Sindh Univ. Res. Jour. (Sci. Ser.) Vol. 48 (3) 501-506 (2016)

SINDH UNIVERSITY RESEARCH JOURNAL (SCIENCE SERIES)

Automated Sentiment Analysis of Natural Language Text using Machine Learning

I. S. BAJWA[++], H.ISMAIL*, A. H. S. BUKHARI**, R. AMIN**

Department of Computer Science, The Islamia University of Bahawalpur

Received 15th December 2015 and Revised 12nd July 2016

Abstract: This paper presents an approach toclassify sentiment in peer reviews of papers submitted in Journals and Conference such as prepublication peer reviews, written before the paper is published and post publication peer reviews, written after the publication. Although the peer reviews are highly technical but they also contain sentiments. The proposed approach performs automatic sentiment analysis and polarity (Negative, Positive) classification of peer reviews of scientific papers. Our approach finds the sentiment strength of word in the sentence by using term frequency and inverse document frequency weighting and then uses a Markov Logic based algorithm to assign weights to sentiment words based on their strength according to SentiWordNet3.0 and TF-IDF weights and calculating the overall sentiment polarity of the sentence.The results of the used approach prove that our approach is comparatively more affective and accurate as compared to similar approaches.

Keywords: Markov Logic, TF-IDF, Subjectivity, Sentiment, Polarity

1. INTRODUCTION

This paper deals with the peer reviews of sentiment analysis of papers presented in journals and conference. Peer review is the evaluation of creative work by other people of same calibre in the same field in order to improve or maintain the quality of work in that field. In colleges and universities students Peer review is a typical practice that helps students to improve their work and writings. Before and after the paper is accepted and published a lot of Peer Reviews are generated. We can divide these peer reviews in to two categories. Pre-publication peer reviews, written before the paper is published and post publication peer reviews, written after the publication.

Sentiment analysis is a border term (Anger, Happy, Positive, Negative, Suggestive and Neutral) but we restrict to the subset of sentiment, opinions that could be negative and positive (Calvo, 2011). People express their opinions while writing. Opinions are subjective whereas facts are objective. Sentiment analysis is all about finding that subjective part of text and classifying it according to opinions expressed in it (Positive and Negative).One of the major problems of sentiment classification is ambiguity and this ambiguity is present at three levels (Alexander 2013).Opinions are subjective and sentiment is all about finding the opinions of the holder in the text. The peer reviews contains opinions about the subject of person of same calibre in that field.As peer reviews are done to judge the quality of work, so the automatically extracting the opinions from the Peer Reviews is an important aspect. Peer reviews are the evaluations of the work by a person of same calibre in that field. As peer reviews are written by highly professional people, so they are highly technical and objective. But technical terms do carry sentiments as well e.g. the "Current state of the art machine translation is poorly presented …." clearly shows the sentiment.

Reviewers tend to be critical about the ideas, theory or methodology that contradict their own school of thought and are more lenient towards those that match their own. Reviewers are established scientist as result the ideas they favour the established ideas, theories, methodologies and algorithms. As a result they show some sentiments. Conflict of interest is another factor responsible for sentiments to creep in.Detecting the sentiments in Peer reviews of scientific papers has a useful application in Natural Language Processing (NLP) that can automatically evaluate the impact of individual scientist and journals.Peer reviews are mostly neutral because they evaluate some method, idea, theory, algorithm or an approach. Therefore they contain a lot of subjective information. Peer reviews do contains a lot of scientific terminologies, but these terminologies also contain sentiment (Alias, 2012) (Apoorv. (2011).The problem we are focussing in this paper is to take peer review as plain text and give polarity indication as output.First the subjectivity of the peer review is detected and separated from the objective information. Then the subjectivity is tested for the sentiment and if some sentiment is detected then the polarity classification (Positive, Negative) (Apoorv 2012) of that sentiment is done. We consider neutral opinion as objective.

