness, surprise, and trust

Figure 1: These are common lists of emotions used in emotion detection methods [4, 3, 6].

2.2 Sentiment Analysis

Sentiment analysis methods are the most similar to emotion detection methods. *Sentiment analysis* seeks to classify documents by sentiment rather than topic [8]. In most studies, sentiment is classified as either positive or negative. Subjectivity has been a beneficial tool for sentiment analysis. [Pang and Lee, 2004] trained three standard classification algorithms on both a subjectivity dataset and a basic subjectivity detector in order to extract the subjective sentences from the reviews before running the algorithm. Additionally, the amount extracted was set manually and tested for the most beneficial extraction. The variation in the amount extracted was tested by taking the N most subjective sentences from the original review — where N is how many subjective sentences are being extracted.

3. EMOTION DETECTION

Emotion detection approaches use or modify concepts and general algorithms created for subjectivity and sentiment analysis. There are many approaches that are being explored but there are no fore-running methods. However, there are a few similarities that appear in a majority of the approaches.

3.1 Annotation

To check the accuracy of any emotion detecting algorithm, the results need to be compared to a human-labeled text. The process in which humans manually label a text is called *annotation*. Annotation can be done on multiple levels: word, sentence, paragraph, section, or even the entire document. In emotion detection studies, text is annotated on polarity, emotion, and intensity. When annotating on polarity, text is labeled with positive, negative, or neutral emotion. Text annotated on emotion is labeled as one of the listed emotions defined in the study. The most common lists of emotions are presented in Figure 1. Additionally, some studies annotate the text by labeling the intensity of the emotion. Intensity is based on a numeric scale, but there are no standards for this type of annotation.

In general, studies either use pre-annotated datasets to test an algorithm or undergo a small annotation process. In the latter case, authors select annotators who are qualified to label emotion in text, such as psychologists, or use a test or training session to establish reliable annotators [10, 1]. Once annotators are qualified, the text is annotated to the given specification and level of analysis. Sometimes annotators are given an additional word list that consists of words from the original text. These lists help determine which words are attached to a specific emotion and which vary by context.

When the annotation process is complete the agreement among the annotators is calculated. The agreement between two annotators is calculated by finding observed and expected agreement. Observed agreement (A_o) calculates how much two annotators agreed on the individual annotations that each annotator made [10, 1].

$$A_o = \frac{1}{I} \sum_{i \epsilon I} arg_i$$

In this equation, i is an annotated item in a set, and arg_i outputs a 1 if the two coders assign i to the same category and 0 if it is assigned to different categories. Then I is the total number of items in the set.

Expected agreement (A_e) calculates how much the annotators are expected to agree if they each randomly assigned emotions to the sentences. This value is calculated based on their frequency of assigning emotions [10, 1].

$$A_e = \frac{1}{I^2} \sum_{k \epsilon K} n_{c_1 k} n_{c_2 k}$$

In this equation, n_{ck} is the number of items assigned by annotator c to category k, and K is the set of all categories used by both annotators.

The overall agreement between two annotators is known as a statistical measure called a kappa value. A kappa value of 1.0 is total agreement and 0.0 is complete random labeling, or the accuracy of flipping a coin for each labeling decision. Kappa value is calculated with the following equation:

$$k = \frac{A_o - A_e}{1 + A_e}$$

This general calculation formula for the inter-annotator agreement (IAA) measure, provided by (Volkova et al. 2010) [10], is used because humans are not always perfect. Subtracting the statistically expected agreement subtracts away the chance that the annotators randomly assigned an item to an emotion. The kappa value is then a more realistic percentage of how much two annotators agree then just calculating observed agreement.

Consider the following: There are 100 sentences that need annotation, two annotators, and four emotion categories; anger, fear, joy, and sadness. Annotator one tagged all 100 sentences; 20 anger, 15 fear, 40 joy, and 25 sadness. Annotator two also tagged all 100 sentences; 25 anger, 20 fear, 30 joy, and 25 sadness. Out of both annotations they agreed on 16 anger sentences, 13 fear sentences, 27 joy sentences, 23 sadness sentences. In this situation, the expected agreement is ((20*25) + (15*20) + (40*30) + (25*25)) / 100^2 = 0.26. The observed agreement is (16+13+27+23) / 100 = 0.80. Plug these two values in and the kappa value is (0.80 - 0.26) / (1 - 0.26) = 0.73.

