Opinion within Opinion: Segmentation Approach for Urdu Sentiment Analysis

Muhammad Hassan and Muhammad Shoaib
Department of Computer Science and Engineering, University of Engineering and Technology, Pakistan

Abstract: *In computational linguistics, sentiment analysis facilitates classification of opinion as a positive or a negative class. Urdu is a widely used language in different parts of the world and classification of the opinions given in Urdu language is as important as for any other language. The literature contains very restricted research for sentiment analysis of Urdu language and mainly Bag-of-Word model dominates the research methods used for this purpose. The Bag-of-Word based models fail to classify a subset of the complex sentiments; the sentiments with more than one opinion. However, no known literature is available which identifies and utilizes sub-opinion level information. In this paper, we proposed a method based on sub-opinions within the text to determine the overall polarity of the sentiment in Urdu language text. The proposed method classifies a sentiment in three steps, First it segments the sentiment into two fragments using a set of hypotheses. Next it calculates the orientation scores of these fragments independently and finally estimates the polarity of the sentiment using scores of the fragments. We developed a computational model that empirically evaluated the proposed method. The proposed method increases the precision by 8.46%, recall by 37.25% and accuracy by 24.75%, which is a significant improvement over the existing techniques based on Bag-of-Word model.*

Keywords: *Sentiment analysis, urdu natural language processing, social media mining, urdu discourse analysis.*

Received December 7, 2014; accept January 20, 2016

1. Introduction

Urdu is the official language of world sixth largest country, Pakistan, and more than sixty-six million people communicate in this language. Urdu borrows a considerable number of words from other languages like Sanskrit, Persian, Arabic, and English; this unique behavior of the language results in a rich morphology and complex grammar. Thus, the computational tools available for other languages [2] are not sufficient for Urdu, and it demands a more specialized set of tools; especially tools for sentiment analysis.

A sentiment is a text sentence that contains subjective beliefs, opinions, and emotions. A sentiment contains single or multiple opinions about same entity or different entities; these various opinions within the sentiment are referred as sub-opinions. Sentiment analysis classifies a sentiment into a positive or a negative class. Literature on English language sentiment analysis provides different types of methods. Liu [9] grouped these methods into four broad categories: namely dictionary or Bag-of-Words (BoW), supervised and unsupervised learning, rule-based, and discourse-based methods. However, the literature contains very restricted research for sentiment analysis of Urdu languages and mainly BoW based models dominate the research methods used for this purpose. BoW model calculates total negative words and positive words of a sentiment with the help of a dictionary. The dictionary contains an entry for each word of sentiment, its Part-of-Speech (POS) tag and orientation (word is positive or negative). Let P_{total} are total positive words and N_{total} are total negative words in a sentiment; the BoW classifies the sentiment in positive, negative or neutral using following function.

$$class(P_{total}, N_{total}) = \begin{cases} positive & \text{if } P_{total} > N_{total} \\ negative & \text{if } P_{total} < N_{total} \\ neutral & \text{if } P_{total} = N_{total} \end{cases} \quad (1)$$

- *Example 1*: wo aik acha $^{(+)}$ aur bahdar $^{(+)}$ larka ha.

(وہ ایک اچھا اور بہدارلڑکا ہے) (He is good and brave boy)
In example 1. Both acha $^{(+)}$ and bahdar $^{(+)}$ are positive words; so P_{total}=2 and N_{total}=0, BoW estimates negative polarity for sentiment.

BoW based models worked perfectly for the sentiment that class is according to the frequency of positive or negative words. However, BoW model failed to classify complex sentiment; the next section defines this type of sentiments.

Complex Sentiments: We categorized a sentiment as complex sentiment if it belonged to one of the following three types:

- *Type 1*: Total positive and total negative words are equal, but the sentiment class is not neutral.
- *Type 2*: Total positive words are greater than total negative words, but the sentiment class is negative.
- *Type 3*: Total negative words are greater than total positive words, but the sentiment class is positive.

Examples 2 and 3 are complex sentiments and BoW model fails to classify them.

- *Example 2*: Afradi aik acha $^{(+)}$ aur fit $^{(+)}$ allrounder ha magar ess ka kaya fida (+) agar wo Pakistan ko match nahi jatwa (-) sakta.

(آفریدی ایک اچھا اور فٹ الرونڈر ہے مگر اس کا کیا فایدہ اگر وہ پاکستان کو میچ نہیں جتوا سکے)

(Afridi is good and fit all-rounder, but what of his use, if he cannot win match for Pakistan)

- *Example 3*: Agarcha pakistan cricket team ko media Tanqeed (-) ka nashna banata rahta haa. Magar ess martaba dubi tower maan ess naa Kamal (+) ker daya.