[++]Corresponding Authors:. IS. BAJWAEmail: imran.sarwar@iub.edu.pkhuzefaismail@yahoo.com, ahsbukhari02@gmail.com

**Sindh Institute of Management and Technology, Gulistan-e-Johar, Karachi

*** Baluchistan University of IT, Engineering, and MS, Quetta

3. PROPOSED APPROACH

For sentiment classification for peer reviews our approach works in three steps. First is finding the sentiment strength of word in the sentence. For this we use SentiWordNet3.0 Second, we use term frequency and inverse document frequency weighting. The third step is based on Markov Logic, designing an algorithm that assigns weights to sentiment words based on their strength according to SentiWordNet3.0 and TF-IDF weight and calculating the overall sentiment polarity of the sentence (King, M., 1996).Negations are handled separately. They are the words that reverse the polarity of sentence or the word. NegEx[1]is use for deducting negations.At the end overall polarity of the Peer Review is determined on the bases of aggregate scores assigned to negative and positive terms by SentiWordNet3.0 and TF-IDF weight and assigning the review to the class (Negative, Positive) (Prem, 2009) with the highest score.

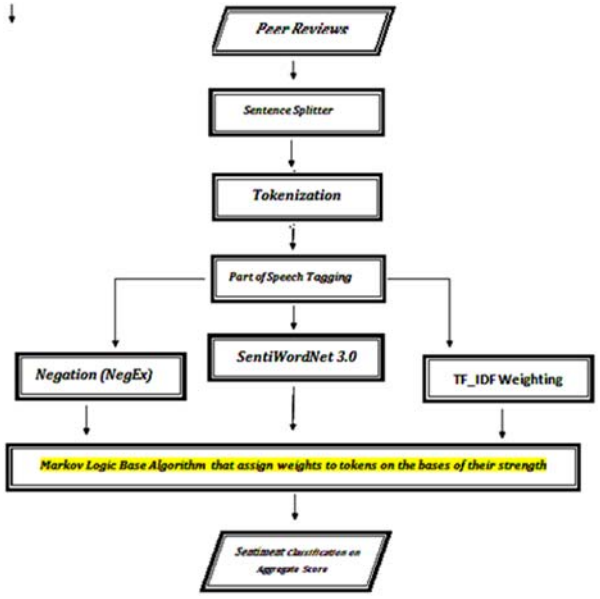

Fig. 3.1: Detailed Architecture of used approach

3.1 Data Preprocessing

Data is presented in plain text before applying the adopted architecture. Data pre-processing is done to reduce the noise. Question words and special characters do not add to polarity, hence they are removed from data to reduce the overhead. Following linguistic techniques are implied.

Sentence Splitting: AT this stage, input text is segmented into sentences. We use GATE[2]splitter for this which uses list of abbreviations as a help to determine the end of the sentences to differentiate the full stops from other kind of full stops.

Tokenization: A process of dividing or chopping up a sequence of characters in tokens is called tokenization. Tokens are set of character important as group. We can say that token is group of characters, grouped in a way that is important as semantic unit for processing. The major question in tokenization is the question what the tokens are? There are different techniques to distinguish tokens that include: RE, sequence of characters known as flags, "delimiters" some special characters like punctuations. Normally tokenization happens at word level but it is difficult to define what word is To kenizer simply relies on hand written rules like all adjoining characters are the part of token. They are separated by white spaces or punctuations.

3.2 Part Of Speech Tagging

Part of speech tagging is a process of tagging of word in text into one of part of speech based on contextual information in a sentence and also called the grammatical tagging. There are eight English parts of speech, Noun, Verb, Adverb, Adjectives, Preposition, Pronoun, Interjection and conjunction. Putting a word into one of the categories is called tagging (Mike, 2010). The main objective of part of speech tagging is assigning a part of speech tag to a word showing its syntactic categories. In some way part of speech tagging is a process of word sense disambiguation as the tagger algorithm gives most likely tag to a word based on contextual information. In other words part of speech tagger help determine the real meaning of the word in its context (Morante, 2011). There are different tag sets and the tag retuned depends upon the tag set used. The most famous tag set use is of Penn Tree Bank Tag Set.