3.2 Emotional Lexicon

The first step in detecting emotion in text is discovering keywords or phrases that associate with emotions. These words are important to training machine-learning algorithms. A subsection of emotion recognition studies involves compiling and improving ways to compile these types of collections.

A list of emotions and words that express each emotion is called an *emotional lexicon* [6]. In general, these lists start with identifying seed words, or words that highly associate with one emotion, and expand by using synonyms. Currently, there are not extensive or lengthy lexicons in any language but there are a few that generated a sufficient start.

The WordNet Affect Lexicon is a manually created collection of words, each annotated with the emotions they evoke. The creation process involved annotating a few seed words with Ekman's six basic emotions then expanding the collection by marking the WordNet synonyms of each word with the same emotion. The full list reached a few hundred words.

The General Inquirer is an emotional lexicon that classifies words into a larger number of categories. The collection contains 11,788 emotion labeled words and 182 word tags, which include positive and negative semantic orientation and affect categories like pleasure, arousal, feeling, and pain that have not extensively been analyzed yet.

A newer lexicon is the NRC Emotion Lexicon, which was created using Amazon's Mechanical Turk, a crowd sourcing online service that allows a person to present a task users can complete for a small amount of money [6, 5]. (Mohammad and Turney, 2010) created a generic multiple-choice questionnaire that could be applied to the target words they wanted annotated. For each word users were asked about the polarity of the word and how well it expressed Plutchik's eight prototypical emotions. An additional question was added to weed out users that were not familiar to the word; they were asked to identify the word (out of four choices) that was closest in meaning to the target word. If a user answered that question wrong, that particular questionnaire did not contribute to the results. The results showed high annotator agreement on about 2000 target words.

Even though a growing emotion lexicon would be beneficial in detecting emotion, an annotated collection of words and phrases would only increase detection accuracy to a certain extent. Emotional lexicons by themselves are not successful in classifying sections of texts to their appropriate emotion. In fact, most of the time emotion is not expressed through the use of emotion-bearing words [2]. The majority of words, or the synonyms of a word, fall under more than one emotional classifier [6]. These words have unclear emotional meaning making the emotion label change by context or by the words surrounding it.

3.3 Emotion Labeled Datasets

Emotion labeled datasets are blocks of text that have been annotated with emotion tags. Manually annotating datasets of text is expensive and time consuming. However, because comparing results to annotated texts is the most stabilized method of checking the accuracy of an algorithm, annotated datasets have been established and consistently used throughout emotion detection studies.

A common dataset, used in many emotion detection studies, is SemEval 2007-Task, an affective text that consists of newspaper headlines. The annotations are labeled with Ekman's six basic emotions along with a neutral category. Additionally, the dataset allowed one sentence to be tagged with multiple emotions [3]. The dataset is composed of 1,250 annotated headlines that is split between a developmental set of 250 headlines and a test set of 1,000 news headlines.

Another annotated dataset is the International Survey on Emotion Antecedents and Reactions (ISEAR). The ISEAR is a compilation of 7,666 sentences provided by 1,096 culturally divergent participants who were questioned about experiences and reactions that related to the emotions of anger, disgust, fear, joy, sadness, and guilt.

The third emotion-labeled dataset is fairy tales. The fairy tales collection is complied of stories by the authors B. Potter, H. C. Anderson, and the Brother's Grimm, with stories annotated on the sentence-level. Varying annotation processes have been conducted by Alm that provide a larger set of specific emotions. A dataset of 1580 sentences compiled in 2005 is labeled with Izard's set of ten basic emotions; and a dataset, including 176 stories, complied in 2009 is labeled with five emotion classes: angry-disgusted, fearful, happy, sad, and surprised. The latter dataset is composed of only sentences that have a high kappa value.