(اگرچہ پاکستان کرکٹ ٹیم کو میڈیا تنقید کا نشانہ بناتا رہتا ہےلکین اس مرتبہ دبئی ٹور میں اس نے کمال کر دیا)

(Although the media criticized Pakistan cricket team all the time, but they gave an excellent performance during Dubai tour.)

Example 2 contains three positive and one negative word; the BoW model classifies this sentiment in a positive class, but the actual polarity of sentiment is negative. Example 3 contains one negative and one positive word; the BoW model classifies this sentiment in a neutral class, but the actual polarity of sentiment is positive.

Thus, new techniques are required to extend the BoW capabilities to classify the complex opinions. The objective of the study is to propose a method that performs Urdu sentiment analysis: the method classifies the complex opinions into the negative or the positive class using sub-opinion level information. The proposed method classifies sentiment into three steps; first step segments the sentiment into two fragments using a set of hypotheses. The next step calculates the orientation scores of these fragments independently and final step estimates the polarity of the sentiment using scores of fragments. We developed a computational model that empirically evaluated the proposed method. The proposed solution increases the precision by 8.46%, recall by 37.25% and accuracy by 24.75%, which is a significant improvement over the existing techniques based on BoW model.

We have organized the paper into five sections. Section 1 gives the introduction of the paper; section 2 presents related work; section 3 provides details of the proposed solution; section 4 contains the experimental details and results; finally, section 5 concludes the paper.

2. Literature Survey

We have divided the related work in two sections, first section contains literature survey of English language and second section contains related work of Urdu language.

2.1. English Language Text Sentiment Analysis

Discourse based methods, used for english language sentiment analysis, are more relevant to our work; this section includes the literature survey related to these methods. Asher *et al*. [4, 5] investigated discourse and rhetorical relations for a sentiment and used shallow semantic features to classify the sentiment. Zhou *et al*. [20] proposed rhetorical structure theory based scheme and developed a supervised semantic sequential representation learner. Somasundaran *et al*. [12] introduced opinion-frames and suggested a graph-based approach [13] that combined these opinion-frames with collective classification framework [7]. Zrin *et al*. [21] presented a Markov logic-based framework to integrate polarity scores from different sentiment lexicons and neighboring segments. Mukherjee and Bhattacharyya [10] used the discourse relations to target the web-based applications that dealt with noisy, unstructured text, like the tweets. Taboada *et al*. [16] presented an approach based on RST and topic classification. Weibe *et al*. [19] identified, for each word, the strenght of subjectivity clues in the surronding context. Then, these clues were used to perform opinion recognition. Thelwall *et al*. [17] introduced Senti strength algorithm to extract sentiment strength from informal English text. They exploited de facto grammars and spelling styles of cyberspace. Turnery and Littman [18]. introduced a method for inferring the semantic orientation of a word from its statistical association with a set of positive and negative paradigm words.

2.2. Urdu Language Text Sentiment Analysis

Afraz *et al*. [1] discussed the approach that recognized the subjective entries in the lexicon through (either positive or negative) and intensity (the force of the orientation). Syed *et al*. [14] preprocessed the sentiment [3] for normalization and segmentation and then applied shallow parsing to classify sentiments. Mukund and Srihari [11] presented a method for polarity classification of code-mixed data based on structural correspondence learning for domain adaptation. Irvine *et al*. [8] proposed a method based on Hidden Markov Model (HMM) to identify the polarity of short messages written in Roman Urdu. However, we remained unable to find any work that utilized the sub-opinion or sentence level information for sentiment analysis of Urdu language. Bilal *et al*. [6] applied machine learning techniques to train the model for classification of opinions given in Roman Urdu. However, this technique improved the classification task but it required labeled examples for prior training of data.

3. Proposed Classifier (SEGMODEL)

We explain the following terms, as these are frequently used in the remaining paper, before discussion of proposed solution.

- *Orientation*: Orientation of a word tells whether the word contains positive (+1), negative (-1) or neutral

(0) thought.