3.3 Sentiment Analysis

Opinions lexicons are used to depict the sentiment inclination of words. The most widely used relation among the SynSet is hyponymy. It relates general SynSet to increasingly more specific ones. Hyponymy relations are transitive in nature. Word Net2.1 differentiates between type and their instances like chapter is the instance of the book. The relation of the part to the whole also holds in SynSets. The parts are acquired from their superior and thus the inheritance is downward not upward. The verbs synonyms are also arranging in hierarchal manners. The verbs at the bottom of hierarchy are more specific in expressing event or action. Adjective are arranged in antonym manner. The opposite adjectives in term are linked to other words of similar meanings. Relational adjectives indicate the noun from which they are derived. There are very few adverbs in WordNet2.1 as they are directly reduced from adjectives

3.3.1 Word Sense Disambiguation

In WordNet2.1 the synonyms set of each word has brief textual description about the word used in what sense called gloss. As we know the SentiWordNet3.0 gives synonyms set of word net three numerical values ranging from 0.0 to 1 and in WordNet2.1 each word is described with all its senses. For example, the word wrong has nine senses as adjectives two senses as noun and one sense as adverb as shown in **(Table 3.1).**

Table 3.1: Showing different uses of same words (Senses) as a POS from WordNet2.1

Word/POS	Adjective	Noun	Adverb
Wrong	9 Senses	2 Senses	1 Sense
Bad	14 Senses	1 Sense	2 Senses

Each word has multiple senses having different positive, negative and objective score to identify the positive or negative strength of a word. We first need to perform the word sense disambiguation. For this a simple algorithm implementation can be done using the NLTK (Python Library of NLP) but we implied the negative positive score of each word in SentiWordNet3.0 lexicon we calculated the average of its presence as a part of speech categories (adjective, adverb, noun and verbs).For example the word "wrong" has nine senses as adjective two senses as noun, one sense as adverb and one sense as verb. Therefore word wrong has four positive and four negative scores **(Table 3.2).** According to the part of speech categories mentioned above.

Table 3.2: Showing average positive and negative score of all the senses of word wrong from SentiWordNet3.0

Category	Average Positive Score	Average Negative Score
Adjective	0.055	0.653
Noun	0.0	0.75
Adverb	0.25	0.0
Verb	0.0	0.75

3.4 Term Frequency and Inverse Document Frequency

The second step of our design is assigning the weights to words according to their importance in the document. For this we use term frequency inverse document frequency (TF-IDF).

3.4.1 Term Frequency

In simple sense, term frequency means the number of occurrences of a word or term in that particular document. Term frequency actually measures how frequent term or a word in the document. Documents vary in their lengths so the term or the word may appear more than one time in a document. Thus the term frequency of a particular word in a particular document can be calculated as:

TF (W) = Number of occurrences of a word in the document

Total number of the words in that document while calculating the term frequency every term is considered equally important but this is not the case.

3.5.2 Inverse Document Frequency

Inverse document frequency measures the importance of the term or word as compared to term frequency which just measures the occurrence of term or word.

