A Context Based Algorithm for Sentiment Analysis

Article *in* International Journal of Computational Vision and Robotics · January 2018

DOI: 10.1504/IJCVR.2017.10005890

4 authors, including:

Srishti Sharma
Netaji Subhas Institute of Technology

Akhil Sharma Sharma
Kumaun University

A Context Based Algorithm for Sentiment Analysis

Srishti Sharma*, **Shampa Chakraverty, Akhil Sharma**
Department of Computer Engineering,
Netaji Subhas Institute of Technology,
Dwarka, New Delhi- 110 078, India
E-mail: srish.060788@gmail.com
E-mail: apmahs.nsit@gmail.com
E-mail: akhil.sharma92@yahoo.com
*Corresponding author

Jasleen Kaur
Department of Electronics and Communication Engineering,
Netaji Subhas Institute of Technology,
Dwarka, New Delhi- 110 078, India
E-mail: jasleenkaurbatra@yahoo.co.in

***Abstract*— With netizens continuing to express a range of opinions and making assessments online, it has become a challenge to mine sentiments accurately from the ever-multiplying Big Data. We present a context-driven Sentiment Analysis scheme with the objective of refining the degree of subjectivity during Sentiment Analysis. The essence of our scheme is to capture in a stable manner, the mutual influence of the sentiments of neighboring words on the sentiment of each word in a document. A parametric Influence Function combines the native sentiment score of each word with the context-derived sentiment score obtained from surrounding words. We apply a Genetic Algorithm to fine tune the parameters of the Influence Function so as to obtain the best possible accuracy for a given corpus. Experimental results on hotel reviews extracted from Tripadvisor.com show an average accuracy of 73.2% which is 3.6% more than the results obtained from the Baseline Sentiment Analysis approach using native scores obtained from SentiWordNet. Though the improvement is small, it re-affirms our belief that**

contextual information provides valuable reinforcement of sentiment scores especially with regard to the borderline cases where words show near neutral sentiments. We also present a comparison with alternative Sentiment Analysis approaches that shows the strength of our proposed Context Based Sentiment Analysis approach.

Keywords-- **Sentiment Analysis, Opinion Mining, Context Based Sentiment Analysis, SentiWordNet, Word Sense Disambiguation**

Biographical Notes: Srishti Sharma is a PhD research scholar in the Computer Engineering Department at Netaji Subhas Institute of Technology, New Delhi, India. She holds a B.E. degree in Computer Science and Engineering from Maharishi Dayanand University, Rohtak and M.Tech degree in Information Systems from Delhi Technological University, Delhi. Her research interests include Sentiment Analysis and Biometrics.

Shampa Chakraverty is Professor and Head of the Computer Engineering Division at Netaji Subhas Institute of Technology, New Delhi, India. She completed her Bachelor's degree in Electronics and Communication ('83) from Delhi College of Engineering, M.Tech in Integrated Electronics and Circuits from IIT-Delhi ('92) and PhD in Computer Engineering from Delhi University, India. Her research interests include Analysis of Big Social Data and Recommender Systems, E-Learning and E-Governance, Machine Learning and Document Classification, Digital Database Watermarking and Design and Exploration of Multiprocessor Architectures.

Akhil Sharma is a Computer Science Graduate from Netaji Subhas Institute of Technology, Delhi. He is presently working as a Software Development Engineer at Amazon and has previously worked at D.E. Shaw. He is passionate about programming, algorithms and technology.

Jasleen Kaur has done B.Tech from Department of Electronics and Communication Engineering, Netaji Subhas Institute of Technology, Delhi. Ideas that bridge the gap between humans and technology interest her. Passionate to code, she is presently working at Amazon as a Software Development Engineer.

I. INTRODUCTION

As social beings, we often seek opinions from each other on topics on which we have only partial knowledge and value the inputs received from experienced peers. The easy accessibility of the Internet opens up the option of availing the opinion of innumerable others who, though geographically apart, share mutual interests. Today, we can visualize a scenario where a person starts browsing reviews written by others in order to shape his/her decision, whether the object of interest is a product or an investment option or choice of a movie or voting for a candidate in an election. Indeed, there is a huge explosion of 'sentiments' available from social media such as Twitter and Facebook, blog sites, message boards, review websites, and user forums. The unstructured opinionated data available on these websites is a rich source of information that can be tapped to mine real time feedback on any issue.

According to (Liu 2010), an opinion is a quintuple (o_j, f_{jk}, so_{ijkl}, h_i, t_l) where o_j is the target object, f_{jk} is a feature k of the object j, so_{ijkl} is the sentiment value of the opinion of the opinion holder h_i on feature f_{jk} of object o_j at time t_l. The sentiment factor so_{ijkl} may be positive, negative, neutral or it may express a more granular assessment. In terms of Liu's definition, Sentiment Analysis (SA) is the task of determining correctly the sentiment so_{ijkl} at time l as expressed by the author i on a feature k of the entity j which is discussed in any opinionated document d.