Finally, there are blog datasets. The nature of some weblogs, like LiveJournal, allow the blogger to attach a mood or an emotion to an entry. The data is then self-annotated by the author and annotated at the entry-level as opposed to sentence-level. Blog entries relevant to a particular study are initially narrowed down from the list of 132 provided moods to the entries labeled with the specific emotions evaluated in the study. The remaining set of entries is then picked based on the intensity an entry connects to one specific emotion, while making sure that there is an equal number of entries for each emotion and that there is author variety. One corpus of LiveJournal entries compiled by Mishne is available for use and contains 815,454 entries [3].

Even though there are annotated datasets out there to test algorithms on, the necessity of an annotated dataset limits the text used in emotion detection studies, especially for machine-learning algorithms that require a large annotated datasets for training.

4. EMOTION DETECTION CASE STUDY

One emotion detection study, by (Kim, Valitutti, et al. 2010), attempted to recognize emotion by testing two different classification methods: categorical classification and dimensional classification.

Categorical classification marks text under discrete categories, such as the many lists of basic emotions mentioned throughout this paper. These methods go as far as establishing emotional classes, or sets of emotions that associate with specific texts. An example is focusing on the emotions boredom, delight, confusion, and frustration when looking at texts that discuss teaching or education. The benefit of this method is its clear-cut emotional tags that make the results easy to understand.

Dimensional classification represents emotions in a dimensional form, meaning that they are defined by two- or three-dimensional space. As opposed to labels, dimensional classification uses emotional states, for instance a valence dimension, or positive versus negative emotions (the common states of sentiment analysis) and an arousal dimension, or excited versus calm emotions. These emotional states act as two ends of a scale.

4.1 Vector Space Model

Vector Space Model (VSM) is an approach that utilizes categorical classification. The process begins by representing the dataset dimensionally through a matrix of co-occurrence frequency vectors. The rows represent words and the columns can represent sentences, paragraphs, or documents. (Kim, Valitutti, et al. 2010) did their study with the columns representing documents. Therefore, the row vectors represent a term and its relation to each document, and the column vectors represent a document and its relation to each term. VSM weighs these frequencies using the *tf-idf* weighting schema.

The *tf-idf* score is the weight of each word in terms of its importance within the dataset of documents. The score is broken down into tf and idf. The tf stands for term frequency and is the frequency of a term within a document. The equation for calculating tf is as follows:

$$\text{tf} = \frac{n_{t,d}}{k_d}$$

In this equation, $n_{t,d}$ is the number of times the term, t, appears in the document, d, and k_d is the total number of words in the document, d.

The second part of the score is idf, which stands for inverse document frequency and is the importance of the term based on its rarity. This value tells if a word is common or rare in the corpus. The equation for calculating idf is as follows:

$$\text{idf} = \log_{10} \frac{|D|}{|D_t|}$$

In this equation, D is the total number of documents in the corpus and the D_t represents all the documents in which the term, t appears. Once the tf and idf values have been calculated, the *tf-idf* score is calculated by multiplying the two values together: tf-idf $=$ tf $*$ idf.

The *tf-idf* score is important because it prevents bias towards a large corpus and provides the importance of each word. If a term appears in more documents, then the ratio inside the idf's log calculation becomes closer to 1 while the actual idf value and *td-idf* score becomes closer to 0. Consider the following example: In a dataset of ten million documents the word *badger* appears in one thousand of them. This dataset has a document of one hundred words where the word *badger* appears three times. Using the previously defined equations, the term frequency for *badger* is (3/100) $=$ 0.03 and the inverse document frequency is $\log_{10}(10,000,000/1,000) = 4$. Then the *tf-idf* score is (0.03 * 4) $=$ 0.12.

After calculating the pair *tf-idf* scores, the co-occurrence frequency matrices are constructed. Each element of the matrix has the corresponding *tf-idf* score vector. Then the columns represent the dimensions that define the hyperspace and the rows represent individual points within that space. However, these term-by-document matrices are very sparse, filled with many unnecessary zeros, and the high amount of dimensions is difficult to manage for most algorithms. To create a less sparse representation, three different dimension reduction methods are tested: Latent Sentiment Analysis (LSA), Probabilistic LSA (PLA), and Non-Negative Matrix Factorization (NMF). Reducing the matrices not only makes them more manageable for algorithms, but helps underline the semantic text and cut out unimportant data.