IDF (W) = Log (Total number of sentences in a document / Number of sentenceshaving that term in a document)

3.5.3 TF-IDF (Term Frequency- Inverse Documents Frequency)

TF considered all term or words equally important. However some terms or words such as "is", "as", "was", "of" may appear a lot of time but has very little importance. TF-IDF working is based on finding the relative frequency of words or terms in that particular document in comparison to inverse proportion of that word over the entire document. Now, combining TF and IDF we produce a combine weight for each terms or word in a document. The weight assigned to a term t in a document d is given by:

TF-IDF (t,d) = TF(t,d) * IDF(t)

3.6 Markov Logic

The two major problem of machine learning are complexity and uncertainty. Probability deals well with uncertainty and for complexity first order logic is used. This combination is used to learn and draw conclusion form representation language in finite domain. Natural languages are non-liner by nature whereas first order logic is well suited for liner data.Markov Logic is the combination of probability and first order logic (KB) in finite domain. This approach is purely based on semantic technology and supported by Context Awareness. Followingis the way, Markov network represents n^{th} joint distribution:

$$P(X=x) = \frac{1}{Z}\prod_{k}\phi_k(x_{\{k\}}) \qquad (1)$$

Here, the joint distribution (Pearl, 1988) of a model is represented as a set of variables i.e. X ϵ (X1,X2, ….,Xn). In a typical network of Markov Logic, a set of pair (Fi ,wi) is used to represent a predicate and a predicate in first order logic is represented by Fi and a

real number depicts wi that is weight of the predicate/formula. To update the weights of the used formula, statistical relational learning approach is incorporated by combining probability with the traditional first-order logic. Here, a typical MLN (Markov Logic Network) with a set of weights and formulas can be represented as below:

$$P(X=x) = \frac{1}{Z}\exp\left(\sum_j w_j f_j(x)\right) \qquad (2)$$

The weights of the formulas are dynamically updated by using diagnolized Newton Method. Here, the weight update formula is

$$\boldsymbol{w = w + D^{-1}g} \qquad (3)$$

First order knowledge base is regarded as hard constraints on a set of possible words. The probability of a word becomes zero if it violates even one formula. The main idea of Markov Logic is to relax these constraints. If the formula is violated in one word of KB then it is less probable but not impossible. Weight is associated with each formula which indicates its strength. Markov Logic works on the following main issues. Logical knowledge base is very hard constraint on different possible conclusions. Relax the restrictions when predicate is violated. Chances are less but not impossible. Violation of constraints does not mean totally wrong. Weightage to the formulas on the basis of the constraints followed higher weight more constraint followed and less weight means more constraints violated. Acceptance and rejection does not mean hundred percent. We use Markov Logic in our sentiment analysis of Peer Reviews. SentiWordNet3.0 gives different positive, negative and objective score to a word ranging from 0.0to 1.Using Markov Logic to design an algorithm that gives weight to these tokens.

4. TOOL AND EXPERIMENTS

Basically the stated design has two phases; first phase is all about data pre-processing which is actually a lexical data processing. In the second phase pre-processed is checked for sentiments and polarity classification. A tool is designed that automatically detect the sentiment from text (English language) and determine its polarity. Basically it is a sentiment detection and polarity classification tool. The input, processing and output details are given below. The input for SDP-Classifier is the text in English language. The user gives the input in the form of a text file it can be word file as well.After the noise reduction the