- *Score*: Score is an integer value that can be positive, negative or zero.
- *Dictionary*: Dictionary, *D* is a file that defines the orientation and POS tag of words.
- *Orientation function*: *Orientation (w)* is a function that finds the orientation of *w* from *D* and returns +1 or -1.
- *Polarity function*: Let polarity be a function defined as:

$$polarity\,(score) = \begin{Bmatrix} positive & \text{if } score > 0 \\ negative & \text{if } score < 0 \\ neutral & \text{if } score = 0 \end{Bmatrix} \qquad (2)$$

- *POS function*: Let *POS (W)* is a function that takes a word as input and returns its part-of-speech tag as defined in *D*.
- *Sentiment*: Let S is the input sentiment with *n* words in linear order:

$$S = \{W_1, W_2, W_3 .. W_n \mid W_i \in D\} \qquad (3)$$

The proposed algorithm classifies the given sentiment *S* into positive, negative or neutral class. The algorithm involves three steps: sentiment segmentation, polarity score calculation, and sentiment polarity identification. First, the classifier segments the sentiment *S* into two fragments S_1 and S_2. The next step calculates the score of both S_1 and S_2. Finally, the algorithm determines the polarity of sentiment *S*. The following sections explain each step in detail.

3.1. Sentiment Segmentation

We introduced the concept of segmentation word to segment the sentiment.

- Segmentation Word (SW_i): SW_i is a word at position i within the sentiment *S* which segments the sentiment into two fragments such as:

$$\begin{aligned} S_1 &= \{W_1, W_2, W_3 .. W_{i-1}\} \\ S_2 &= \{W_{i+1}, W_{i+2}, W_{i+3} .. W_n\} \\ S &= S_1 + SW_i + S_2 \end{aligned} \qquad (4)$$

In Example 2, let magar (مگر) is selected as SW_i then it segments the sentiment into following two fragments.

- First fragment S_1: “Afridi aik acha player ha”.(آفریدی ایک اچھا پلیئر ہنے)
- Second fragment S_2: “Pakistan ko match nehi jatawa sakta”.(پاکستان کو میچ نہیں جتواسکتا)
- *Sub-opinions*: SW_i segments the sentiment into two fragments; we called each fragment a sub-opinion. These sub-opinions contain positive, negative or neutral polarity.

This step selects the SW_i and then partitions the sentiment *S* into two sub-opinions S_1 and S_2.We explain following terms before discussion of SW_i selection process.

- Haruf-Ataf (حرف عطف): Every language has a different type of conjunctions that combines sentences or clauses together; in Urdu, Haruf Ataf (حر ف عطف) is a type of conjunction. These words connect the two clauses or two sentences. In Example 2 the word *magar (مگر)*, and in Example 3 the word *laken (لکین)* is Haruf-Ataf. These words are further divided into sub-types but for this paper we are only interested in following sub-type.

Haruf Shaart u Jaza (حرف شرط و جزا): A set of two words that joins conditional and resultant clauses are called Haruf Shaart u Jaza; these words are sub type of Haruf-Ataf. Haruf-Shart (حرف شرط) indicates the conditional part and Haruf-Jaza (حرف جزا) refers to the resultant clause. Examples of Haruf-Shart are agar (اگر), jo (جو), agarcha (اگرچہ) and examples of Haruf-Jaza are tu (تو), tub (تب), and ess laya (اسلیے).

In our opinion, Haruf-Ataf is the best candidate for segmentation word (SW_i) as it connects two sentences. Urdu contains a large number of Haruf-Ataf (حرف عطف) but we shortlisted a set of frequently used words (Table 1). The H1 hypothesis summarized above discussion.

H1: Let $W_i \in S$ and $POS(W)$ is Haruf-Ataf (حرف عطف) then W_i is selected as segmentation word provided both H2 and H3 are satisfied.

Table 1. List of selected haruf ataf.

Type	Haruf
Haruf-Ataf	aur اور, woo وہ, pherپھر , neezنیز , kerکر, ka کے , yaan یاں , kaya کیا
	chaho چاہو , ka کا , yaan tu یاں تو, chahyچا ہیے
	magar مگر, magar haan مگر ہاں, perپر , laken لیکن albata البتہ. sawa سوا, alwa علاوہ, kyn ka کیونکہ , ess laya ka اسلے کہ , ess wasty ka اس واسطےکہ , taa kaتا کہ
Haruf-Shaart	agar اگر , jo جو, agarchaاگرچہ , jub taak جبتک and choon ka چونکہ
Haruf-Jaza	tu تو, so سو, tub تب , aur اور and ess laya اسلے.

The next section discusses the exceptional cases of H1.

- *Exceptional Cases*: We have identified two exceptional cases.

Case: A set of words, belongs to Haruf-Ataf, also plays different roles other then connecting two sentences. The other roles include connecting two words and using as proposition.

