Scientific Journal of Impact Factor (SJIF): 5.71
e-ISSN (O): 2348-4470
p-ISSN (P): 2348-6406
International Journal of Advance Engineering and Research Development
Volume 6, Issue 07, July -2019

Aspect Based Hindi Sentiment Analysis through Neural Networks

Geeta Kolkonda[1], Dhanashri Kulkarni[2]

[1] *Computer Engineering Dept., D. Y. Patil College Of Engineering, Ambi, Talegaon Dabhade, Pune*
2 Computer Engineering Dept., D. Y. Patil College Of Engineering, Ambi, Talegaon Dabhade, Pune

Abstract — *Checking websites for online reviews and opinions online before buying a product or visiting a place has become quite common now-a-days. Millions of people express their opinions on social media networking sites, blogs and review sites on various topics including movies, restaurants, products etc. These websites provide myriad amount of information that are not only useful to the creators of these entities, but also to their customers and rivals. Analyzing and categorizing the sentiments expressed around the aspects present in these opinions, instead of overall polarity, have become a critical factor to make decisive recommendations. Consider the sentence,*

"फोन में बड़ा बैटरी लगा है जो एक बैटरी चार्ज पे तीन दिन चल सकता है, मगर प्रोसेसर बहुत धीमी है|"

In this sentence, the customer has expressed his opinion about a mobile phone for 'battery' aspect with a positive sentiment but for 'processor speed' aspect the sentiment expressed is negative. A user may set his priority on certain aspects or features and decide to buy the product. Hence, it becomes important to provide a granular level of recommendation. Our project aims at improving ABSA model on Hindi reviews with latest technologies of Deep Neural Networks.

Keywords*- Hindi Sentiment Analysis, Natural Language Processing, Sentiment Analysis, Aspect Based Sentiment Analysis.*

I. INTRODUCTION

Hindi is the widely spoken language in India, when compared in terms of speaker population, Hindi ranks 5th in the world. The main issue with sentiment analysis involving Hindi is due to the non-availability of benchmark datasets and the scarcity of various other resources and tools required.

Initially, sentiment analysis was performed mainly to detect the overall polarity (e.g., positive or negative) of a given text or text span [1][4] Sentiment analysis in Indian languages are still largely unexplored due to the non-availability of various resources and tools such as annotated corpora, lexicons, Part-of- Speech (PoS) tagger etc. Existing works [3]-[9] involving Indian languages mainly discuss the problems of sentiment analysis at the higher level –sentence or document level, with the aim of classifying sentiments. However, the need for a more fine-grained approach, such as aspect-based (or feature based) sentiment analysis (ABSA), has become the need of the hour [2].

In 2014, a SemEval shared task [12] was contributed to address this problem in two domains namely, restaurant & laptop. It included four subtasks:

- Aspect Term Extraction (ATE)
- Aspect Term Sentiment (ATS)
- Aspect Category Detection (ACD)
- Aspect Category Sentiment (ACS)

In this project we are concentrating on the first 2 subtasks namely Aspect Term Extraction (ATE) & Aspect Term Sentiment (ATS).

II. ASPECT BASED SENTIMENT ANALYSIS

There are 4 subtasks in an Aspect Based Sentiment Analysis as defined in SemEval shared tasks [11]. Each of these is defined below:

- Aspect Term Extraction (ATE): Aspect Term Extraction aims to extract explicit aspect terms that are present in user reviews. It is treated as a sequence labeling problem, where labels are assigned to a sequence of terms. Here we identify all aspect terms irrespective of whether it expresses sentiment or not.
- Aspect Term Sentiment (ATS): This is treated as a classification problem, i.e. classifying the sentiment around the aspect term into positive, negative, neutral and conflict. At this stage we assume that the aspect terms are given and we are required to determine the polarity of each aspect term (positive, negative, conflict, or neutral).
- Aspect Category Detection (ACD): It is treated as a multi-label classification problem, where each review belongs to zero or more predefined categories.
- Aspect Category Sentiment (ACS): Here the aspect category is given and we have to identify the sentiment expressed towards that aspect category in the review has to be classified into positive, negative, neutral and conflict.

Our goal here is to identify the aspect terms in a given review, extract sentiments expressed by the aspect in the review and classify the sentiment into positive, negative, neutral or conflict.

III. RELATED WORK

Not much work has been done in this field for Hindi language. Earliest work done in this field was in 2009 by Rao et al. [14] where they worked on three languages –English, French and Hindi. The concept behind their work was having prior knowledge of polarity of the term based on which polarity of the sentence can be deduced. They obtained polarity of terms from WordNet & other thesauri for cases where WordNet did not exist. They also covered synonyms and hypernym relationships between words to obtain better results. They obtained an F1-score of 91%. Their method failed when monolingual raw text was provided. Hypernym, antonym, etc. relationships were suggested for future work.