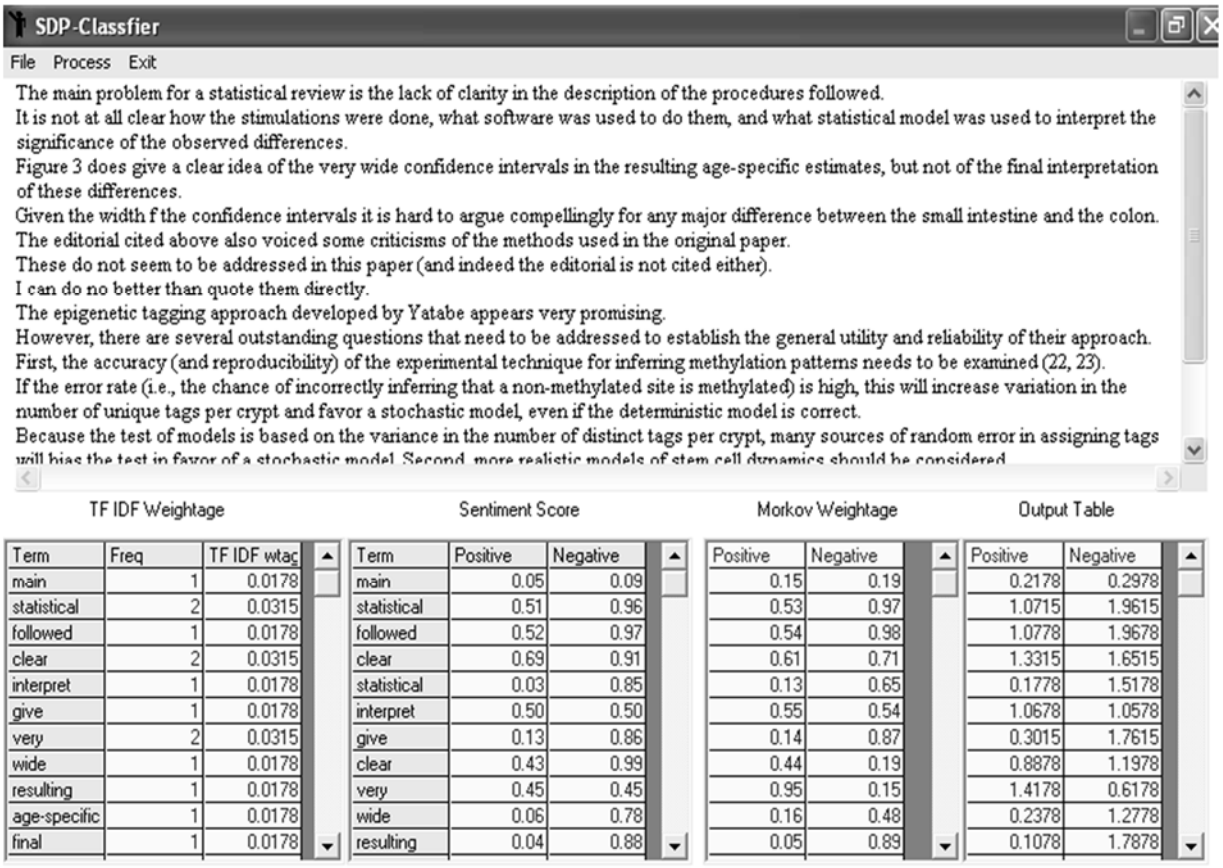

Fig. 4.3 Screen shoot of SDP-Classifier

second module of SDP-Classifier, take the input data (noise free) and calculate the weight age of terms on the bases of their presence in the document. In **(Table 4.2),** the TF-IDF score of selected tokens is shown for their negative and positive values.

Table 4.2 Showing the calculated negative, positive score

Tokens	TF-IDF weightage	Sentiments Strength		Markov Weightage
		Positive	Negative	
poor	0.0071	0	0.625	0.3
wrong	0.0047	0	0.752	0.6
difficult	0.0047	0	0.63	0.8
incorrect	0.0047	0	0.82	0.9
simple	0.0047	0.52	0.1	0.6
correct	0.0047	0.625	0	0.8
hard	0.0047	0	0.45	0.3
missing	0.0097	0	0.53	0.7

This chapter gives details of implementation of SDP-Classifier and shows its input, output and processing detail. The input is in the form of text (English language text). Lexical data processing is done on the input text and in the second phase sentiments are calculated and further weight ages (TF-IDF, Markov) are assigned to the tokens (words). At the last stage aggregate score is calculated for polarity classification of document

5. RESULT AND DISCUSSION

A human expert is consulted to annotate the documents and identify the terms in documents considered to be carrying sentiments. The resulting list of terms for each document was considered gold standard for that document. A Tp (true positive), Fp (false positive) and Fn (false negative) terms are shown in the **(Table 5.1)**. There were 19 terms in gold standard list of that document.