- *Example 4*: wo naak aur bahdar larka ha وہ نیک اور بہادر لڑکا ہے (He is noble and brave boy).
- *Example 5*: yea kitab maaz per rakh doo (یہ کتاب میز پر رکھ دو) (put this book on the table).
- *Example 6*: ess ka camera tahk kaam nahi kerta aur bettary b tahk nahi ha.

اس کا کیمرہ ٹھیک کام نہیں کرتا اور بٹری بھی ٹھیک نہیں ہے
(its camera and battery is not working).

In Example 4 *aur (اور)* is connecting two adjectives, in Example 5 *per (پر)* is acting as proposition and in Example 6 *aur (اور)* is connecting two sentences.

Thus, for segmentation word selection, H1 hypothesis is insufficient condition; role of Haruf-Ataf is also required to be checked. We introduced the concept of stop-words to handle these cases; these words came at the end of the sentence and made it a complete thought. The Table 2 provides the selected list of stop-words, many of these stop-words are auxiliary verbs.

Table 2. List of stop words.

Root Stop Word	Example forms
Aya (آیا)	Ata (اتا), ati(اتی), ateen (آتیں), ayeen (آییں)
Chukay (چکے)	Chaya (چایا), chukey (چکے),
Daya (دیا)	Diya (دیا),dete (دتے)
Gaya (گیا)	Gaya (گیا),gae (گے)
Ha (ہے)	Ha (ہے),haan (ہیں),hoti (ہوتی),hogay (ہوگے) (),howay (ہووے),
Saktay (سکتے)	Saktay (سکتے),sakti (سکتی)
Ja (جا)	Ja (جا),jayan (جایں),jati (جاتی),
Kr (کر)	Ka (کا),ker (کر),keran (کریں),kertay (کرتے), koi (کوئی)
Lga (لگا)	lagty (لگتے), lagi (لگی)
Mila (ملا)	Mila (ملا),mliay (ملے),mili (ملی)
Rahay (رہے)	rahi (رہی),rahtay (رہے)
Tha (تھا)	Thay (تھے),thi (تھی)
Wala (والا)	Wali(والی), walay(والے)

The hypothesis (H2) partially offers a solution to the problem; however, for this study only aur (اور) and per (پر) is considered in this category.

H2: Let $W_i \in S$ *,* W_{i-1} *is previous word of* W_i *in sentiment* S *and* W_{i-1}*='aur' or* W_{i-1}*='per' and then* W_i *is segmentation word (*SW_i*) if POS(*W_{i-1}*) =stop-word.*

Case 2: There are cases in which a sentiment contains more than one Haruf-Ataf; consider following examples.

```
Algorithm 1: SentimentSegmentation(S)
tokens=tokenize(S)
while (tk=tokens.nextToken()!=null)
   {
      if (pos(tk) is HaroofAtaf)
      {
          if (pos(tk) is HaroofSharat)
          {
             hJaza=findHarufJaza(S)
             if (hJaza !=null)
               SW = hJaza
             else
               SW=token
            exit
         }
      if (tk='aur' or tk ='per')
       {
            pw=getPreviousWord(tk)
            if (pw==StopWord)
            {
             sw =tk
               exit
            }
       }
     else
            continue
     sw = token
    }
 }
#segmentation word position
pt =getPosition(SW)
S1=substring(S,0,pt-1)
S2=substring(S,pt+1,length(S)-1)
return S1,S2
```

- *Example 7*: Agar mehnat karoo ga tu kamyab ho gaya.

(اگر محنت کرو گے تو کامیاب ھو گے)

(If you will work hard, then you will succeed)

- *Example 8*: yea mobile boht acha hota agar ess ki battery timing khrab na hoti

(یہ موبائل بہت اچھا ہوتا اگر اس کی بیٹری ٹائمنگ حراب نا ہوتی)

(This mobile would be excellent if its battery is not damaged).

In Example 7, *agar (اگر)* is Haruf-Shaart and *tu (تو)* is Haruf-Jaza; clearly *agar (اگر)* is not segmentation word. However, in example 8 *agar (اگر)* connects two sub opinions so it is a segmentation word.

Thus, to handle these cases, when sentiment has multiple Haruf-Ataf; we suggested a third hypothesis (H3). However, the hypothesis only handles cases when part-of-speech tag of the first word is Haruf-Shart and second word is Haruf-Jaza.

H3: Let W_a *,* $W_b \in S$*, a <b and POS(*W_a*)=Haruf-Shart and PoS(*W_b*)=Haruf-Jaza then* W_b *is a segmentation word (*SW_i*)*

Algorithm 1 explains the process of segmentation, based on the three hypotheses stated above.