4.2 Reduction Methods

4.2.1 Latent Sentiment Analysis

LSA is a method used in many textual analysis approaches. LSA decomposes the term-by-document matrix using singular vector decomposition (SVD). SVD decomposes the matrix into a product of three matrices.

$$X = U \Sigma V^T$$

X is the original term-by-document matrix, U represents the term-by-term matrix in which the columns are known as the left singular vectors of X, and V^T (V^T being the transpose of V) represents the document-by-document matrix in which the columns are known as right singular vectors of X. These matrices are orthogonal, meaning the matrix multiplied by its transpose equals the identity matrix. The Σ is a nonnegative diagonal matrix of singular values.

SVD takes one more step with its decomposition: It selects the largest singular values. They will be represented by the letter k, along with their corresponding singular vectors from U and V^T. This step computes a new approximation matrix of rank k with the smallest error. The following is the final representation of SVM's decomposition of the matrix:

$$X = U \Sigma V^T \approx U_k \Sigma_k V_k^T = X_k$$

where X_k is the closest matrix of rank k to the original matrix.

4.2.2 Probabilistic Latent Sentiment Analysis

The PLSA method is a very different approach from LSA; it defines proper probability distributions and the reduced matrix does not contain negative values. The distributions help find the probability that the documents and emotions are associated, and the probability that the terms and emotions are associated. These associations help reduce the matrix through this equation [4]:

$$P(d, t) = \sum_z P(z) P(t|z) P(d|z)$$

where z is discrete emotional category, and as with LSA, d is the elements of the document vectors and t is the elements of the terms vectors. The $P(t|z)$ matrix is emotion-specific term distributions and $P(d|z)$ matrix is the emotion-specific document distributions.

4.2.3 Non-negative Matrix Factorization

The other dimensional reduction method is NMF. The method name states that this method only works on non-negative matrices. NMF decomposes the original matrix into a product of two, also non-negative, matrices: $X \approx TD$ [4]. The original matrix is represented by X. T represents the matrix of the term vectors and D represents the matrix of the document vectors. Multiplied together, the product of the two matrices is an approximation of the original matrix. NMF finds the two matrices, T and D, by minimizing the Frobenius norm of the difference between the original and the reduced matrix:

$$||A|| = \sqrt{\sum_{x=1}^{n} \sum_{y=1}^{m} a_{x,y}^2}$$

The difference between the original and the reduced matrix is A, which is also a term-by-document matrix. The x and y point to the position of the vector in the matrix, n is the total number of terms, and m is the total number of

Methods	SemEval			ISEAR			Fairy Tales		
	Prec.	Rec.	F1	Prec.	Rec	F1	Prec.	Rec	F1
MCB	0.077	0.250	0.118	0.100	0.250	0.143	0.102	0.250	0.145
LSA	0.363	0.348	0.340	0.484	0.282	0.228	0.662	0.640	0.630
PLSA	0.189	0.282	0.219	0.260	0.317	0.270	0.282	0.307	0.280
NMF	**0.523**	**0.506**	**0.505**	0.461	0.258	0.166	**0.747**	**0.731**	**0.733**
VAD	0.466	0.422	0.386	**0.528**	**0.417**	**0.372**	0.530	0.404	0.419

Table 1: Overall average results for the three datasets using precision, recall, and f-score; best results are in bold [4].

documents. The goal here is to minimize A or the square root of the sum of the squares of all A's elements $(a_{x,y})$.

4.2.4 Categorical Classification Result

After the matrices are reduced with one of the three dimensional reduction methods, they abstractly represent the vocabulary of the document. To then assign an emotion to each sentence, the similarity between the sentences and the emotional synset vectors is calculated. The syneset is a list of synonyms or words related to an emotion. These syneset vectors are represented in a subgraph similar to the reduced matrix. The input text and the emotional synset vectors are compared using cosine similarity, which calculates the emotion that "fits" best with the input text by looking at the cosine angle between the vectors. To make sure that the similarity evokes a strong emotional analogy, a threshold value is added. If the text sentence does not exceed the threshold the text is labeled as neutral. The similarity method checks each emotional synset and gives the emotion with maximum similarity as the result.