Das et al. [8] in 2010 created SentiWordNet, a sentiment lexicon for Bengali, Hindi and Telugu languages using bi-lingual dictionary and existing WordNet. They used resources like Shabdkosh and Shabdanjali along with English language's SentiWordNet. To test the system, manual evaluation was done by picking up 100 random words from Hindi SentiWordNet. Accuracies of 88% and 91% were obtained for positive and negative polar words respectively. Complete evaluation of Hindi SentiWordNet was never made though.

Alternatively, a fallback strategy was suggested by Joshi et al. [3] in 2010 for Hindi sentiment analysis. As per them sentiment classification can be performed using –In-language technique, where a model is created on a training data in the same language, Machine translation technique, where given sentences are translated to English language and an English language trained classifier is used, and lastly, HSWN (HindiSentiWordNet) technique, where polarity of the sentence is calculated from the polarity of individual terms in it. As per the results obtained, In-language technique provided best results with the accuracy of 78.14%.

Hindi Subjective Lexicon (HSL) was created by Bakliwal et al. [15] in 2012 that consisted of adjectives and adverbs with their polarity scores. They used WordNet and pre-annotated seed list with 45 adjectives and 75 adverbs. They considered synonym and antonym relationship among words too. To evaluated HSL, they performed classification, tested against existing resources and manual evaluation by experts. Piyush Arora [16] in 2013 too created a subjective lexicon and proposed different techniques of performing sentiment classification.

Later Mittal et al. [6] in 2013 worked on improving HSWN by adding more opinion words to it. They also worked on handling negation and discourse relation among words. To evaluate the system, they performed sentiment classification that gave an accuracy of 80.21%.

Opinion Mining system for Hindi language was put forth by Sharma et al. [7] in 2014 which took care of negation handling as well. They obtained an accuracy of 65%.

Hindi Senti Lexicon containing nouns, adjectives, verbs and adverbs was created by Sharma and Bhattacharya in 2014 [17]. A Multi-module sentiment analysis system was created to verify the senti lexicon. But this system heavily depended on an effective POS tagger; a window size of 3 words was considered which might have missed out on discourse relation among terms.

In 2015 a shared task was created for sentiment analysis for Indian languages Hindi, Bengali and Tamil –SAIL Shared Task 2015. They provided a set of annotated tweets for classifying them into positive, negative or neutral.

Many researchers participated in this like Prasad et al. [18] who classified using decision tree algorithm and obtained an accuracy of 40.47%.

Kumar et al. [20] also participated in SAIL Shared Task 2015 by performing regularized least square based sentiment analysis. They obtained an accuracy of 47.96%.

Ayush Kumar et al. [21] also participated in SAIL Shared Task 2015, they used existing Indian sentiment lexicon along with distributed thesaurus and sentence level co-occurrences. They obtained an accuracy of 49.68%.

Vandana Jha et al. in 2015 [19] created Hindi Opinion Mining System (HOMS) for movie review using machine learning algorithm and POS tagging. They obtained an accuracy of 80%.

Deepali Mishra et al. [22] in 2016 built a polarity lexicon algorithm that was domain specific and context sensitive. They obtained an accuracy of 77% for movie domain and 88% for hotel domain.

Pontiki et al. proposed 4 sub-tasks for Aspect based sentiment analysis –SemEval 2014 Task 4 [11]. Akhtar et al. in 2016 set up a benchmark for Hindi sentiment analysis [9-10] based on the 4 sub-tasks proposed in SemEval 2014[11]. They created Hindi review dataset for 12 domains, identified aspect terms and sentiment around them; finally identify the category these aspect terms belong to and sentiment around each category. They obtained an F-measure of 41.07% for aspect term identification, 54.05% accuracy for aspect term sentiment.

Akhtar et al. in 2016 proposed a hybrid deep learning architecture [23] for Hindi sentiment analysis. Where data is passed through a framework based on Genetic Algorithm is used along with CNN, it is then fed to a non-linear SVM classifier to obtain the results. This method resulted in an accuracy of 65.96%.

Komal Garg and Preetpal Kaur Buttar also worked on aspect based sentiment analysis in 2017 [24]. They proposed an algorithm that could extract aspect terms and their sentiments using SentiWordNet database. They also handled negation and conjunctions in the sentences which gave an accuracy of 83.3%.

Most recent work in this area was by Akhtar et al. in 2018 where they created an efficient word embedding for Hindi language [12]. This word embedding was trained on a parallel corpus of English and Hindi sentences. To evaluate the system, multilingual and cross-lingual set ups were made and aspect level sentiment classification was performed. This work recorded highest accuracy of 76.29%

IV. NEURAL NETWORK ARCHITECTURE FOR ABSA

To identify the aspect terms and the sentiment around them, we train a set of neural networks. In this section we describe the architecture of the neural network involved in this.