3.2. Orientation Score Calculation

A simple BoW base approaches calculate the orientation score by summing up the orientation of each adjective [14, 15].

Let A is sub set of S.

$$A \subset S$$
$$A = \{A_1, A_2, \ldots A_n \mid pos(A_i) = adjective\} \tag{5}$$

The orientation score calculated with help of following equation.

$$score = \sum_{i=1}^{n} Orientation(A_i) \text{ where } A_i \in A \tag{6}$$

However, a set of words reverse the orientation of an adjective and make above equation ineffective.The next section explains these words.

- *Orientation Shifters*: Orientation shifters are the words that reverse the orientation of another word. In Urdu there are two types of orientation shifters, Forward Negation and Backward Negation.
- *Forward Negations*: Reverse the orientation of the adjective that comes after it like words maat (مت) and na (نا) are forward negations.
- *Backward Negations*: Reverse the orientation of the adjective that comes before it, like nahi (نہیں) is example of backward negation.

The decision that the word belongs to forward negation or backward negation depends upon its usage within the text. However, for this research the negation type is fixed irrespective of its use in a sentence. We shortlisted nahi (نہیں) as backward negation; maat (مت) and na (نا) as forward negation. In the light of the above discussion, a stack-based algorithm (Algorithm

2) calculates the orientation score of a sentiment and also takes care of forward and backward negations.

- *Example 9*: Wo acha$^{(+)}$ larka nahi ha
(وہ ایک اچھا لڑکا نہیں ہے)He is not good boy

In Example 9, acha $^{(+)}$ (good) is a positive word but nahi (نہیں) reverse its orientation.

Algorithm 2: OrientationScore(S)

```
tokens[] = tokenize(S)
isApplyForwardNegation = false
while (token = tokens.nextToken)
{
    #checking from dictionary
    tokenOrientation = Orientation(tk)
    partOfSpeechTag = POS(token)
    if partOfSpeechTag == Adjective then
    {
        if isApplyForwardNegation=true
            tokenOrientation
=reverse(tokenOrientation)
        stack.push(tokenOrientation)
    }
    else if partOfSpeechTag==ForwardNegation
        isApplyForwardNegation=true
    else if partOfSpeechTag==BackwardNegation
    {
        previousOrientation = stack.pos()
        newOrientation = reverse
(previousOrientation)
        stack.push(newOrientation)
    }
}
score=sumAllOrientation(stack)
return score
```

3.3. Sentiment Polarity Classification

We observed opinions from different social media sites and reached the conclusion that in Urdu language, people tend to give a negative opinion at the end of the sentiment; in this case polarity of the second opinion dominates the polarity of the overall sentiment. The hypothesis *H4* summarize the concept.

H4: Let sentiment S holds two sub opinions S_1 and S_2. If the polarity of S_2 is a negative, then the sentiment polarity is also negative.

The Algorithm 3, SentimentPolarity, calculates the polarity of sentiment based on the hypothesis.

Algorithm 3: SentimentPolarity(S)

```
S1,S2 = sentimentSegmentation(S)
if S1 !=null and S2 !=null then
{
    score1 = OrientationScore (S1)
    score2 = OrientationScore (S2)
    if score2 < 0 then
    {
        return negative
    }
}
score = score1 + score2
return Polarity(score)
```

4. Evaluation

This section is divided into three subsections: First section contains information about experimental setup, second section provides construction of corpus and third section gives details of results.

4.1. Experimental Setup

The two classifiers BoW and SEGMODEL were implemented using C#. BoW is a legacy type of classifier that calculates polarity of sentiment by summing up the orientation of each adjective within the sentiment. The current version of BoW Algorithm 2 also handles both forward negation and backward negation. Algorithm 3 provides detail of the proposed classifier, SEGMODEL. Both Algorithms 2 and 3 required two files, dataset file and the dictionary file, to start processing.

Each line of the dataset file consists of sentiment text and user-defined polarity separated by a # symbol. Dictionary file contains all unique words exists in the data file. Each line of the dictionary file defines the word, POS tag, and polarity, all separated by the # symbol. The startup program reads the sentiment from the data file, loads the word information from a dictionary file and finally passes the sentiment to the classifier. The classifier returns the estimated polarity of the sentiment. This process repeats for each sentiment of data file and calculates the overall performance of the classifier.

We conducted two independent experiments to compare SEGMODEL with BoW. The first experiment was conducted using BoW classifier and second experiment was performed using SEGMODEL classifier. These experiments used the datasets D1 and D2; the next section contains the details of both datasets. Four metrics Precision (P), Recall (R), Accuracy (A) and F-measure (F) were used to evaluate and compare the performance of both classifiers.