Marketing managers, PR firms, campaign managers, politicians, equity investors and online shoppers are direct beneficiaries of Sentiment Analysis (SA) technology and highlight its growing prominence. Significant research work has been carried out in the domain of SA since the past decade. However, the problem of tapping local contextual information in SA has largely gone unnoticed. The main focus of this paper is on developing a scheme for SA that takes into account the textual context presented by words within the same sentence as well as beyond the boundaries of a sentence and allows their subjectivity scores to exercise influence on the sentiment expressed by any given word. We capture the synergy between the words comprising a document and investigate its impact on the overall sentiment conveyed by its contents.

II. RELATED WORK

Some of the earliest works in the area of SA can be traced back to the efforts of (Turney 2002) in discovering sentiments in product reviews. Around the same time, (Pang et al., 2002) focused on extracting sentiments from movie reviews. In the beginning, SA was mostly limited to document analysis to determine their sentiment polarity to be either positive or negative. Currently, SA can be carried out at the document or review level (Grabner et al., 2012);

(Kasper and Vela 2011), (Pang et al., 2002), (Turney 2002); at the sentence level (Celikyilmaz et al., 2010), (Esmin et al., 2012), (Fiadhi et al., 2012), (Shimada et al., 2011), (Wu and Ren 2011) and at the feature level (Singh et al., 2013). Broadly speaking, opinion mining methods may be grouped as (a) lexicon based methods wherein SA is carried out by developing a lexicon or using an existing lexicon that separates the words into positive, negative and neutral (Esuli and Sebastiani 2005); (Esuli and Sebastiani 2007); (Hatzivassiloglou and McKeown 1997) (b) machine learning based methods wherein computers perform SA by training a machine learning based text classifier like Naïve Bayesian, SVM, k-NN, etc. with a suitable feature selection scheme and then using it to classify unknown data (Gamon 2004), and (c) a combination of the above two approaches, as in (Weichselbraun et al., 2010). In this paper, we adopt the lexicon based approach to perform SA at the document level.

Context Based SA Techniques are those that take into account the contextual and semantic knowledge encapsulated within documents. Some approaches where contextual information is utilized to improve the performance of SA systems have been reported. (Nasukawa and Yi 2003) attribute sentiment to specific subjects in the text. They determine the document sentiment by identifying subjective expressions, determining their polarity and the relationship of these expressions to their subject. They label "verbs" as the sentiment transmitters that transfer sentiment from one argument to the other. For example: in the sentence "this book is good", the verb "is" transmits the sentiment extracted from "good" to the subject "book" and associates them. Terms having part-of-speech tags different from "verb" are dealt with in an easier way - they directly transfer their sentiment to the related argument.

Many of the Context Based SA systems focus on a single application domain and are based on the assumption that words always have the same connotation in that domain. For example, the word "*unpredictable"* may invariably imply a negative sentiment when used in the context of a car's engine but is always positive when used in the context of a thriller movie's plot. However, this may not be true always. Consider a camera's description that says "*long* battery life" and "*long* time to focus". One can easily see that within the same domain, the word "long" expresses positive sentiment in the first case and negative sentiment in the other. Thus, it is reasonable to conclude that domain considerations alone cannot suffice for performing context based SA.

(Polanyi and Zaenen 2006) argue that a simple summation of word polarities may not be the ideal way of calculating the sentiment of a document. They identify contextual interactions in a document and categorize the concepts responsible for context switches into Sentence based contextual valence shifters and Discourse based contextual valence shifters. Furthermore, they discuss the case when a document can describe more than one entity/topic/fact and propose that in such cases, the calculation of point of view must be done separately with respect to each entity and must take into account higher order factors such as genre that influences the document structure.

(Weichselbraun et al., 2010) propose a context dependent approach to SA in large textual databases that addresses the problem of polarity shifts of terms depending on their usage context. Most sentiment detection approaches have no way to deal with language ambiguity and ignore it altogether, although the same word may have more than one different meaning and different polarities that change according to the context in which a statement is made. To address this shortcoming, they use Naïve Bayes Method to identify ambiguous terms in text. They develop a contextualized lexicon that stores the polarity of these ambiguous terms, together with a set of co-occurring context terms. They further show that considering the usage context of ambiguous terms improves the accuracy of high-throughput sentiment detection methods.

In (Context Assisted Sentiment Analysis n.d.), the authors suggest that context may affect sentiment at two levels; at the domain level of the content and at the sentence-structure level. Accordingly, they first improve the performance of sentiment-lexicon based methods through domain ontology support. Secondly, they adopt a machine learning approach to extract frequently appearing substrings or expressions and learn the sentiments of these expressions. Finally, they combine these approaches using strategies for voting and chaining.

Taking cue from the above works, we explore how context may affect sentiment across larger units of discourse like sentences, paragraphs and full texts or narratives rather than simply the words and phrases contained within a sentence. The proposed context-based approach builds up on the lines suggested by these researchers and factors in the possibility that indeed, words may have different meanings in different contexts. We also delve deeper to understand how the effective sentiment of words may change due to the presence of other polar words in their vicinity. The proposed algorithm presents a mathematical model to represent this effect and evaluate the enhanced sentiment.

