CSC 594 Final Project Paper

Zack Larsen

[zlarsen4@gmail.com](mailto:zlarsen4@gmail.com)

May 25, 2017

**Abstract**

*Personal communication has shifted increasingly towards the digital realm in the years since the introduction of mobile computing, smart phones, and social media networks. There exists a wealth of natural language data available on the web for companies, individuals, and organizations to make use of. In this paper, I will attempt to provide a summary of research currently being done to extract value from these data sources by using state-of-the-art natural language processing techniques such as word2vec and GloVe neural network approaches (skip gram negative sampling, or sgns) as well as multi-class sentiment classification techniques for more informative analyses of consumer sentiment online.*

**1 Introduction**

Naïve natural language processing systems can lead to disastrous results for companies. According to an article on United Airlines [4], “a six-year-old story about the company’s 2002 bankruptcy filing gained new life on the Internet, triggering a cascade of stock sales. In a matter of about 12 minutes more than $1 billion in stock market value evaporated.” This happened because no humans were consulted before trades were automatically placed on the stock market in reaction to this news story, based on the work of Prof. Andrew W. Lo of the M.I.T. Sloan School of Management. This story highlights the perils of misclassification of sentiment and the breakdown of natural language understanding in intelligent systems that can be deployed in financial systems and other areas of our economy where superhuman speed and scale are needed to react to complex systems.

If we wish to benefit from these kinds of NLP and NLU artificial intelligence systems, we need to ensure their accuracy and reliability, and work needs to be done to address some of their current limitations. In the 1990’s, the rise of the statistical approaches to natural language processing (such as text classification) drew research attention away from natural language understanding. After realizing that the statistical approaches have their limitations, research has returned to natural language understanding and researchers have chosen to return to more rule- and structure-based approaches to computational linguistics that incorporate richer knowledge of linguistic semantics. In this paper, I seek to survey several research articles to give the reader an understanding of more recent approaches and motivate a discussion of where research needs to head in the future to derive the greatest benefit from advances in this domain of artificial intelligence research.

Some of the areas of current research in natural language processing that may be used to advance the field are distributed vector representations of words, multiclass sentiment and emotional classification, and the use of non-verbal communication in text- and oral-based interfaces for intelligent agents that respond to human users’ emotions. Broadly speaking, these are the areas of “natural language understanding” and “sentiment classification.”

Natural language understanding is defined by Wikipedia as “a subtopic of natural language processing in artificial intelligence that deals with machine reading comprehension. NLU is considered an AI-hard problem.” The reason that it is considered an “AI-hard” problem is that there are unknown features that must be appropriately handled, and it is not feasible for these to be handled by hard-coded rules or formal logic.

Word2vec uses a distributed representation of words in a vector space to generate high-quality word embeddings. It uses shallow neural networks to train a skip gram model of the context surrounding words. This skip gram model assumes that linguistic structure and “meaning” are linked to the window of other words that surround a given word. The ability of models like word2vec to capture the *semantic meaning* of words in an *unsupervised* fashion allows such a statistical/ probabilistic approach to absorb more informative content from natural language than previous statistical or computational approaches. Because this approach is unsupervised (or semi-supervised in some cases) and heuristic approximations like noise-contrastive elimination can be used to reduce training time, word2vec and related models have a great potential to scale up and be deployed in a variety of applications.

GloVe (Global Vector embeddings) is a very recent approach to embeddings tasks that attempts to combine the strengths of traditional and recent models, and it achieves state-of-the-art performance on analogy tasks. It is not comprehensively covered in this paper but serves as an example of areas for future research in the area of natural language understanding.

Sentiment classification is a specific subtask of text classification which aims to identify the “sentiment polarity” of the opinion in an article, product review, text message, etc. Here, the “sentiment polarity” refers to whether the text is “positive” or “negative” (and, potentially, “neutral”). This can be very useful to identify general trends in real time for companies to monitor how their customers are reacting to news or products, or for public figures to monitor public opinion of their campaigns. Existing implementations mostly focus on this “polarity”, but more value can be extracted from text by analyzing more fine-grained aspects of the text such as emotions, which will be detailed later in this paper.

Most text-based communication has considerable ambiguity because meaning and semantics in human language depend on non-verbal content such as body language, prosodic (audio/tone of voice) cues, eye contact, and facial expressions. To improve the performance of sentiment classification systems, non-textual sources of information such as emojis can be studied.

**2 Background and summary of current research**

# Word2vec

Word2vec is a distributed vector space model for words that offers superior performance to traditional bag-of-words (bow) models in various tasks. According to the researchers at Google who first proposed word2vec, “learned vectors from the neural network explicitly encode linguistic regularities and patterns, and these can be represented as linear translations.”[2] These *linear translations* that the authors mention are important because there is a kind of relational algebra that can be applied to these vectors to make interesting comparisons and analogies, such as the (now) ubiquitous equation from word2vec: “queen minus woman plus man = king.”

