# Sentiment Analysis with Word Embedding

Oscar B. Deho
*Computer Science and Engineering Department*
*University of Mines and Technology*
Tarkwa, Ghana
iam.oscardeho@gmail.com

William A. Agangiba
*Computer Science and Engineering Department*
*University of Mines and Technology*
Tarkwa, Ghana
waakotam@umat.edu.gh

Felix L. Aryeh
*Computer Science and Engineering Department*
*University of Mines and Technology*
Tarkwa, Ghana
flaryeh@umat.edu.gh

Jeffery A. Ansah
*School of Information Technology and Mathematical Sciences*
*University of South Australia, UniSA*
Adelaide, Australia
jeffery.ansah@mymail.unisa.edu.au

***Abstract*—The basic task of sentiment analysis is to determine the sentiment polarity (positivity, neutrality or negativity) of a piece text. The traditional bag-of-words models deficiencies affect the accuracy of sentiment classifications. The purpose of this study is to improve the accuracy of the sentiment classification by employing the concept of word embedding. This study uses Word2Vec to produce high-dimensional word vectors that learn contextual information of words. The resulting word vectors are used to train machine learning algorithms in the form of classifiers for sentiment classification. Our experiments on real world datasets shows that the use of word embedding improves the accuracy of sentiments classification.**

***Index Terms*—Word embedding, Word2Vec, Machine Learning, Bag-of-words.**

## I. Introduction

The advent of the World Wide Web and the Internet makes available a high volume of accessible information with many freely available platforms for opinion sharing. Online Forums, blogs, social media platforms and other content-sharing services help people to share insightful information [24]. Social events, product reviews, political issues among other things, are the central idea of the information that is shared on these media platforms [24]. It is estimated that the total number of social media users would be at least 3.2 billion in 2018 [3]. Facebook, YouTube, Twitter, Reddit, Instagram and Pinterest are leading the social media global market race [7].

A variety of sentiments are borne in posts on these social media platforms. The posts therefore can be an invaluable source of information which can be leveraged to gain insights and make decision on issues and policies [12]. Analyzing a piece of text to determine whether it conveys a favorable or unfavorable emotion or opinion about an entity leads to a concept known as sentiment analysis. Sentiment analysis, (or opinion mining) is a natural language processing task which aims at determining whether a piece of text carries positive, neutral or negative sentiment. Sentiment analysis can be done at the document-level [20], sentence level [6], or aspect level (sentiment about specific aspects of an entity) [23]. Corporate bodies use sentiment analysis as a tool to gain insights on market trends and consumer expectations, to gain competitive edge advantage [12]. Political agencies also use sentiment analysis to predict election outcomes, plan campaign messages and policies. The authors of [22] used sentiment analysis to gain insight into the 2012 presidential election of the United States of America.

The main approaches for sentiment analysis are the lexicon-based approach and the machine learning approach [21]. The lexicon-based approach measures the polarity and subjectivity of a textual data against a database (lexicon) of emotional values of words [21]. Different approaches to creating dictionaries have been proposed; viz. manual and automatic techniques [19]. In lexicon-based approach such as [18], a piece of text is represented generally as a bag-of-words and a combining function such as average or sum is applied to determine the overall sentiment embedded in the text. Lexicon-based approaches are easy to implement, but has a downside of disrupting word order and discards semantic information [23]. With the machine learning methods, sentiments are classified by applying a machine learning algorithm in the form of classifiers to a piece of text [11]. There have been lots of researches done in the area of sentiment analysis with an extensive survey presented in [21].

The bag-of-words (BOW) model is a common method used for text representation. According to [15], the BOW is at best, good for topic-based text classification and not sentiment analysis. The BOW loses contextual information (a key requirement in accurate sentiment classification) by disrupting word order and discards contextual information [15]. The authors of [15], coupled the BOW model with an ensemble method made up of 5 classifiers namely; Naive Bayes, MultinomialNB, Bernoulli, Stochastic Gradient Descent (SGD) and Support Vector Classifier (SVC) which resulted in an average accuracy of 72%. The result in Gargs work stems largely from the deficiencies of BOW model. Polarity shifts such as negation are handled poorly by the BOW model according to [15]. The introduction of a negation word like *"dont"* to the text *"I like this phone"* reverses the polarity from positive to negative. The

deficiencies of the bag-of-words model affect the accuracy of sentiment classification. It is therefore necessary to identify more effective methods for performing sentiment analysis in order to improve accuracy and to preserve context.