III. A SCHEME FOR CONTEXT BASED SENTIMENT ANALYSIS

The block diagram shown in Fig. 1 describes the general flow of our proposed scheme for context driven SA. We now describe each of its steps.

A. Data Collection

For conducting SA with an acceptable degree of accuracy, we need a corpus of sentiment carrying textual data. As a case study for our experiments, we compiled a dataset of 500 hotel reviews extracted from the hotel review website ‘Tripadvisor.com’. Out of these, 250 reviews were labelled as positive and remaining 250 were labelled as negative. This forms our labelled corpus in Fig. 1. This case study serves to illustrate the usefulness of our approach. It can be easily extended to other domains such as movie review websites, blogs, newspaper articles etc.

B. Pre-Processing

The pre-processing steps are illustrated in Fig. 2. This module begins by converting the entire review text to lower case to obtain a uniform casing and tokenizing the resultant text so that each review document is split into distinct words or tokens. Subsequently, punctuations such as full-stops and commas are removed. Next, stop-words like ‘he’, ‘she’, ‘the’, etc. are removed because such words do not contribute to sentiments. Finally, hyphens are removed from words that are hyphenated. For example: ‘well-appointed’ is changed to well appointed. This is necessary because Word Sense Disambiguation (WSD) considers hyphenated words as undefined and does not assign any sense to them.

C. Word-Sense Disambiguation

Many English words have more than one senses or meanings depending on the context in which they are used. Word Sense Disambiguation is an important pre-requisite to resolve the true implication of a word in its presented context. For example, a simple word like "fly" can mean either “an insect” or “lifted and moving in air”. Thus, "fly" has at least two senses and it is of utmost

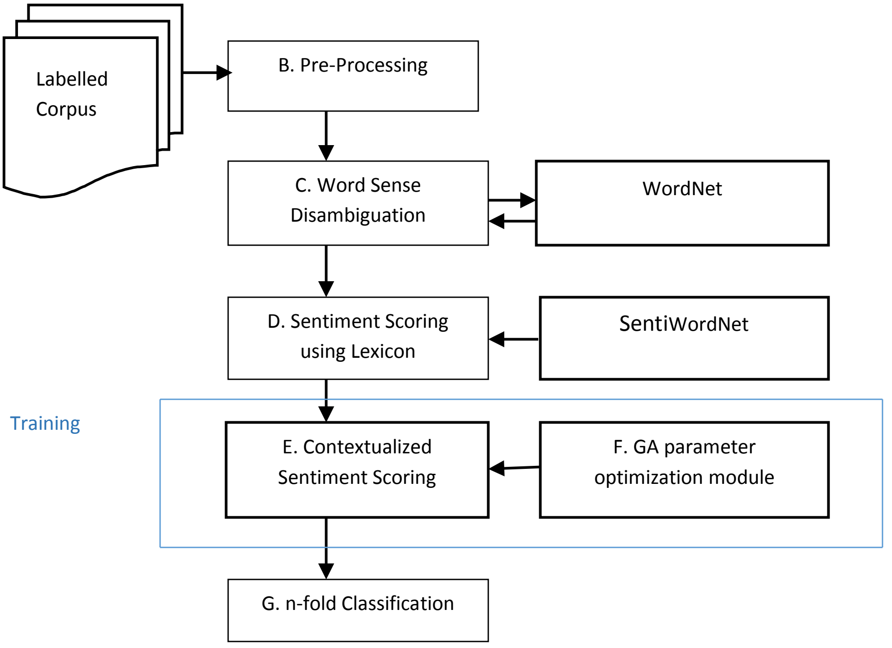

Fig. 1: Block Diagram of the Proposed Context Based SA Approach

importance to correctly identify which sense of a word has been used in a given context.

WSD checks for the meaning of each word in a document, given a window of neighboring words and using this, assigns a sense number to every word as listed in WordNet (Princeton University 2010). This has also been shown in Fig.1. Given a sentence, "Birds fly in the sky", the output after WSD for "fly" is *verb* and the score corresponding to this sense is used. For a sentence, "She was bit by a fly", the output after WSD for "fly" is *noun*. Nouns generally don't carry sentiments. The significance of WSD is that the sentiment score assigned for "fly" is different for these two senses. And since our approach works on sentiment scores of words, this step is critical to the working of the proposed algorithm. Once a word is assigned a sense number, other senses of that word are not considered when disambiguating subsequent words.