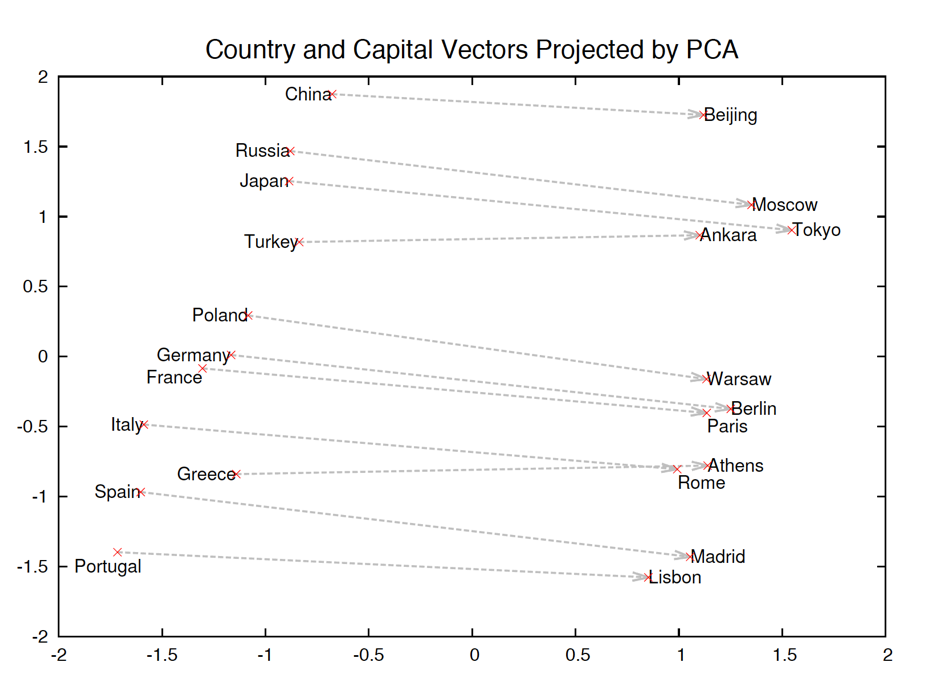


Figure 1: Two-dimensional PCA projection of 1000-dimensional skip-gram vectors of countries and their capitals.

The figure above shows a plot of country and capital vectors after being dimensionally reduced via principal components analysis. It is visually apparent that the countries tend to vertically align in some subspace of this plane, as do the capitals. Additionally, each country’s respective capital tends to lie on a very similar latitude such that few vectors cross paths. This embedded organization allows for algebraic/geometric manipulation of concepts and words that can produce meaningfully rich results, as we will see later in table 1.

The ability of word2vec to “automatically organize concepts and learn implicitly” these kinds of relationships is a crucial step forward in natural language understanding because these rules do not need to be hard-coded as logic into the system. Rather, these concepts can be generated without any instruction from programmers. As mentioned in the introduction, NLU is an “AI-hard” problem because of the difficulty of handling unseen instances and unknown features. The unsupervised nature of word2vec addresses this difficulty in NLU systems. This approach is one step closer to the way humans learn abstract concepts, and it is an exciting advance in the field.

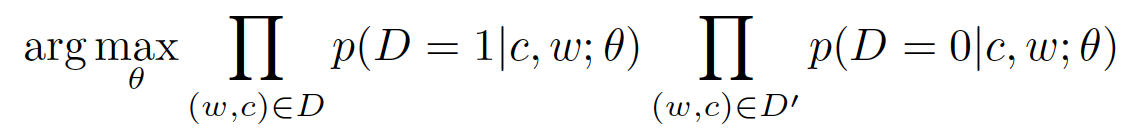
The author of [9] explains that the reason that the vectors capture the semantic meaning of words is “…that the vectors adhere surprisingly well to our intuition. For instance, words that we know to be synonyms tend to have similar vectors in terms of cosine similarity and antonyms tend to have dissimilar vectors. Even more surprisingly, word vectors tend to obey the laws of analogy.” In other words, the vectors perhaps organize themselves in a similar fashion to the way humans do, so the output of the embeddings naturally makes sense to humans in terms of meaning.

Prior to word2vec, a popular method to establish semantic relationships between words was Latent Semantic Analysis (LSA), first proposed by Dumais et. al 1988 [8]. In this approach, it is assumed that words occurring together frequently in similar text will be meaningfully related (the distributional hypothesis). In this technique, a matrix populated by word frequency counts is created and then decomposed via singular value decomposition (SVD), a dimensionality reduction technique. Once SVD is performed, the remaining columns are compared using cosine similarity to determine which words are meaningfully related. A disadvantage to LSA is that it does not scale well, whereas word2vec does scale very well. Additionally, LSA employs a more traditional bag-of-words (bow) representation that does not offer the rich semantic relationships that word2vec explicitly encodes by using a window of context (skip-grams).