4.2. Datasets

The literature lacks publicly available datasets of roman Urdu sentiment; therefore, we prepared two separate datasets for evaluation of both classifiers. The First Dataset (D1) consisted of 443 product reviews of cars and cosmetic products and the Second Dataset (D2) comprised of 401 product reviews of electronic devices (Table 3).

The D1 and D2 included reviews from different social media forums; forums were of type mobile phones, cars and beauty products. After sentiment collection, three reviewers independently marked polarity, positive or negative, to each sentiment. These reviewers were the student of computer science and active user of social media. Maximum voting algorithm, out of three votes, selected the final polarity of sentiment.

Table 3. Corpus detail.

Dataset	Total Reviews	Average Length	Positive Opinions	Negative Opinions
D1	443	93	194	249
D2	401	81	197	204

4.3. Results

Both classifier BoW (Table 4) and SEGMODEL (Table 5) gave output in terms of the confusion matrix; these results involved in the calculation of precision, accuracy, and recall.

Table 4. Confusion matrix of BoW.

Data Set	Class	TP	TN	FP	FN
D1	Positive	114	191	58	80
	Negative	76	212	20	135
D2	Positive	122	175	29	75
	Negative	54	188	16	143

Table 5. Confusion matrix of SEGMODEL.

Data Set	Class	TP	TN	FP	FN
D1	Positive	133	207	42	61
	Negative	124	204	28	87
D2	Positive	145	180	24	52
	Negative	102	187	17	95

At first step, all three metrics were measured for positive and negative classes of each dataset D1 and D2, using following formulas.

Let $c \in \{positive, negative\}$ and $i \in \{DataSet1, DataSet2\}$

$$\begin{aligned}
&\text{Precision}_c^i = \text{TP}_c^i \ / \ \text{TP}_c^i + \text{FP}_c^i \\
&\text{Recall}_c^i = \text{TP}_c^i \ / \ \text{TP}_c^i + \text{FN}_c^i \\
&\text{Accuracy}_c^i = \text{TP}_c^i + \text{TN}_c^i \ / \ \left(\text{TP}_c^i + \text{FP}_c^i + \text{FN}_c^i + \text{TN}_c^i\right)
\end{aligned} \tag{7}$$

Next, average performance of classifiers was separately measured for both D1 and D2.

$$\begin{aligned}
&\text{Precision}^i = (\text{Precision}^i_{positive} + \text{Precision}^i_{negative}) / 2 \\
&\text{Recall}^i = (\text{Recall}^i_{positive} + \text{Recall}^i_{negative}) / 2 \\
&Accuracy^i = (Accuracy^i_{positive} + Accuracy^i_{negative}) / 2
\end{aligned} \tag{8}$$

Finally, overall performance was calculated by taking the average of both D1 and D2 using following formulas

$$\begin{aligned}
&\text{Precision} = (\text{Precision}^i + \text{Precision}^i) / 2 \\
&\text{Recall} = (\text{Recall}^i + \text{Recal}^i) / 2 \\
&Accuracy = (Accuracy^i + Accuracy^i) / 2
\end{aligned} \tag{9}$$

Results of all these calculations are shown in Tables 6 and 7 for BoW and SEGMODEL respectively.

Table 6. BoW performance metrics.

Data Set	Class	Precision	Recall	Accuracy
D1	Positive	66.28	58.76	68.85
	Negative	79.17	36.02	65.01
Average of D1		72.72	47.39	66.93
D2	Positive	80.79	61.93	74.06
	Negative	77.14	27.41	60.35
Average of D2		78.97	44.67	67.21
Average		75.85	46.03	67.07

Table 7. SEGMODEL performance metrics.

Data Set	Class	Precision	Recall	Accuracy
D1	Positive	76	68.56	76.75
	negative	81.58	58.77	74.04
Average of D1		78.79	63.66	75.4
D2	Positive	85.8	73.6	81.05
	Negative	85.71	51.78	72.07
Average of D2		85.76	62.69	76.56
Average		82.27	63.18	75.98

Table 8 summarized the overall improvement in the performance of SEGMODEL.

Table 8. Comparison between BoW and SEGMOEL.

	Precision	Recall	Accuracy	F-measure
DRT	82.27	63.18	75.98	71.47
BoW	75.85	46.03	67.07	57.29
Increase	6.42	17.15	8.91	14.18

4.4. Significant Test

We performed a statistical test to check whether second opinion significantly leads the polarity of sentiment.