Table 5.1 showing results detected by SDP-Classifier

No	Term Class	Gold List	Tp	Fp	Fn
1	Adjectives	9	5	2	2
2	Adverbs	8	4	3	1
3	Verbs	2	2	0	0
	Total	19	11	5	3

The above **(Table 5.2)**. shows that there are total 19 terms in gold standard list and designed system identified 16 terms and out of which 11 are correct (true positive), 5 are incorrect (false positive) and 3 are missing (false negative).

Table 5.2 Recall and Precision calculated by SDP-Classifier

Terms	Gold List	Tp	Fp	Fn	Recall%	Precision %
Text Document	19	11	5	3	78.35	68.75

The above **(Table 5.3)**. shows the recall and precision of the designed tool for the text document. The average recall is 78.35% and precision 68.75%. The initial results of performance evaluation are quite encouraging and support the methodology adopted. Four unseen text documents of varying lengths were tested for sentiment detection and polarity classification. Calculated recall, precision and f-measure are shown in the table below.

Table 5.3 Results of four test documents of SDP-classifier

Terms	Gold List	Tp	Fp	Fn	Recall%	Precision %	F-measure
Doc1	21	12	4	5	70.58	75.00	72.72
Doc2	17	10	3	4	71.42	76.92	74.07
Doc3	27	18	3	6	75.00	88.57	79.93
Doc4	13	8	2	3	72..72	80.00	76.18
Average					**72.33**	**80.12**	**75.72**

The average f-measure is 75.72 which are good for initial experiments. The **(Fig. 5.1,2)** below shows the evaluation results of SDP-Classifier.

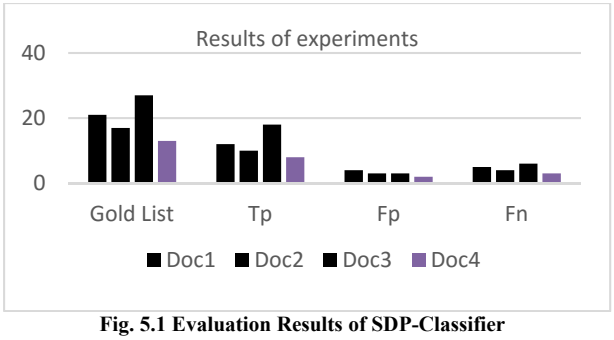

Fig. 5.1 Evaluation Results of SDP-Classifier

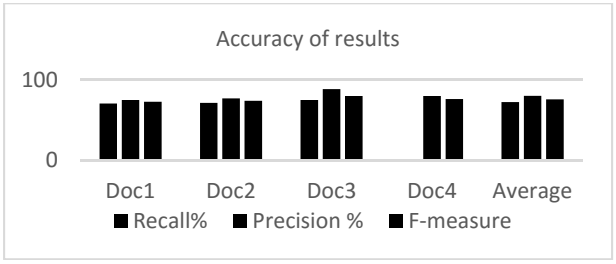

Fig. 5.2Accuracy and precision of the achieved results

The**(Fig. 5.1 and Fig. 5.2)** show Tp (true positive), Fp (false positive) and Fn (false negative) evaluation results measured by SDP-Classifier. Magenta colour line shows the gold standard list which is highest for document 3. Red colour line shows the false positive (incorrect results detected) which is highest for document 1. Green line shows the false negative results (missing results) which are highest for document3. In this chapter the SDP-Classifier is tested and the results are evaluated using three matrices recall, precision and

f-measure. Results show that recall is 72.33% and precision is 80.12%. Recall describes the completeness (how relevant are the results) and precision shows the quality or accuracy (more relevant results then irrelevant results). F-measure is the harmonic mean between recall and precision.