A key advantage to the word2vec implementation is that it is very efficient, allowing the researchers to train the model on 33 billion words in less time than it took researchers such as Geoff Hinton from the University of Toronto (an eminent researcher in neural networks and artificial intelligence) to build models less than half the size of word2vec. As the authors of the seminal paper note, “training of the Skipgram model…does not involve dense matrix multiplications. This makes the training extremely efficient: an optimized single-machine implementation can train on more than 100 billion words in one day.” [2]

Frequently occurring words are omitted from the training phase of the word2vec implementation, in much the same way that more frequently occurring words are given smaller weights in the term frequency-inverse document frequency approach often adopted in text classification tasks and web-scale indexing. This is because the less frequently occurring words typically offer more information about the concept being conveyed, and they allow different sentences to be more easily distinguished because there is a lower probability that these words will co-occur in different sentences or documents. The researchers note that subsampling of these frequently occurring words enables anywhere from a 2x to 10x improvement in speed for the training of the model. Additionally, “the effective window size grows, including context-words which are both content-full and linearly far away from the focus word, thus making the similarities more topical.” [6] In other words, removing the frequently occurring words not only eliminated uninformative words; it also allows for the skipgram to effectively extend past the original window size to include more words in the “neighborhood” of the word.

In attempting to explain the mathematics behind negative sampling in word2vec, researchers [6] show the following formulation for the objective of the skip gram model achieved by negative sampling:

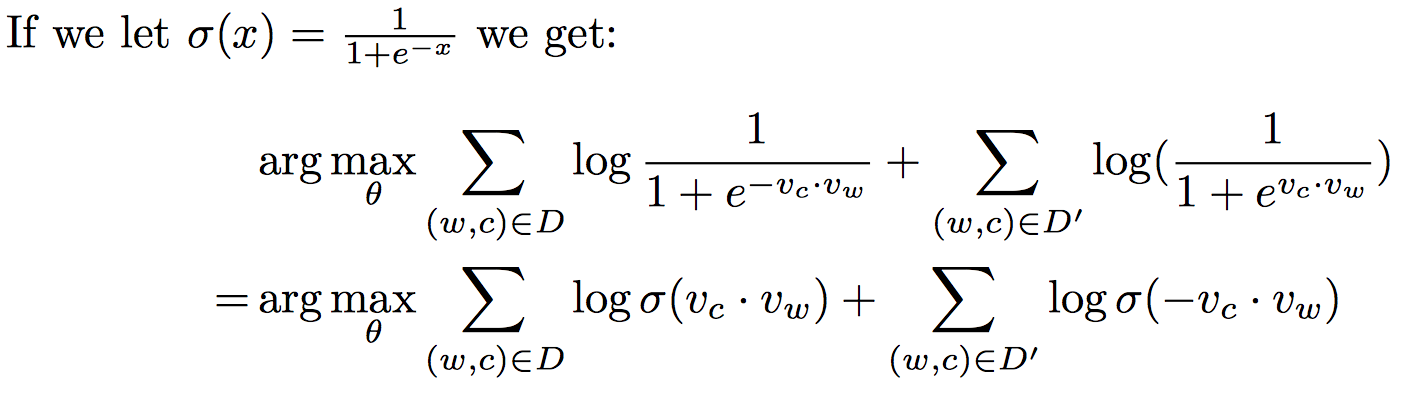
,

where w are the words, c are the contexts, D or D=1 is the training data, D’ or D=0 is the randomly generated data, and theta (Θ) is the set of tuning parameters.

The researchers note that not all vectors can have the same value. To address this constraint, the researchers generate a set D’ of random (word, context) pairs that do not occur in the data. This is where the “negative” in “negative sampling” comes from, where the negation implies that the random pairs do not occur in the training data.

The basic intuition here is that we want to maximize the product of two probabilities. The first is the probability that the observations came from the real training data set, given the context and the word and theta, which is the set of parameters. Here, we assume the words and context to come from the training data. The second element is the probability that the observations did not come from the training data (i.e., they came from the randomly generated pairs), given the context, the word, theta, and assuming that the word and context came from the randomly generated dataset D’. There are two approaches that can be taken with this task; either we remove the word from its context and try to predict the word that belongs given the context (skip gram), or we try to generate the correct context given the word itself (continuous bag-of-words, cBoW) using gradient descent with randomly initialized vectors.