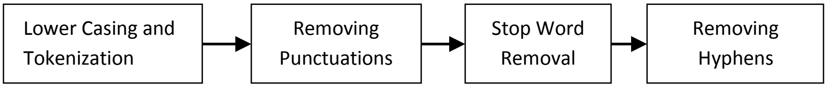

Fig. 2: The Pre-processing steps

D. Sentiment Scoring using Lexicon

After WSD, the next step is to assign raw scores to all the words in a review document. These are the native sentiment scores of all words independent of all other words. As represented by Fig.1, this step is carried out making use of the lexicon SentiWordNet (Esuli and Sebastiani 2006). SentiWordNet is a lexical resource for SA that assigns three scores to each set of synonyms (synset) of WordNet: positivity score, negativity score and objectivity score. All the words in this set of synonyms, called synset or synonym ring are considered to semantically equivalent and interchangeable in some context by WordNet. The sentiment scores assigned by SentiWordNet to any word are in the range *0* to *1* where the sum of all three scores for every word is unity.

The process *GetSentiWordNetScore*(.), described in Fig. 3, takes as input a word *w* and assigns to it a raw score, *rawsc*(*w*) and a label, denoted by *label*(*w*) which is a subjectivity indicator. Word labels are either objective i.e. factual or bearing no sentiment indicated by *obj*, subjective-positive indicated by *1* or subjective-negative indicated by *0*. In Line 1, the process extracts the positive and negative scores of word *w*, represented as *pos_score(w)* and *neg_score*(*w*) respectively, from SentiWordNet. If both the positive and negative scores of a word are zero, the word is labeled as objective *obj* and its raw score is set to zero (lines 2-5). If the positive score is greater than negative score for the word, the word is positive and its label is set to *1* and its positive score is taken as its raw score (lines 6-9). Otherwise, the word is negative, labelled *0* and its negative score is taken as the raw score (lines 10-13).

GetSentiWordNetScore(.) is run taking all the words in a document turn by turn for assignment of raw scores and subjectivity indicators to all words comprising the document. While further processing, none of the words having label=’obj’ are taken into consideration for SA.

E. Contextualized Sentiment Scoring

The proposed approach for context based SA makes use of the interrelationships between words in a review document in order to evaluate their semantic relevance to a particular sentiment. The intuition is that the relevance of a word is calculated by taking into account its intrinsic relevance as well as the relevance of terms that appear within the same context. That is, along with the raw scores of all the words, the proposed algorithm also incorporates context scores of all the words, which are computed based on the neighboring words and their influences on one another.

```
GetSentiWordNetScore(.)

Input: Word w
Output: Native score of w: rawsc(w), Label of w: label(w)
Begin
    Extract pos_score(w), neg_score(w) from SentiWordNet
    If ((pos_score(w)==0) && (neg_score(w)==0)){
        rawsc(w)=0;
        label(w)=obj;
    }
    Else if (pos_score(w) >neg_score(w)) {
        rawsc(w)=pos_score(w);
        label(w)=1;
    }
    Else {
        rawsc(w)=neg_score(w);
        label(w)=0;
    }
End
```

Fig 3: Pseudocode for obtaining native scores and labels of words

The influence exercised by a word on another word is computed using a parametric function called the Influence Function, *IF*. This function computations make use of context. “Context” refers to the text surrounding a given word. The idea is that, in order to measure the relevance of a particular word to a given sentiment, the features of the context in which the word appears, are as important as the word itself. Thus, for any positive word, say, if the context words are also positive, the score of the positive word increases, representing a stronger probability towards positivity and if the context words are negative, the score of the positive word decreases. Similarly, for a negative word, the reverse is true. The context *ctxt*(*w*) of a word *w* is the set of words that surround *w* within the dimensions z_L(left margin) and z_R (right margin). Thus, if

"$. . . u-zl-1, u-zl, . . . u-1$,w, $u+1. . . u+zr, u+zr+1 . . .$" is a text passage of any document, then $ctxt(w) = \{u-zl , ..., u-1, u+1, ..., u+zr\}$. In our experiments, we set z_L as the start and z_R as the end of the document.

Two factors used to calculate the change in the native scores of individual words are:

1) Distance of the neighboring word: Of course, the farther a neighboring word is, the lesser is its impact on the present word. We reasonably assume that the distance of neighboring word affects the present word in an inverse manner.

2) Score of the neighboring words: If the neighboring word is highly subjective, it affects the present word more strongly. That is to say, the absolute value of the sentiment score of neighboring words affects the present word in a direct relationship.

The effects of the absolute score of a neighboring word *u* and its distance from the target word *w* are computed by using two separate influence functions- IF_{score} and IF_{dist} respectively, as given below:

$$IF_{score} = \frac{1}{\left(1+ e^{-(\alpha_1 * rawsc(u)+\beta_1)}\right)} \quad (1)$$

$$IF_{dist} = [1 - \frac{1}{\left(1+e^{-(\alpha_2 * reldist(w,u)+\beta_2)}\right)}] \quad (2)$$

The factor *reldist*(*w*,*u*) denotes the distance between the present word *w* and the neighboring word *u* in terms of number of intervening words, divided by the total number of words in the document. The factor *rawsc*(*u*) is the native score of the neighboring word *u*. The factors *rawsc*(*u*) and *reldist*(*w*,*u*) have either zero or positive values.