In this paper, we demonstrate the application of word embedding; a context preserving and high accuracy technique for sentiment analysis.

## II. Related Work

In this section we discuss closely related works and some key concepts crucial to implementation of our methodology.

### A. Word Embedding

Word Embedding emerged from the field of Natural Language Processing (NLP) which is an intersection of Computer Science, Artificial Intelligence, Machine Learning and computational linguistics [4]. Word embedding is a text mining technique of establishing relationship between words in textual data (Corpus). The syntactic and semantic meanings of words are realized from the context in which they are used. The concept of distributional hypothesis suggests that words occurring in similar context are semantically similar [16]. Count based embeddings and prediction based embeddings are the two broad approaches to word embedding [17]. Just like the traditional bag-of-words model, the count based embeddings performs poorly at preserving contextual information in textual data according to [2]. The prediction based embeddings try to predict a target word given a context word [2]. Global Vectors for Words Representation (GloVe), an unsupervised learning algorithm for obtaining vector representation of words developed by researchers at Stanford [13] does very well at context preservation.

### B. Word2Vec (Skip-Gram Model)

The Word2Vec model is a prediction-based algorithm developed by Tomas Mikolov and his colleagues at Google in 2013. The Word2Vec model uses a shallow neural network to embed high quality word vectors [9]. The underlying principle of the word2vec model is that words occurring in similar context are related [14]. The Word2Vec algorithm comes in two flavors: the continuous bag-of-words (CBOW) model and skip-gram (SG) model. The continuous bag-of-words model predicts a target word, given a context of words while the skip-gram model flips the CBOW architecture around by predicting the context of a given word. The skip-gram model coupled with negative sampling outperforms the CBOW making it the preferred choice for this paper.

## III. Methodology

In this section, we discuss our approach of leveraging word embedding for sentiment classification. To be specific we adopt the Skip-Gram Model due to effectiveness in preserving context without trading-off accuracy.

Recall, that the underlying principle of the word2vec model is that words occurring in similar context are related [14]. These input words could be keywords, phrases or terms extracted from documents, new articles etc. In our context, we use words from tweets. Given an input word, the neural network of the skip-gram model is able to pick a random word from the context window as the target word. Word2Vec is a shallow neural network with an input layer, a single hidden layer and output layer [9]. The architecture of word2vec is such that there is a weight matrix between the input layer and the hidden layer and another one between the hidden layer and the output layer. The weight matrix between the input and hidden layers is actually the word vectors we are after.

The weight matrix has an m*n dimension where m is the size of the dictionary (a list of all the unique words (tokens) in the corpus) and n is the size of the hidden layer. The size of the hidden layer is a hyper-parameter hence can be tuned as deemed fit considering the corpus being used for training according to [8]. One-hot encodings of word pairs from the dictionary are fed into the neural network during training. We adopt the three adjustments made by the authors of Word2Vec [10] in their second paper to ensure fast computational speed as well as quality word vectors [10]. These adjustments can be summarised as:

- Common word pairs or phrases were treated as single words
- Frequent words were subsampled to decrease number of training examples
- A technique called Negative Sampling was introduced which causes the training sample to update only a small percentage of the models weights.

## IV. IMPLEMENTATION

In this paper, we test our methods by performing for sentiment analysis on tweets relating to the establishment of U.S. Military base in Ghana. The steps that were taken in the sentiment analysis process are shown in the Fig. 1

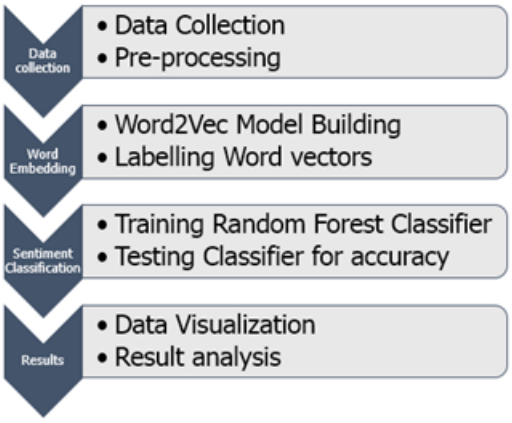


Fig. 1. Steps in Sentiment Analysis.