The researchers in [6] propose a manipulation of the equations given in the Mikolov paper [2] by transforming the probability products into summations of the natural log of the sigmoid function, where the exponent of the sigmoid function is the dot product of the context and word vectors:

,

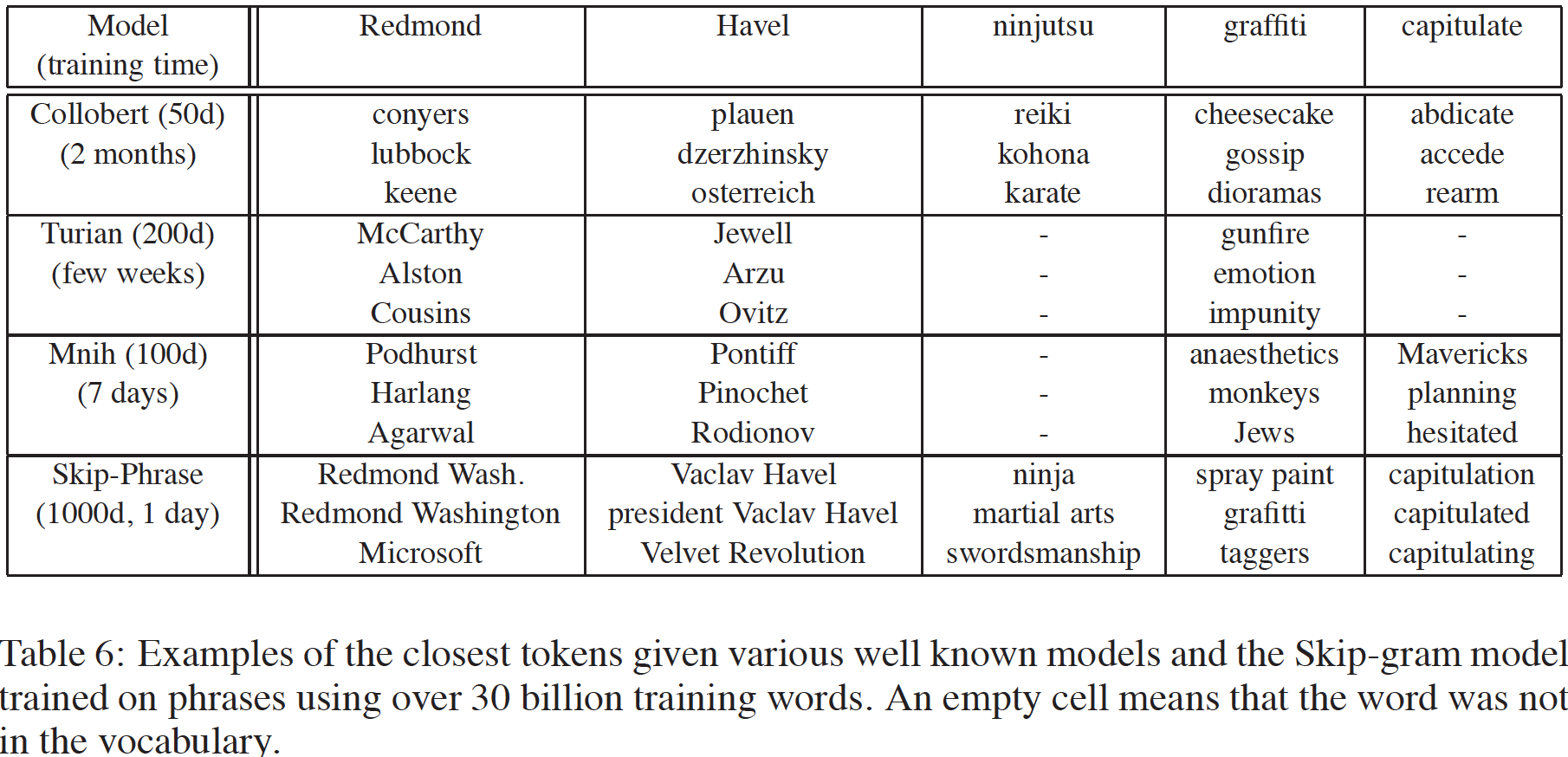
where D is the set of all word and context pairs extracted from the text corpus, w is the word, c is the context, and vc and vw are the vector representations for contexts and words, respectively.

It is implicitly assumed here that the word and context vectors each come from different vocabularies, because “words hardly appear in contexts of themselves.” This assumption is important to make because certain dot product calculations should have a low value (e.g., p(cat|cat) should be low), and this would not be possible because v\*v cannot be almost zero unless we are describing two *distinct* vectors.

Word2vec is made even more efficient by the use of Noise Contrastive Estimation (NCE, or negative sampling), which uses logistic regression to separate the signal from the noise. This is performed in place of softmax or hierarchical softmax (applied to neural network language models by Yoshuo Bengio of the University of Montreal) to train the skipgram model on the data. The reason that the authors of word2vec were able to achieve a drastic decrease in training time by the use of NCE is that “…the Skip-Gram model is only concerned with learning high-quality vector representations, and as such the authors were free to simplify NCE as long as the vector representations retain their quality.” [9] This simplification of the noise-contrastive elimination is described by David Meyer in [9] thusly: “NCE connects the problem of PDE (probability density estimation) to supervised learning, in particular to logistic regression, and provides a hint as to how the proposed estimator works: By discriminating, or comparing, between data and noise, NCE can learn properties of the data in the form of a statistical model. That is, the key idea behind noise contrastive estimation is learning by comparison.”