Note that the above equations have a sigmoidal form $1/(1+ e^{-x})$ in the R.H.S. This logistic sigmoid is a monotone increasing function which approaches *0* when $x \to -\infty$ and approaches *1* when $x \to \infty$. These functions thus act as two soft switches to enable or disable the effects of the distance and score respectively controlled by the model's parameters α_1, α_2 and β_1, β_2. Parameters α_1 and α_2 control the slope of the distance and score curves respectively. Parameters β_1 and β_2 control the point where the switch assumes a median value ½. As shown in Fig.1, these parameters will be optimized by Genetic Algorithms (GA), which is described later.

The context score of a word, given by *surr*(w,*u*) is the resultant effect after multiplication of the two influence functions. This is represented mathematically in equation 3. We have multiplied the effects of score and distance to obtain the final context score because we want that the context score of a word is high only when both the distance and score effect values are high i.e. when there are other words in the close proximity of the present word having high subjective scores. If the ex-or of labels of u and w is zero, that is, both the words have the same subjectivities, then context score will be added else subtracted from the native score of w.

$$surr(w,u) = \begin{matrix} IF_{score} * IF_{dist} & ;if\ label(w) xor\ label(u) = 0 \\ -IF_{score} * IF_{dist} & ; \qquad\qquad otherwise \end{matrix} \tag{3}$$

Thus, the final revised score of a word will result from the combination of its native score and its context score $\sum_{u\in Q(w)} surr(w,u)$ and is given in equation 4. Here $rawsc(w)$is the native score of word w, $Q(w)$ is the set of words that belong to the context of w, $surr(w,u)$ is the context effect of the surrounding word u in the vicinity of w and $\lambda\in[0, 1]$ is the damping factor.

$$s(w) = [(1-\lambda) * rawsc(w)] + \lambda \sum_{u\epsilon Q(w)} surr(w,u) \tag{4}$$

F. Optimization using Genetic Algorithms

Genetic algorithms (GA) generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover. As shown by Fig. 1, the optimum values of α_1, α_2 and β_1, β_2 used in Equations 1 and 2 for contextualized sentiment scoring are obtained by fine-tuning using Genetic Algorithms. The objective function of the GA is set to maximize the accuracy of the classification task, which is described in Equation 5.

G. Classification

As represented in Fig.1, the last step of the proposed context based SA approach is classifying a review file into positive or negative. To calculate the subjectivity for each file, process *GetFileSubjectivity*(.), explained in Fig. 4 is used. The positive score count and negative score count of the file denoted by *sc_pos*, *sc_neg* are initialized to zero (line 1). Then for every word encountered, its label is checked to determine whether the word is positive or negative. For a positive word, the positive score count *sc_pos* is incremented by the score of the word. For a negative word, the negative score count *sc_neg* is incremented by the score of the word (lines 2-6). At the end, the normalized positive score *normsc_p* is calculated which represents the ratio of positive to total scores and similarly the *normsc_n* represents the ratio of negative to total scores (lines 7-8).

If normalized positive score is greater than the normalized negative score, the file is labelled positive, else it is labelled negative (lines 9-11).

IV. EXPERIMENTS AND RESULTS

All review files are pre-processed as shown in Fig. 2 using the Python Natural Language Toolkit. WSD is performed using the module WordNet::SenseRelate::AllWords in Perl. Next, a SentiWordNet Baseline is established which provides a reference point for the purpose of comparing the performance of the proposed context oriented classification algorithm. The baseline and the context based approach are as described:

A. *Baseline approach*

This approach makes use of the raw scores of the words in a document as obtained from SentiWordNet using process *GetSentiWordNetScore*(.) explained in Fig. 3. The baseline established ignores contextual considerations completely. The subjectivities of files for this approach, are computed replacing each occurrence of the revised score *s(w)* by raw score *rawsc(w)* in process *GetFileSubjectivity*(.). Then, the factors *normsc_p* and *normsc_n* indicate the baseline positive and negative subjectivities respectively.

B. *Proposed Context Based approach*

The Python library Pyevolve is used for fixing the parameter values using GA. The model's parameter values α_1, α_2 and β_1, β_2 in Equations 1 and 2 are fixed keeping a population size=10 and training for 300 generations. Then, context scores of all words are computed using Equations 1, 2 and 3 and then the final revised scores of words, taking into account the raw scores as well as context scores, are computed as described in Equation 4. The subjectivities of all review files are computed using process *GetFileSubjectivity*(.). The factors *normsc_p* and *normsc_n* indicate the positive and negative subjectivities of the review files.

```
GetFileSubjectivity(.)
Input: File F
Output: Subjectivity of file F: Subj(F)
Begin
Set sc_neg=0; sc_pos=0;
    For each word w in F {
    If (label(w)==1)
        sc_pos+= s(w);
    Else sc_neg +=s(w);
    }
    normsc_p=(sc_pos/sc_pos+sc_neg)*100;
    normsc_n=(sc_neg/sc_pos+sc_neg)*100;
    If (normsc_p>normsc_n)
        Subj(F)=pos;
    Else Subj(F)=neg;
End
```