After segmentation, we filter the sentiments, which have two sub-opinions. Then we executed SEGMODEL classifier and this time we logged the processing detail for first opinion polarity and second opinion polarity. From the raw data, we developed two subsets:

S_0: set of all sentiments when polarity of first opinion is same as the actual sentiment polarity.

S_1: set of all sentiments when polarity of second opinion is same as the actual sentiment polarity.

From each subset, we removed the sentiments those are common in both S_0 and S_1. Let p_0 and p_1 are the number of elements in set S_0 and S_1 respectively. We defined a null hypothesis *H0*: $p_1 > p_0$ and alternative hypothesis H_1: $p_0 > p_1$. We used following statistic to calculate Z and P-value.

$$\begin{aligned}
&p = (p_1 * n_1 + p_2 * n_2) / (n_1 + n_2) \\
&SE = \sqrt{p * (1-p)\left(\frac{1}{n_1} + \frac{1}{n_2}\right)} \\
&Z = p_1 - p_2 / SE
\end{aligned} \tag{10}$$

The results and calculation are logged in Table 9. The very low value of level rejects the null hypothesis that is *p0>=p1*

Table 9. Significant test.

Variable Detail	Value
P0 = number of sentiments, in which first opinion determines the polarity of sentiment	80
P1 = number of sentiments when second opinion determine the polarity of sentiment.	392
Significance Level	0.01
z-value	17.1
P	0

5. Discussion and Conclusions

The purpose of the study was to extend the capability of BoW based approaches; to classify complex and ambiguous opinions. The four hypotheses were proposed to segment and classify the sentiment; two experiments were performed to test these hypotheses. First experiment evaluated the performance of BoW model and the second experiment evaluated the performance of proposed classifier, SEGMODEL.

SEGMODEL (Table 1) classified total 226 (D1=124, D2=102) negative opinions correctly; on other hand, BoW (Table 2) model classified total 130 (D1=76, D2=54) negative opinions correctly. Thus, it proves the hypothesis; if second sub-opinion of sentiment is negative then overall polarity of sentiment is also negative.

SEGMODEL also improved the classification of positive opinions: BoW classified total 168 (D1=114, D2=54) and SEGMODEL classified 235 (D1=133, D2=102) positive opinions correctly. These results proved the concept that division of sentiment into sub-opinions increase the performance of the classifier.

The significant test was performed to find whether second opinion is lead opinion within a sentiment. In 80 sentiments, the first sub-opinion determines the polarity of sentiment and in 392 second sub-opinion was the lead opinion. The Z-test (Table 7) with significance level 0.01, z-value =17.1 and p=0 leads the conclusion that in significant number of opinions people tend to conclude the sentiment at end of sentence.

Overall improvement in precision, accuracy and recall (Table 8) showed that the segmentation of sentiment was correctly done, thus it proves the hypothesis that Haruf-Ataf segments the sentiment into two sub-opinions. Above discussion concluded that the sentence-level information indeed improves the classification of complex opinions where BoW failed to classify these sentiments.

Sentiment analysis of Urdu language leads to social media mining, brand monitoring, observing the political situation of the country and prediction of possible future turmoil. However, researcher should explore methods other than BoW based approaches; the results encouraged that the sub sentence-level information improved the classification of sentiment. Future research may include identification of complex relationships between sub-opinions and rigorous theory to handle sentiment containing more than two sub opinion.

References

[1] Afraz Z., Muhammad A., and Martinez-Enriquez A., "Sentiment-Annotated Lexicon Construction for an Urdu Text Based Sentiment Analyzer," *Pakistan Journal of Science*, vol. 63, no. 4, pp. 222-225,2011.

[2] Al-Kabi M., Al-Ayyoub M., Alsmadi I., and Wahsheh H., "A Prototype for a Standard Arabic Sentiment Analysis Corpus," *The International Arab Journal of Information Technology*, vol. 13, no. 1A, pp. 163-170, 2016.

[3] Annett M. and Kondrak G., "A comparison of Sentiment Analysis Techniques: Polarizing Movie Blogs," *in proccedings of the Canadian Society for Computational Studies of Intelligence*, Windsor, pp. 25-35, 2008.

[4] Asher N., Benamara F., and Mathieu Y., "Appraisal of Opinion Expressions in Discourse," *Lingvisticœ Investigationes*, vol. 32, no. 2, pp. 279-292, 2009.