The efficient implementation of word2vec is very important, because the training required to achieve the best accuracy critically depends on very large volumes of data. The researchers at Google note that the model improved in accuracy from 66% to 72% when the data was comprised of 33 billion words compared to 6 billion words.

Table #1: Training time, dimensionality, and sample generated output of leading models.



The above table shows that word2vec significantly cuts down on training time compared to other leading models. It is also capable of handling high dimensionality (1000 dimensions). Most importantly, the generated embeddings are much more reasonable compared to those produced by the other leading models. For example, for the word “Redmond”, word2vec was the only model to associate this with Microsoft, which is headquartered in Redmond, Washington. It also successfully associates the state with Redmond, which is important when dealing with proper nouns pertaining to physical places. This kind of relationship mapping country to capital or city to state can be viewed in figure #1.

Recent follow-up research on this topic has helped to elucidate why word2vec has achieved state-of-the-art performance and offered insights into how other methods can perform better. Yoav and Goldberg [10] argue for using a “multiplicative combination” instead of an additive one:

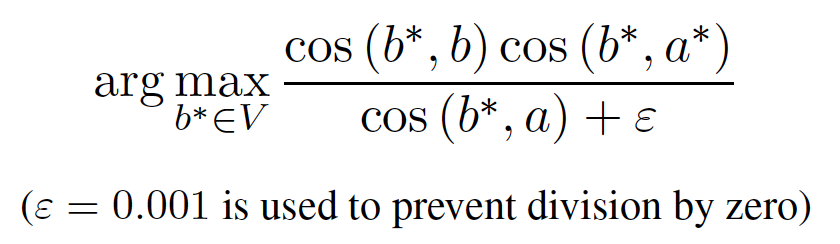
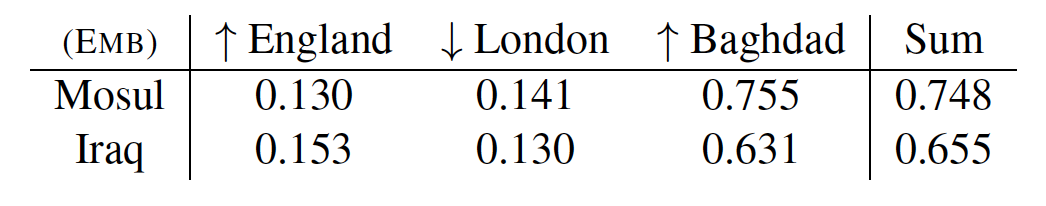


Figure #2: Multiplicative combination

, where the similarity attempts to maximize the product of the cosine similarities of word a and word b, normalized and “smoothed” by epsilon = 0.001 to avoid the problem of division-by-zero introduced by the multiplication approach. This multiplicative combination resulted in improved results over the additive approach in the Mikolov et. al word2vec model. The reason this improved performance is that it prevents any one aspect of the similarities from dominating others. For example, comparing countries and their capitals to derive the relationship between the two, the “city” similarity is much larger than the country similarity, so analogies of this form will generate embeddings of other large cities rather than focusing on the relationship between the country and its capital specifically. See table #2, below, for an example of this imbalance.

Table #2: Embedding similarities across contextual aspects



Although not covered comprehensively in this paper, the GloVe (Global Vector) model for word embeddings proposed by Pennington et. al (2014) attempts to capture the strengths of the two major approaches to word embeddings. The first approach, which uses matrix factorization techniques on word co-occurrence matrices (such as Latent Semantic Analysis), makes use of the global corpus statistics, but it does not scale well and is slower to train. The second approach, which uses neural network language models such as word2vec, does not make use of the global statistics but does scale much better. The “global” in the name of the GloVe model refers to the authors’ attempt to make an efficient model that also incorporates the global statistics on the corpus. Results demonstrate that this model outperforms word2vec and also performs better than the multiplicative combination that Goldberg and Levy advocate for. In GloVe, cosine similarity outperformed Goldeberg and Levy’s 3COSMUL. The main intuition is that the semantic relationships are encoded in the *ratio* of the probabilities in the word co-occurrence matrices, rather than the probabilities themselves. The ratios allow for the efficient cancellation of noise in the data. Because the log of a ratio of probabilities is a difference of log probabilities, this allows the model to be represented as a weighted least-squares regression.

# Sentiment classification

According to the Association for Computational Linguistics, sentiment classification is “a special task of text classification whose objective is to classify a text according to the sentiment polarities of the opinions it contains.” Sentiment classification is a well-researched topic with strong interest in academia as well as industry for its commercial applications. Currently, there is a wealth of research into sentiment polarity classification, but there is much work to be done in the area of emotional classification, which uses multiple classes. Polarity classification’s binary approach uses two classes: positive and negative, whereas emotional sentiment classification can use as many classes as there are human emotions, and the implementation of this task can be further complicated by fuzzy set membership, where there might be emotional overlap for a given instance (i.e., an instance can have multiple class membership).