### A. Data Collection and Preprocessing

The data for this project were tweets relating to the establishment of U.S Military base in Ghana. A twitter account was created, and the login credentials of the twitter account

were used to sign in to apps.twitter.com in order to create a mini twitter application. The mini twitter application comes with consumer keys and secret, access tokens and access token secret. The access and consumer keys and secrets are passed to the authentication handler which allows access to twitter data. Tweepy, a python twitter data streaming library was used to stream the tweets. Tweets were collected for 6 consecutive days when the topic U.S. Military Base in Ghana and hashtags like #PutGhanaFirst, #StopUSMilitaryBaseInGhana were trending. The tweets were saved in JSON (JavaScript Object Notation) file format. The saved tweets were then preprocessed so that the data can be fit for feature extraction. The final dataset1 was then derived after stop words were removed and tweets tokenized and stemmed to remove inflectional forms of words to their stem word.

### *B. Word2Vec Model Building*

The skip-gram model of the word2vec algorithm is used for this project. The word2vec implementation of the Gensim python library was trained on the processed data. For better word embeddings, certain hyper-parameters were given utmost consideration. These parameters are: training algorithm, dimensionality, context window and sub-sampling. The training algorithm used for this project was negative sampling as it proved to be computationally efficient compared to hierarchical softmax. A dimension of 300 was assigned to the hidden layer of the neural network as it resulted in better word embeddings. A context window of 10 was used as it was the prescribed context window for skip-gram models [10]. The sub-sampling rate of 1e-3 was used to counter the imbalance between rare and frequent words in the dataset.

Due to the size of the data used, the minimum count was set to 1 so that every word in the corpus was considered during training. The word2vec model was trained on the processed data with the aforementioned hyper-parameter settings and saved in a file data format. The word vectors produced by the word embedding model have m*n dimension where m is the size of the dictionary and n is the size of the hidden layer.

### *C. Labelling Word Vectors*

The word vectors were split into training and testing set. The train-test-split function of sci-kit learn was used to split the dataset where 70 % of the word vectors were used as training sample and the rest for testing. VADER (Valence Aware Dictionary for sEntiment Reasoning), a sentiment analysis engine was used to determine the polarities of the various tweets. VADER was used to determine the polarities of the raw tweets after which these polarities were assigned to 70 % of the respective word vectors as the training data set. The labelled word vectors were used to train a random forest classifier fitted with 100 decision trees.

### *D. D. Sentiment Classification*

The random forest classifier used for this project was fitted with 100 decision trees and trained on the labelled word vectors. Random forest classifier was used for this project as it handles overfitting by randomly selecting subsets of the training set and creating a set of decision trees from those subsets. After training of the classifier, it was used to predict the sentiment polarities of the test dataset. Scikit learns metrics module was used to measure the accuracy, precision, recall and F1-score of the classifier as shown in Fig. 2.

```
Accuracy:  0.81
================== Results ==================
         Negative     Neutral     Positive
F1       [0.87747036 0.64788732 0.73684211]
Precision[0.83458647 0.88461538 0.68292683]
Recall   [0.925      0.51111111 0.8       ]
Accuracy 0.81
=============================================
```

Fig. 2. Accuracy and Validation Metrics Score of the Random Forest Classifier.