6. CONCLUSION

The main objective of the research was to design an automatic system to detect the sentiment from the text and on the bases of these sentiments decode the polarity of that particular document. Experiments are done and results are calculated carefully. The outcome is quite promising that support our design and techniques used in detecting the sentiment and deciding the polarity. The results are good but there are several directions we are planning to investigate in future. Handling negations with respect to their context in a sentence as the impact of the negation is far from the preceding words.

REFERENCES:

Calvo, S. M. (2011) Sentiment-Oriented Summarization Peer Reviews. Aied 2011, Lnai 6738 (Spring - Verlag Berlin Heidelberg 2011), 491-493.

Alexander H. D. B. (2013). Exploiting Emoticons in Sentiment Analysis. Acm 978-1-4503-1656-9/13/03.

Alias, A. T. (2012). Three Class Sentiment Analysis Adopted To Short Text. Proceedings of the Xxviii Conference Of The Spanish Society For Natural Language Processing (Sepln). Castelló (Spain): Gmt – Group de Recerca En Teconolgies Media.

Apoorv A.. (2011). Sentiment Analysis Of Twitter Data. Lsm '11 Proceedings Of The Workshop On Languages In Social Media (Pp. 30-38). Association For Computational Linguistics Stroudsburg, Pa, Usa ©2011.

Apoorv A., (2012). End To End Sentiment Analysis Of Twitter Data. Proceedings Of The Workshop On Information Extrection And Entity Analysis On Social Media Data, (Pp. 39-44). Mumbai.

Cardie, Y. C. (2008). LerningWith Compositional Semantics As Structural Inference For Subsentential Sentiment Analysis. Proceedings Of The 2008 Conference On Empirical Methods In Natural Language Processing, (Pp. 739 - 801). Honolulu.

Efthymios T. W. (2011). Twitter Sentiment Analysis: The Good The Bad And The Omg. Proceedings Of The Fifth International AAAI Conference On Web Log And Social Media, 538 - 541. Edinbrugh.

ErikBoiy, P. H. (2007). Automatic Sentiment Analysis In Online Text. Proceedings Elpub2007 Conference On Electronic Publishing. Vienna.

Andrew H. J. C. Schwartz, (2013). Personality, Gender, And Age In The Language Of Social. (U. O. Tobias Preis, Ed.) Plos One, 8(9).

Andrew H. J. C Schwartz, (2013). Toward Personality Insight From Language Exploration In Social Media. Retrieved From Association For Advancement For Artificial Intelligence: Www.Aaai.Org.

Hassan S. (2012). Allevating Data SparsityFor Twitter Sentiment Analysis. Ceur. Vol. 838.

King, M. (1996). Evaluating natural language processing systems. Communications of the ACM, 39(1), 73-79.

IqraJaved, H. A. (2013). Opinion Analysis Of Bi-Lingual Event Data From Social Networks. National University Of Science And Technology, Department Of Computer Software Engineering, Islamabad.

Kun-Lin L. (2012). Emoticon Smoothed Language Models For Twitter Sentiment Analysis. Department of Computer Science And Engineering. Shanghai: Association For The Advancement of Artificial Intelligence (www.aaai.Org). Last accessed: 21 June, 2015

Lee, B. P. (2004). A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based On Minimum Cuts. Acl-2004. New York.

Mike T.. (2010). Sentiment Strength Detection In Short Informal Text. Journal Of The American Society For Information Science And Technology, 61(12), 2544–2558.

Morante, R. S. (2011). Corpus-Based Approaches to Processing The Scope of Negation Cues: An Evaluation of the state of the Art. Ninth International Conference on Computational Semantics (Iwcs 2011).

Namrata G.. (2007). Larg Scale Sentiment Analysis For News And Blogs. Icwsm, Boluder.

Onur, B. B. (2012). Large Scale Sentiment Analysis For Yahoo Answers. ACM 978-1-4503-0747 5/12/02, 633 - 642.

Prem M. (2009). Sentiment Analysis of Blogs By Combining Lexical Knowledge With Text Classification. Acm 978-1-60558-495-9/09/06.