Although the more fine-grained emotional classification task is crucial in further developing the utility of sentiment classification models, the basic polarity classification step is crucial because the system will not produce the correct emotion (or even close to it) if the polarity is not correct. Therefore, the reduction of errors at this step in the training of the model is of paramount importance. Errors in the more fine-grained emotional classification task are not as severe. According to [3] “misclassifying a case of happiness as, for example, pleasantness may well be tolerable.” A two-step approach can be taken, where sentences are first classified into their respective polarities (positive, neutral, negative), and then further classified into fine-grained emotions.

In practice, it is difficult to obtain neutral training instances for emotional classification because most examples on the web are not emotionally neutral. Researchers in [3] simply perform a coarse sentiment polarity classification and then classify as neutral those instances that lie near the decision boundary. This is an important step in the overall sentiment classification step because it is undesirable to have neutral sentences generating an emotional response from the system. Presumably, the users of such systems would find it to be inappropriate and unexpected. To train the polarity classifier, support vector machines are used with unigram, bigram, and trigram features of the data. Research has demonstrated that highly accurate classifiers can be trained to successfully distinguish between positive, negative, and even neutral examples on n-gram word features alone without polarity features. This is important because even the available polarity corpus is only 57% “exactly” correct. [3] Even though these systems are capable of training successful polarity classifiers, future work will need to be done to improve the error rate on the neutral instances. Some work has been done in [7] to improve upon the SentiWordNet lexical resource to improve the performance of polarity classifiers. The accuracy of the newest version, SentiWordNet 3.0, has seen “accuracy improvements of about 20% with respect to SENTIWORDNET 1.0.” The newest version uses ensemble classification methods such as Rocchio’s algorithm and Support Vector Machines, and the gloss similarity of synsets is determined with Dice’s coefficient. Rankings are compared using p-normalized Kendall distance. These steps that have been taken to improve this valuable resource will likely lead to improved sentiment classifiers.

In contrast to traditional sentiment classification tasks, more recent approaches use *multiple* classes for different emotions, in addition to the first step of polarity classification. For example, researchers in [3] “…used ten emotions: happiness, pleasantness, relief, fear, sadness, disappointment, unpleasantness, loneliness, anxiety, (and) anger.” This classification task can be performed using a k-nearest neighbor approach with cosine similarity used as the similarity measure between bag-of-words vectors. These researchers used three values of k (1,3, and 10) and compared performance to a baseline using three corpora.

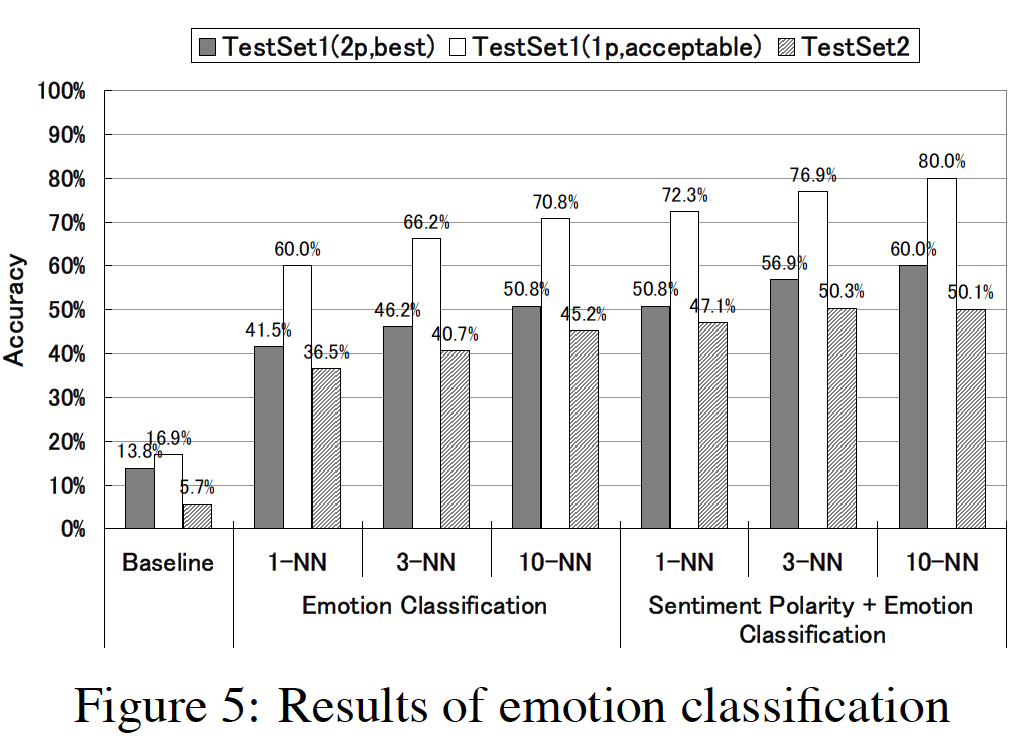


Figure #2 Performance with respect to accuracy for various models of sentiment classification.