## V. Results and Discussion

### *A. Results*

The Fig. 2 shows the classification report of the random forest classifier for each sentiment class. The precision rate, F1-score, recall and overall accuracy of the classifier for the various sentiment classes is shown in the Fig. 2. An overall accuracy of 81% shows that the classifier did well in predicting the sentiment polarities which is due to the quality of word vectors that were produce by the skip-gram model. The confusion matrix shown in Fig. 3 gives a detailed view of how the classifier did the classification with respect to the true positives, false positives, true negatives and false negatives. The Fig. 4 is a pie chart that was used to substantiate the various percentages of tweets that bore positive, neutral or negative sentiments. From the pie chart in Fig. 4, it can be inferred that more than half of the tweets were negative which denotes disapproval of citizens to allow the establishment of a military base in Ghana. The neutral sector denotes those tweets which either were not so clear with their stance on the issue or probably were simply indifferent about the decision to allow the establishment of the U.S. Military Base in Ghana. The positive sector denotes those tweets that were of the view that the establishment of the U.S. Military Base in Ghana would not cause any harm but rather comes with some benefits Ghana stands to enjoys should the military base be established.

Comparing the results from this study with those of [15] and [1] it can be inferred that similar ranges of accuracy were obtained which goes to prove that word embedding helps improving classification accuracy. Reference [15] in their work combined both word2vec and the BOW model for text representation and Logistic Regression Classifier as the classification algorithm to perform sentiment analysis on 50,000 movie reviews. For the positive class, they had 77.9% precision, 77% F-score and 76.7% recall. For the negative class they had 76% precision, 77.4% F-score and 78.8%

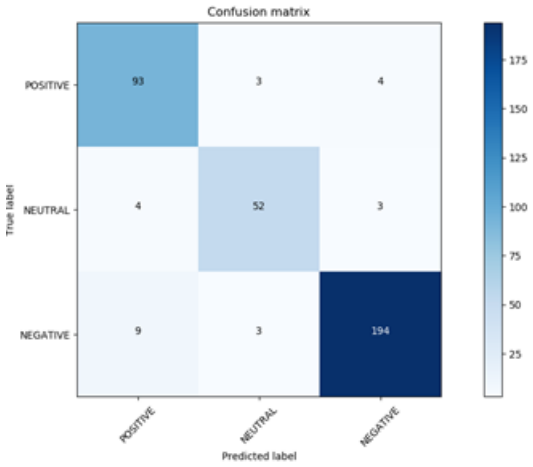


Fig. 3. Confusion Matrix of Random Forest Classifier.

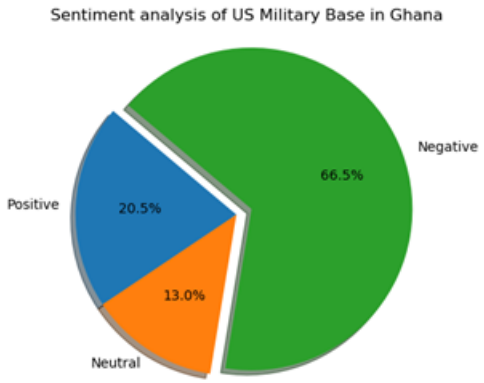


Fig. 4. Pie Chart showing Sentiment Polarity Distribution.

recall. Reference [1] also used word2vec in their work to perform sentiment analysis on 14,640 tweets on U.S airlines and obtained similar range of values. For the negative class, the classifier accurately predicted 75% of 2750 negative test instances correctly with a precision rate of 87% and an F1-score of 81%. For the neutral class, the classifier accurately predicted 62% of 936 neutral test instances correctly with a precision rate of 51% and an F1-score of 56%. For the positive class, the classifier accurately predicted 70% of 706 positive test instances correctly with a precision rate of 57% and an F1 -score of 63%.

## VI. CONCLUSION AND RECOMMENDATIONS

In this work, we have presented our approach of using Word Embedding to determine the sentiments polarity (positivity, neutrality, or negativity) of a given text. We adopted a Skip-gram variant of the Word2Vec implementation and tested our results on real world datasets obtained from Twitter. The results from our experiments shows that word embedding greatly improves the accuracy of sentiment classification. Further interpretable insights reflecting peoples emotions obtained from this work is useful to political agencies and other bodies in understanding the concerns of citizens for decision making. We plan to incorporate Sarcasm detection and handling in future work for sentiment evaluation.

## REFERENCES


[1] J. Acosta, N. Lamaute, M. Luo, E. Finklestein and A. Cotoranu, Sentiment Analysis of Twitter Messages Using Word2Vec, Proceedings of Student-Faculty Research Day, CSIS, Pace University, Pleasantville, New York, 2017, 7pp.