[5] Asher N., Benamara F., and Mathieu Y., "Distilling Opinion in Discourse: A Preliminary Study," *in Proceedings of International Conference on Computational Linguistics*, Manchester, pp. 7-10, 2008.

[6] Bilal M., Israr H., and Shahid M., "Sentiment classification of Roman-Urdu Opinions using Naιve Bayesian, Decision Tree and KNN Classification Techniques," *Journal of King Saud University-Computer and Information Sciences*, vol. 28, no. 3, pp. 330-344, 2015.

[7] Bilgic M., Namata G., and Getoorl L.,"Combining Collective Classification and link Prediction," *in Proccedings of International Conference on Data Mining Workshops*, Omaha, pp. 381-386, 2007.

[8] Irvine A., Weese J., and Burch C., "Processing Informal Romanized Pakistani Text Messages," *in Proceedings of the Second Workshop on Language in Social Media*, Montreal, pp. 75-78, 2012.

[9] Liu B., *Sentiment Analysis and Opinion Mining*, Morgan and Claypool Publishers, 2012.

[10] Mukherjee S. and Bhattacharyya P., "Sentiment Analysis in Twitter with Lightweight Discourse Analysis," *in Proceedings of 24th International Conference on Computational Linguistics*, Bombay, pp. 1847-1864, 2012.

[11] Mukund S. and Srihari R., "Analyzing Urdu Social Media for Sentiments Using Transfer Learning with Controlled Translations," *in Proceedings of the Second Workshop on Language in Social Media,* Montreal, pp. 1-8, 2012.

[12] Somasundaran S., Namata G., Getoorl L., and Wiebe J., "Opinion Graphs for Polarity and Discourse Classification," *in Proceedings of the Workshop on Graph-based Methods for Natural Language Processing,* Suntec, pp. 66-74, 2009.

[13] Somasundaran S., Ruppenhofer J., and Wiebe J., "Discourse level Opinion Relations: An Annotation Study," *in Proceedings of the 9th Sigdial Workshop on Discourse and Dialogue*, Stroudsburg, pp. 129-137, 2008.

[14] Syed A., Aslam M., and Enriquez A., "Lexicon based Sentiment Analysis of Urdu Text Using SentiUnits," *in Proceedings of Advances in Artificial Intelligence*, Pachuca, pp. 32-43, 2010.

[15] Syed A., Aslam M., and Martinez-Enriquez A., "Sentiment Analysis of Urdu Language: Handling Phrase-level Negation," *Advances in Artificial Intelligence*, Puebla, pp. 382-393, 2011.

[16] Taboada M., Voll K., and Brooke J., "Extracting Sentiment as a Function of Discourse Structure and Topicality," Technical Report School of Computing Science Simon Fraser University, 2008.

[17] Thelwall M., Backley K., Paltoglou G., Cai D., and Kappas A., "Sentiment Strength Detection in Short Informal Text," *Journal of the American Society for Information Science and Technology*, vol. 61, no. 12, pp. 2544-2558, 2010.

[18] Turney P. and Littman M., "Measuring praise and criticism: Inference of Semantic Orientation from Association," *ACM Transactions on Information Systems*, vol. 21, no. 4, pp. 315-346, 2003.

[19] Wiebe J., Wilson T., Bruce R., bell M., and Martin M., "Learning Subjective Language," *Computational Linguistics*, vol. 30, no. 3, pp. 277-308, 2004.

[20] Zhou L., Li B., Gao W., Wei Z., and Wong K., "Unsupervised Discovery of Discourse Relations for Eliminating Intra-Sentence Polarity Ambiguities," *in Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Edinburgh, pp.162-171, 2011.

[21] Zirn C., Niepert M., Stuckenschmidt H., and Trubem S., "Fine-Grained Sentiment Analysis with Structural Features," *in Proceedings of the 5th International Joint Conference on Natural Language Processing*, Chiang Mai, pp. 336-344, 2011.

Muhammad Hassan is assisstant professor at Computer Sciecne and Engineering Department at the University of Engineering and Tecnhology. He is Gold Medalist of Punjab University. He has completed his MS from UET lahore and Currently doing Ph.d from the same university. His research interest includes Natural Language Processing, Semantic Web, Software Arechitecture and Open source Software Development.

Muhammad Shoaib is a professor at Computer Science and Engineering Department at the University of Engineering and Technology Lahore, Pakistan. He received his MSc in computer science from Islamia University,Pakistan. He has completed his PhD from the University of Engineering and Technology, Pakistan in 2006. His Post Doc. is from Florida Atlantic University, USA, in 2009. His current research interests include information retrieval systems, information systems, software engineering and semantic web.