4.3 Valence-Arousal-Dominance

The VAD approach utilizes the dimensional classification of emotions. The method developed with the help of the ANEW lexicon, which asked subjects to report on their affective experience toward a word in a three dimensional representation. The three dimensions are valence, arousal, and dominance (VAD). Valence is equivalent to emotional polarity, or positive versus negative emotions. Arousal is similar, except the two sides of the spectrum are excited versus calm. The dominance dimension judges whether or not the subject feels in control of the situation or not. Each dimension is represented as a number between 0 and 10, where a word of neutral emotion would be represented as (5,5,5). All words in ANEW are listed as [4]:

$$\overline{w} = (valence, arousal, dominance)$$

Unfortunately, ANEW did not annotate a sufficient amount of words for this approach, so a larger list was created by converting the synonyms of the annotated words, found in WordNet Affect, into VAD space.

This approach uses these annotated words and their synonyms to weigh sentences on an emotional plane. Sentences are classified to the emotion of smallest distance in the VAD space. The VAD value of each sentence is calculated by averaging the VAD values of each word in the sentence. The VAD value of each emotion is calculated by the same averaging equation but instead of summing up the VAD values of each word of the sentence, it sums up the VAD values of each synonym of the emotion word [4].

$$\overline{value} = \frac{\sum_{i=1}^{n} \overline{w}}{n}$$

In this equation, n is the total number of words in the sentence when finding the value of a sentence, and is the total number of the emotion word's synonyms when finding the value of an emotion.

Once all VAD values are calculated, each sentence is labeled with the emotion whose VAD value is closest to that sentence. There are two immediate problems with this approach. One, because many words change their meaning when used in different contexts, this is a naive approach. Each word is annotated based on the general feeling evoked by a word and not how it changes with context. The second problem is that if opposing emotion words are found in a sentence it is possible that the average VAD value of a sentence falls into a third emotional state.

4.4 Results

The (Kim, Valitutti, et al. 2010) study analyzes two approaches that create a total of four methods. The overall results of these methods are shown in Table 1. The methods classify SemEval, ISEAR, and fairy tales into four emotion categories: anger, fear, joy, and sadness. Each input sentence is given a signal emotion label. Majority Class Baseline (MCB) is used as a baseline algorithm to help analyze the results. MCB labels each corpus with the emotion that has the highest amount of sentences associated with it. In this case, SemEval and fairy tales have 'joy' the as majority class and ISEAR has 'anger' as the majority class.

Like most emotional detection approaches, results are presented with the common measures of precision, recall, and f-score. In all measures, 1.0 is the highest value and 0.0 the lowest. *Precision* is the number of relevant instances retrieved by the given algorithm over the total number of instances retrieved by the algorithm. In this case, precision is the number of correctly labeled sentences retrieved by the algorithm divided by all the sentences retrieved by the algorithm [11]. *Recall* is the number of relevant instances retrieved by the given algorithm over the total number that should have been returned. In this study, recall is the number of correctly labeled sentences retrieved by the algorithm divided by all the sentences annotated as correct [11]. After precision and recall are calculated, the values are used to calculate the *f-score*, or the harmonic mean of precision and recall that functions as a weighted average equation.

$$\text{f-score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

The results shown in Table 1 suggest that using VSM with the NMF method yields the best results. The authors do not discuss why this might be the case; however, they point out that because NMF did not yield the best results for every dataset, they cannot conclude it as the best performer. Another interesting result in this paper is that the precision, recall, and f-score are generally higher in fairy tales than in the other datasets. The authors believe that the methods work better with this dataset because of how fairy tales are written; the sentences of fairy tales tend to use more emotional words, and the sentences themselves are longer.