[2] M. Baroni, G. Dinu and G. Kruszewski, Dont count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors, ACL, 2014, pp. 238-247.

[3] D. Chaffey, Global social media research summary 2018, 2018. [Online]. Available: https://www.smartinsights.com/social-media-marketing/social-media-strategy/new-global-social-media-research/. [Accessed: May 15, 2018].

[4] A. Chopra, A. Prasha and C. Sain, Natural Language Processing, International Journal of Technological Enhancement and Emerging Engineering & Research, Vol.1, No.4, 2013, pp.131-134.

[5] P. Garg, Sentiment Analysis of Twitter Data using NLTK in Python, Thapar University, Patiala, 2016, 50pp.

[6] M. Hu and B. Lui, Mining and summarizing customer reviews, 2004, KKD.

[7] P. Kallas, Top 10 Social Networking Sites by Market Share Statistics, 2017, [Online]. Available: https://www.dreamgrow.com/top-10-social-networking-sites-market-share-of-visits/. [Accessed: May 15, 2018].

[8] C. McCormick, Word2Vec Tutorial -The Skip-Gram Model, 2017, [Online]. Available: http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/. [Accessed: May 15, 2018].

[9] T. Mikolov, K. Chen, G. Corrado and J. Dean, Efficient Estimation of Word Representation in Vector Space, Google Inc, 2013, 12pp.

[10] T. Mikolov, I. Sutskever, K. Chen, G. Corrado and J. Dean, Distributed Representation of Words and Phrases and their Compositionality, Google Inc, 2013, 9pp.

[11] B. Pang and E. Lee, Thumbs up? Sentiment Classification using Machine Learning Techniques, Proc.Conf. Empirical Methods Natural Language, 2002, pp. 79-86.

[12] B. Pang and E. Lee, Opinion mining and sentiment analysis, 2008, [Online]. Available: http://www.cs.cornell.edu/home/llee/omsa/omsa.pdf. [Accessed: May 15, 2018].

[13] R. Pennington, R. Socher, and D. C. Manning, GloVe: Global Vectors for Word Representation, Stanford University, Stanford, 2013, 12pp.

[14] X. Rong, Word2Vec Parameter Learning Explained, arXiv:1411.273v4, 2016.

[15] S. P. Shamseera and E. S. Sreekanth, Word Vectors in Sentiment Analysis, International Journal of Current Trends in Engineering & Research (IJCTER), Vol.2, No.5, 2016, pp.594-598.

[16] M. Sahlgren, The Distributional Hypothesis, Italian Journal of Linguistics, 2006, 18pp.

[17] R. Sunil, An Intuitive Understanding of Word Embeddings: From Count Vectors to Word2Vec, 2017, [Online]. Available: https://www.analyticsvidhya.com/blog/2017/06/word-embeddings-count-word2veec/. [Accessed: January 10, 2018].

[18] M. Taboada, J. Brooke, M. Tofiloski, K. Voll and M. Stede, Lexicon based methods for sentiment analysis, Computational linguistics Vol.37, No.2, 2011, pp 267-307.

[19] P. Turney and P. Pantel, From Frequency to Meaning: Vector Space Models of Semantics, International Journal of Artificial Intelligence Research, Vol.37, No.1, 2010, 145pp.

[20] P. Turney, Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews, 2002, ACL.

[21] A. K. Vishal and S.S. Sonawane, Sentiment Analysis of Twitter Data: A Survey of Techniques, International Journal of Computer Applications (0975-8887), Vol.139, No.11. 2016.

[22] H. Wang, D. Can, F. Bar and S. Narayana, A system for real-time Twitter Sentiment Analysis of 2012 U.S. presidential election cycle, Proc. ACL 2012 System Demonstration, 2012, pp. 115-120.

[23] T. Wilson, J. Wiebe and P. Hoffman, Recognizing contextual polarity in phrase level sentiment analysis, 2005, ACL.

[24] Ansah J, Kang W, Liu L, Liu J, Li J. Information Propagation Trees for Protest Event Prediction. InPacific-Asia Conference on Knowledge Discovery and Data Mining 2018 Jun 3 (pp. 777-789). Springer, Cham.