An important distinction about the model proposed in [3] is that it requires no supervised training instances. All instances can be obtained without supervision from the web, meaning that the superior results over previous work can be obtained without the expense of having to manually classify training instances. Particularly for models that require a great deal of training data, this is an enormous advantage and allows this kind of model to be scaled up for future improvement. Another unsupervised technique that applies here is clustering of Wikipedia documents to disambiguate sense embeddings in sentiment classification.

Future research is needed to examine other machine learning approaches that may be able to improve the model. K-nearest neighbors is an expensive model to use at query time because it does not store the structure of the model in training, so having a vast vocabulary of emotions to draw from after the polarity classification task might be time-expensive. The latency of dialog systems can be frustrating to users, and this issue will need to be addressed. When humans interact with any kind of artificial intelligence that attempts to respond emotionally to the content of their text or speech, it is important to respond quickly because there is no such delay when humans interact with other humans due to the way we process emotions and speech instantaneously.

# Non-textual sources of natural language

In an attempt to extract more emotional meaning and semantics from text-based digital communications, any natural language understanding system will reach a limit to the accuracy of the interpretation or prediction due to the importance of non-textual meaning cues in human language. Humans have evolved very powerful and complex neural architectures to process speech and writing, and much of the information used to “decode” communications from other humans comes from facial expressions, prosodic cues, eye contact, and body language. As these sources cannot be imbedded into text communication, it can be difficult even for humans to deconstruct the meaning of a sentence written in a tweet, email, SMS text message, or other digital communication short of video. As such, natural language understanding systems must incorporate more than just textual information. An interesting source of such information lies in emoticons, or “emojis.”

Research in [11] shows that machine learning approaches that can efficiently handle multiclass sentiment classification tasks need to be further researched, but significant improvements are demonstrated in their results by taking advantage of the presence of emoticons and negation identifiers and by performing Chi-square feature selection. The researchers here used a set of emoticons that are “unambiguous and widely used.” Using multi-class linear support vector machines, naïve Bayes, and maximum entropy, they were able to improve classification accuracy from around 70% to as high as 84% using a combination of the new approaches. These researchers “…are also working on lexicon-based sentiment-classification methods, hybrid methods (combining lexicon and machine learning), ruled-based methods, and emotion-pattern methods to develop more powerful tools for sentiment classification. Our experience shows that the crucial problem is the accuracy of the classifiers.”

Previously, it was mentioned that the polarity of a sentence is very important to establish before more fine-grained emotional labels can be attached. It may be the case that emojis can offer information that NLU systems can incorporate when attempting to disambiguate meaning or word sense during polarity classification. For example, “emoticons can also change the direction of a message to the opposite direction, which is the case of sarcastic utterances…emoticons are more expressive than punctuation marks in the disambiguation of sarcastic utterances.” [5]

Research [5] has shown that “the tweets with emojis are significantly more positive (mean = +0.365) than the tweets without emojis (mean = +0.106)” (see figure, below). However, the recent rise in popularity of hyperbolic speech and sarcasm in the presence of plaintive speech that is prevalent in social media networks can make it difficult to discern exactly how positive a given body of text is. Efforts to better understand exactly how emojis are used can allow researchers and practitioners to more accurately assess how positive a message is. Pre-labeled datasets of ratings such as those found on Amazon.com, Yelp, etc. offer the ability for researchers to examine how frequently a “positive” word occurs in the higher ranges of the scale. For instance, a word that occurs in 10-star reviews (on a scale out of 10 stars) more frequently than it does in 9- and 8-star reviews, etc. can help researchers identify the strength or intensity of such a word beyond a simple positive/negative classification.