Fig. 4: Pseudocode for Computing the Subjectivity of an Input File

The performance of the proposed algorithm is tested on a total of 500 hotel reviews of varying lengths, extracted from the hotel review website ‘Tripadvisor.com’. A sample review is shown in Fig 5. In order to determine the accuracy of our approach and facilitate comparison with Baseline, we apply 10-fold cross validation, without randomizing the training and test sets. Thus, in each fold the training and test sets are completely disjoint. For all folds we keep a total of 450 review files containing 225 positive and 225 negative files for tuning the parameters α_1, α_2 and β_1, β_2 using GA. The remaining 50 review files are used for testing the performance of the proposed context-based algorithm.

C. Evaluation Metrics

The following well-known metrics were used for evaluation of SA:

Accuracy: The accuracy of classification is as represented in Equation 5.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \qquad (5)$$

TP: True Positive, TN: True Negative, FP: False Positive and FN: False Negative. True positives indicate the number of positive files which are rightly

This hotel/B&B served us great for the one night we were in Rome. It's fairly discrete from the outside, so if you are staying here make sure you know the address of the place and exactly where it is.

The staff are friendly and make a great effort to speak to you in English. The rooms are spacious by European standards and the bed is comfortable. There is free wireless internet but it doesn't work so good in most of the rooms, it's best to use it in the lobby.

Location wise it's a 5 or 10 minute walk to the Colosseum and right next to a metro station from which you can get to anywhere in Rome. It is also a 5 to 10 minute walk to Roma Termini which is where you can get express trains to and from the Airport (cheaper than a taxi).

Breakfast is really nice, pastries, coffee and yogurt etc brought to your room at the time you prefer. We were leaving really early in the morning to get a flight but the staff made our breakfast and brought it to us earlier than they normally would so we could eat that morning. They could have easily just told us it was too early and we wouldn't be getting breakfast, I thought it was a very nice gesture and good service.

Checking in and out is very easy because it's not a very big place- just remember there is an elevator in the building and you don't need to haul your luggage up any stairs, the elevator is a little hidden when you walk into the building.

Fig. 5: A Sample hotel review extracted from Tripadvisor.com

classified as positive. Similarly True negatives indicate the number of negative files which are correctly classified as negative. False positives denote the number of negative files which are misclassified as positive. False negatives indicate the number of positive files that are misclassified as negative.

Precision: Precision, *Pr*, for a positive class is the number of items correctly labeled as belonging to the positive class, (True Positives) divided by the total number of elements labeled as belonging to the positive class (the sum of True Positives and False Positives). This is represented in Equation 6.

$$Pos\ Pr = \frac{TP}{TP+FP} \tag{6}$$

For a negative class, we consider True Negatives and False Negatives instead, as represented in Equation 7.

$$Neg\ Pr = \frac{TN}{TN+FN} \tag{7}$$

Recall: Recall, *Re*, is defined as the number of True Positives divided by the total number of elements that actually belong to the positive class (the sum of True Positives and False Negatives). This is represented in Equation 8.

$$Pos\ Re = \frac{TP}{TP+FN} \tag{8}$$

For a negative class, we consider True Negatives and False Positives instead, as represented in Equation 9.

$$Neg\ Re = \frac{TN}{TN+FP} \tag{9}$$

F1- Score: This is the harmonic mean of Precision and Recall. Positive and Negative F1 are calculated as shown in Equations 10 and 11 respectively.

$$Pos\ F1 = 2 * \frac{Pos\ Pr * Pos\ Re}{Pos\ Pr + Pos\ Re} \tag{10}$$

$$Neg\ F1 = 2 * \frac{Neg\ Pr * Neg\ Re}{Neg\ Pr + Neg\ Re} \tag{11}$$

We also report the macro-level F1 measure, which is the average of the Positive and Negative F1 scores.

The accuracies obtained by the Baseline approach and by the proposed Context based approach, are shown in Table I. The difference between the two is highest in Fold 3 where it is 8% and least in Folds 7 and 8 where the Context and Baseline accuracies are the same. In other folds, the difference between Context and Baseline accuracies varies from 2 to 6%. The Context accuracy is better or equal to the Baseline accuracy for each of the folds. The average accuracies over all the folds for Baseline and Context approaches are 69.6% and 73.2% respectively, thus giving an improvement of 3.6%. Thus, the proposed Context Based SA method shows a significant improvement as compared with the Baseline approach.