The paper also examined the word frequencies generated by the fairy tale dataset and found certain emotions had unexpected words in their list. For instance, the word 'good' appears in many of the sentences tagged fear, and the word 'cried' appears in many of the sentences tagged anger, fear, and joy. Finding unexpected words in sentences of a particular emotion suggests that emotion detection cannot by determined just using a lexicon. Instead the structure of the sentence also attributes to labeling emotion.

5. CONCLUSIONS AND FUTURE WORK

Emotion detection has a promising future. Although not enough time has passed to have established standards in the field, there is some consistency between the approaches, and the algorithms are continuing to increase in accuracy.

A glaring issue with this field is the inability to compare a majority of the algorithms. Few studies have taken the time to compare different algorithms on multiple datasets and start defining standards to steer this field away from its current free reign. Many of the studies hope, in the future, to adjust their algorithms so they apply to a more general textual dataset [4].

One forthcoming issue that has been briefly addressed is the emerging and changing messages language, or the language developing in instant messaging, chat rooms, and text messages. The rise in virtual communities has created a new textual language that is based on communication that is short and speedy [7]. The use of emoticons and acronyms are a new form of language that textual analysis will have to learn to accommodate.

There are many advantages in being able to detect emotion in text. Some of the proposed applications include: the ability to search based on emotions; the ability to study how emotional expression changes over time, between genders, or between ethnic groups; and the capability to gather the overall emotion of a specific text [5]. In addition to being able to create some applications, the ability to detect emotion in text can increase human-computer interaction. If the computer could tell a person's mood or emotional state, it would be able to switch to an accommodating form of interaction.

6. ACKNOWLEDGMENTS

Thanks to: Elijah Mayfield for taking time out of his busy schedule to read my paper and providing in-depth responses to all my questions; Elena Machkasova for being my advisor on this paper and helping me flesh out some of the paper's confusing equations; Will Martin for sharing his knowledge of useful LaTeX techniques; and Kate Saddison for being a harsh editor.

7. REFERENCES

[1] R. Artstein and M. Poesio. Inter-coder agreement for computational linguistics. *Comput. Linguist.*, 34(4):555–596, Dec. 2008.

[2] A. Balahur, J. M. Hermida, and A. Montoyo. Detecting implicit expressions of sentiment in text based on commonsense knowledge. In *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, WASSA '11, pages 53–60, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[3] F. Keshtkar and D. Inkpen. A corpus-based method for extracting paraphrases of emotion terms. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, CAAGET '10, pages 35–44, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[4] 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, pages 62–70, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[5] S. Mohammad. From once upon a time to happily ever after: tracking emotions in novels and fairy tales. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, LaTeCH '11, pages 105–114, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[6] 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, pages 26–34, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[7] A. Neviarouskaya, H. Prendinger, and M. Ishizuka. Analysis of affect expressed through the evolving language of online communication. In *Proceedings of the 12th international conference on Intelligent user interfaces*, IUI '07, pages 278–281, New York, NY, USA, 2007. ACM.

[8] B. Pang and L. Lee. A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[9] C. Strapparava and R. Mihalcea. Learning to identify emotions in text. In *Proceedings of the 2008 ACM symposium on Applied computing*, SAC '08, pages 1556–1560, New York, NY, USA, 2008. ACM.

[10] E. P. Volkova, B. J. Mohler, D. Meurers, D. Gerdemann, and H. H. Bülthoff. Emotional perception of fairy tales: achieving agreement in emotion annotation of text. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, CAAGET '10, pages 98–106, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[11] J. Wiebe and E. Riloff. Finding mutual benefit between subjectivity analysis and information extraction. *Affective Computing, IEEE Transactions on*, 2(4):175 –191, oct.-dec. 2011.

[12] J. Wiebe, T. Wilson, R. Bruce, M. Bell, and M. Martin. Learning subjective language. *Comput. Linguist.*, 30(3):277–308, Sept. 2004.

[13] Y. Wu and F. Ren. Improving emotion recognition from text with fractionation training. In *Natural Language Processing and Knowledge Engineering (NLP-KE), 2010 International Conference on*, pages 1 –7, aug. 2010.