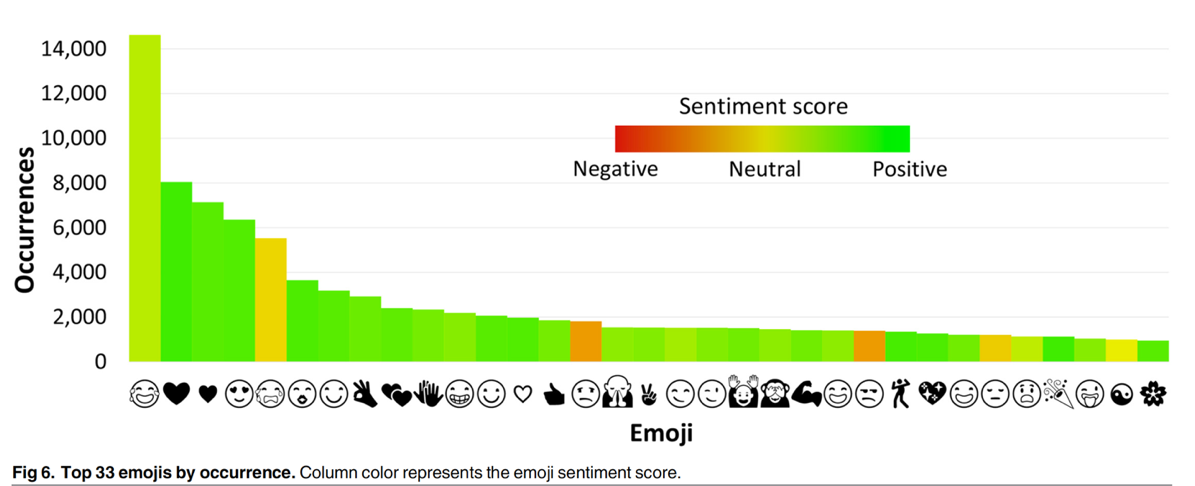


Figure #3: The top 33 emojis by occurrence

**3 Conclusion**

In this paper, several areas of research within natural language understanding have been surveyed with the intent of informing the reader of current research in this exciting field. Additionally, future avenues for research have been presented.

It has been shown that distributed vector space models for word embedding generation such as word2vec can be highly scalable and capture more of the true “sense” of words and phrases than traditional statistical techniques in natural language understanding. Additionally, word2vec has been shown to be more efficient compared to previous models due to the use of negative subsampling and approximations of the softmax function. Improvements to word2vec using multiplicative approaches can help us increase its performance.

Approaches such as GloVe have attempted to capture the best aspects of both the more traditional embeddings approaches as well as the neural network approaches like word2vec. Results are promising, suggesting that future work in this area can attempt to make more robust models that have strengths in multiple areas. Although word2vec and GloVe exhibit state-of-the-art performance on benchmark analogy tasks, there is a significant difference between the performance on semantics tasks compared to syntactical tasks. This suggests that future research needs to be done to improve the syntactical tasks to increase the model’s utility across a variety of tasks and domains.

Extensions to traditional sentiment classification show promise for improved performance in customer dialog systems, reputation monitoring, and active listening. Traditional approaches focus on binary classification, but the most recent research has focused on multiclass and multilabel classification problems. Although they can present more difficulty and complexity, these approaches can offer much more information to users of these systems.

The use of non-textual information in natural language data sources could potentially increase performance of various systems and allow for better disambiguation techniques. It has been shown in this paper that the successful polarity classification is of even higher importance than emotional or “affective” classification, and the use of emojis and other non-textual cues can help address the difficulty of this task. Specifically, the use of emojis can be even more powerful than the traditional approach of using punctuation to identify negation and handle ambiguity.

Some of the applications that are likely to benefit from current and future research in the areas covered in this paper are customer service, reputation monitoring, ad placement, market intelligence/research, regulatory compliance, dialog interfaces and digital assistants (siri, alexa, mobile search, etc.), business intelligence (from SQL queries to natural-language voice or text queries), text summarization, real-time monitoring of social media networks for news outlets, e-commerce, autism and elderly therapeutic speech system interventions, recommender systems (music playlists, social networks, etc.) and e-learning, among others.

**4 References**

[1] <http://www.nytimes.com/2008/09/14/weekinreview/14arango.html?src=tp>

[2]<http://web2.cs.columbia.edu/~blei/seminar/2016_discrete_data/readings/MikolovSutskeverChenCorradoDean2013.pdf>

[3] https://www.aclweb.org/anthology/C/C08/C08-1111.pdf

[4] http://www.nytimes.com/2008/09/14/weekinreview/14arango.html?src=tp

[5]https://www.researchgate.net/publication/312102025\_An\_Integrated\_Review\_of\_Emoticons in\_Computer-Mediated\_Communication

[6] https://arxiv.org/pdf/1402.3722.pdf

[7] http://nmis.isti.cnr.it/sebastiani/Publications/LREC10.pdf

[8]https://www.researchgate.net/profile/Scott\_Deerwester/publication/2462489\_Using\_Latent\_Semantic\_Analysis\_To\_Improve\_Access\_To\_Textual\_Information/links/543c33b80cf2c432f741775f.pdf

[9] http://www.1-4-5.net/~dmm/ml/how\_does\_word2vec\_work.pdf

[10] https://levyomer.files.wordpress.com/2014/04/linguistic-regularities-in-sparse-and-explicit-word-representations-conll-2014.pdf

[11] https://www.researchgate.net/publication/275219783\_Enhancing\_Machine-Learning\_Methods\_for\_Sentiment\_Classification\_of\_Web\_Data