Table II compares the performance of the Baseline and the Context Based SA methods on both positive as well as negative reviews. It enlists the average values of Precision, Recall and F1-measure on all evaluation runs for both the Baseline and the Context Based approach. When we compare the Context Based approach with Baseline, we register an improvement of 3.46% in the Precision positive reviews, but at the cost of Recall, which decreases by 2%. Overall, we still record a gain of 1.81% in positive F1-score by using the Context Based approach. For negative reviews, we register an improvement of 9.20% in Recall by the Context Based approach, with a slight loss of 1.45% in Precision. On the whole though there is a significant gain of 7.96% from Baseline F1-score to Context F1-score for negative reviews.

Table I: Classification accuracies of proposed Context Based SA scheme with cross-validation over different folds

	Accuracy in Percent(Baseline)	**Accuracy in Percent (Context)**
Fold 1	60	64
Fold 2	68	72
Fold 3	60	68
Fold 4	72	76
Fold 5	68	72
Fold 6	66	72
Fold 7	74	74
Fold 8	74	74
Fold 9	78	80
Fold 10	76	80
Average	**69.6**	**73.2**

Thus, for positive as well as negative reviews, the proposed Context Based SA approach works better than the Baseline approach. A closer examination revealed that several words with borderline sentimental scores gained strength when augmented with high contextual scores. Likewise, some words with high sentimental scores diminished in strength when surrounded by more objective neighboring words.

Table II: The average Precision, Recall and F1 values after 10-fold Cross-validation for Baseline and Context Based SA approaches

	Baseline Approach			**Proposed Context Based SA Approach**		
	Re %	**Pr %**	**F1 %**	**Re %**	**Pr %**	**F1 %**
Pos	96.00	63.16	76.08	94.00	66.62	**77.89**
Neg	43.20	91.32	57.81	52.40	89.87	**65.77**

V. COMPARISON WITH RELATED WORK

We compare the performance of our approach with the performance of two recent papers that propose alternative SA approaches. The first paper by (Hogenboom et al., 2011) does not make use of any contextual information from the corpus. The authors investigate a specific aspect of SA- negation and its impact on SA. They proposed and compared several approaches to account for negation in SA, differing in their methods of determining the influence of negation keywords. The authors also experimented with different values of sentiment inversion factor that controls to what extent the scores of negated words are inversed. They evaluated the performance of their approaches on a collection of 1,000 positive and 1,000 negative English movie reviews, which have been extracted from movie review web sites (Pang and Lee 2004).

Table III shows the best results reported in (Hogenboom et al., 2011) and those obtained by using our approach on the same corpus. On comparing the F1 scores of both approaches, we find that for positive reviews, the F1 score obtained by our Context Based Approach is 15.29% more than by that obtained by them. For negative reviews, again our F1 score is better by 6.39%. Overall, our context driven SA approach outperforms their negation-feature based SA approach by a significant margin of 10.84%. This clearly establishes our contention that including contextual information in SA improves its performance significantly.

Next, we compare the performance of our approach with that of another context-based SA approach proposed in (Weichselbraun et al., 2010). The authors tested their approach on hotel reviews extracted from Tripadvisor with 10-fold cross validation. The results reported in (Weichselbraun et al., 2010) and those obtained by running our context based approach, are reproduced in Table IV. On comparing the F1 scores, we can see that on an average, for positive reviews, their F1 score is higher by 1% than the F1 score obtained by our approach. On the other hand, for negative reviews, our approach achieves a 3% higher F1 score. On the whole, our approach registers a 1% increase in macro-F1 scores. The improvement is small, but on large datasets, even this value cannot be considered insignificant. The results re-affirm the superiority of our context based SA approach.

The principal advantage of our approach is the flexibility it offers in terms of two parametric functions that show the effects of score and distance. The tunable parameters α and β are optimized for achieving the best possible results for different corpora. On the other hand, in (Weichselbraun et al., 2010) the authors have applied WSD but they have completely ignored the issue of corpus-specific fine tuning of the performance of SA.

Table III: Comparison of F1 scores of proposed context based and alternative SA approach for positive and negative review files

	Positive F1 %	**Negative F1 %**	**Macro F1 %**
SA approach in (Hogenboom et al., 2011)	54.70	51.90	53.30
Proposed Context Based SA	**69.99**	**58.29**	**64.14**

Table IV: Comparison of F1 scores over 10-folds between proposed and alternative Context Based SA approach for positive and negative review files

	Positive F1 %	**Negative F1 %**	**Macro F1 %**
Proposed Context Based SA	78	**64**	**71**
Context Based SA in (Weichselbraun et al., 2010)	**79**	61	70

VI. CONCLUSIONS AND FUTURE WORK

We developed a novel Sentiment Analysis algorithm utilizing the contextual information of the words comprising a document. Using this as a foundation we were able to measure the degree to which the words in a document influence each other and the impact of the dynamics of this relationship on the overall document sentiment. On the basis of the results obtained, we conclude that context plays a very significant role in determining a document's sentiment and should not be ignored. To extend the work, we plan to compute the contextual influence amongst words not only on the basis of their physical distances but also other features such as WordNet distances.

REFERENCES

Celikyilmaz, A, Hakkani-Tur, D and Feng, J (2010), 'Probabilistic Model-Based Sentiment Analysis of Twitter Messages', *in the 2010 IEEE International Conference on Spoken Language Technology, Workshop*, pp 79-84

Context Assisted Sentiment Analysis, Available from:<http://www.ceti.cse.ohio-state.edu/architectures-services/technical-reports/OSU-CISRC-6-09-TR28%20-%20Context-Assisted%20Sentiment%20Analysis.pdf>, [17 March 2014]

Esmin, Ahmed A.A. , Jr., Roberto L. de Oliveira ; Matwin, S (2012), 'Hierarchical Classification Approach to Emotion Recognition in Twitter', In *Proceedings of the 11th International Conference on Machine Learning and Applications, ICMLA*, Boca Raton, FL, USA, Volume 2, pp 381–385,IEEE.

Esuli, A and Sebastiani, F (2005), 'Determining the semantic orientation of terms through gloss classification', *In Proceedings of CIKM-05,* pp 617–624

Esuli, A and Sebastiani, F (2006), 'Sentiwordnet: A publicly available lexical resource for opinion mining' *In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC '06)*, pages 417–422.

Esuli, A and Sebastiani, F (2007), 'Pageranking Wordnet Synsets: An application to opinion mining', *In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL).*

Fiaidhi, J, Mohammed, O, Mohammed, S, Fong, S and Tai hoon Kim (2012) 'Opinion Mining over Twitterspace: Classifying Tweets Programmatically using the R Approach', in *Seventh International Conference on Digital Information Management*, pp 313-319, IEEE

Gamon, M (2004),'Sentiment classification on customer feedback data: Noisy data, large feature vectors, and role of linguistic analysis', in *Proceedings of the 20th International Conference on Computational Linguistics (COLING),* Geneva, Switzerland, Article no. 841

Gräbner, D, Zanker, M, Fliedl, G and Fuchs, M (2012),'Classification of Customer Reviews based on Sentiment Analysis', *19th Conference on Information and Communication Technologies in Tourism (ENTER), Springer,* Helsingborg, Sweden.

Hatzivassiloglou, V and McKeown, KR (1997), 'Predicting the semantic orientation of adjectives', in *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics,* pp 174–181.

Hogenboom, A.; van Iterson, P.; Heerschop, B.; Frasincar, F.; Kaymak, U (2011), ‘Determining negation scope and strength in sentiment analysis’, in *Proceedings of the 2011 IEEE International Conference onSystems, Man, and Cybernetics (SMC)*, pp 2589-2594.

Kasper, W and Vela, M (2011), ‘Sentiment Analysis for Hotel Reviews’, in *Proceedings of the Computational Linguistics-Applications Conference,* Jachranka*, pp. 45–52*

Liu, B (2010), ‘Sentiment Analysis and Subjectivity’, A Chapter in *Handbook of Natural Language Processing*, 2nd Edition, 2010.

Nasukawa, T and Yi, J (2003), ‘Sentiment Analysis: Capturing Favorability Using Natural Language Processing’, in *Proceedings of the International Conference on Knowledge Capture,* New York, USA, pp 70-77

Pang, B, Lee, L and Vaithyanathan, S (2002), ‘Thumbs up? Sentiment Classification using Machine Learning Techniques’, *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. pp. 79–86.

Pang, B. and Lee L (2004), “A Sentimental Education: Sentiment Analysis using Subjectivity Summarization based on Minimum Cuts,” in 42nd Annual Meeting of the Association for Computational Linguistics ACL, 2004, pp. 271–280.

Polanyi, L and Zaenen, A (2006),’ Contextual Valence Shifters’, In *Shanahan J.G., Yan. Qu, and Wiebe J (Eds.), Computing Attitude and Affect in Text: Theory and Applications*, Chap.1 Springer, Heidelberg, pp. 1–10.

Princeton University “*About Wordnet*”. WordNet. Princeton University, 2010. Available from: <http://wordnet.princeton.edu/>, [17 March 2014]

Shimada, K, Inoue, S, Maeda, H and Endo T (2011), ‘Analyzing Tourism Information on Twitter for a Local City’, *First ACIS International Symposium on Software and Network Engineering, SSNE’11,* pp-61-66.

Singh, V.K, Piryani, R, Uddin, A and Waila, P (2013), ‘Sentiment Analysis of Movie Reviews’, *in Proceedings of International Multi Conference on Automation, Computing, Control, Communication, and Compressed Sensing*, Kerala, India, pp 712-717

Turney, P (2002), ‘Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews’, *Proceedings of the Association for Computational Linguistics*. pp. 417–424

Weichselbraun, A, Gindl, S and Scharl, A (2010), ‘A Context-Dependent Supervised Learning Approach to Sentiment Detection in Large Textual Databases’, in *Journal of Information and Data Management, Vol. 1, No. 3, pp 329-342.*

Wu, Y and Ren, F (2011), 'Learning Sentimental Influence in Twitter', *International Conference on Future Computer Sciences and Application*, pp. 119-122