4.1. Aspect term identification:

A neural network for sequence labeling task was created [25], first a convolutional neural networks (CNNs) to encode character-level information of a word into its character-level representation is created. Then character- and word-level representations are combined and feed into bi-directional LSTM (BLSTM) to model context information of each word. On top of BLSTM, a sequential CRF is used to jointly decode labels for the whole sentence.

A model is constructed using Convolutional Neural Network, Bi-directional Long Short Term Memory and Conditional Random Fields. Each of these layers performs the following task: Convolutional Neural Network (CNN) – Each word is fed to this layer that computes its character level representation using character embedding. Bi-directional Long Short Term Memory (BiLSTM) – The character level vector from the CNN layer is concatenated with the word embedding vector and fed to BiLSTM which calculates the relationship among the words in the sentence depending on the past and future information of the sentence. Conditional Random Fields (CRF) – Output vectors from BLSTM is fed to this layer to jointly decode the best label sequence for the input sentence.

4.1.1. Obtaining Character level representation using CNN:

CNN effectively extracts morphological information (like the prefix or suffix of a word) from characters of words and encode it into neural representations. First a dropout layer is applied before character embedding is input to CNN. CNN is then used to extract character-level representation of a given word.

4.1.2. Obtaining relationship among words using BLSTM:

RNNs are capable to capturing long-distance dependencies, in practice; but they fail due to the gradient vanishing/exploding problems. To handle these gradient vanishing problems, LSTMs were designed. An LSTM unit consists of three multiplicative gates that control the proportions of information to forget and to pass on to the next time step. But LSTM's hidden state takes information only from past, knowing nothing about the future. An elegant solution whose effectiveness has been proven by previous work is bi-directional LSTM (BLSTM). In a BLSTM, past and future information is captured by two separate sets of hidden states, one for forward state and other backward. The final output is obtained by concatenating outputs of these two hidden states.

4.1.3. Decoding best label sequence using CRF:

Conditional Rando Fields (CRF) is used for structured prediction tasks. Correlations between labels in neighborhoods in input sentence are considered which are then decoded to assign best chain of labels. Here a label sequence is modeled jointly using a conditional random field (CRF), instead of decoding each label independently.

4.2. Aspect Term Sentiment Analysis:

To identify the sentiment around an aspect term, deep memory network is used. Weston et al. in 2014 introduces a machine learning framework called Memory network. Its central idea is inference with a long-term memory component, which could be read, written to, and jointly learned with the goal of using it for prediction. A memory network is made up of a memory m –an array of objects like vectors; and four components I –converts input to internal features, G –updates old memories with new input, O –combines new input and current memory state to generate output representation; and finally R –generates a response based on output representation.

The task here is to find the sentiment around the given aspect word in the given sentence. This is done by mapping each word into its embedding vector; these words are separated into two parts –aspect vector and context vector. Context word vectors are then stacked and regarded as the external memory. There are multiple computational layers (hops), each of which contains an attention layer and a linear layer. In the first computational layer (hop 1), aspect vector are treated as the input to adaptively select important evidences from memory through attention layer. The output of attention layer and the linear transformation of aspect vector are summed and the result is considered as the input of next layer (hop 2). In a similar way, multiple hops are stacked together and run multiple times, so that more abstractive evidences could be selected from the external memory. The output vector in last hop is considered as the representation of sentence with regard to the aspect, and is further used as the feature for aspect level sentiment classification.

The output vector from the last hop is fed to a softmax layer for aspect level sentiment classification. The model is trained in a supervised manner by minimizing the cross entropy error of sentiment classification.

V. PROPOSED SYSTEM FOR ABSA IN HINDI LANGUAGE

We are using the data created in [10]. The data was crawled from various online Hindi sources like newspapers, blogs, e-commerce websites etc.to obtain review sentences covering 12 domains.

The 12 domains covered are:

- Laptops,
- Mobiles,
- Tablets,
- Cameras
- Home appliances,
- Mobile apps,
- Smart watches,
- Headphones
- Speakers,
- Television,
- Travel and
- Movies

The data has to pass through various stages to perform aspect based sentiment analysis. Following subsections explain those stages.

5.1. Pre-processing Stage

Crawled, raw data (Fig1) was pre-processed to convert it to the desired format through steps given below. The annotated data is in XML format as shown in Fig2.
Steps taken to pre-process the crawled data are:

- Removing irrelevant reviews from the raw data. These may be reviews that were not part of the 12 domains that were considered or contained some junk information.
- All kind of emoticons were dropped like smiles, thumbs-up, etc.
- Spelling mistakes were taken care of as well at this stage.
- Missing end of sentence mark (|) was added as well.

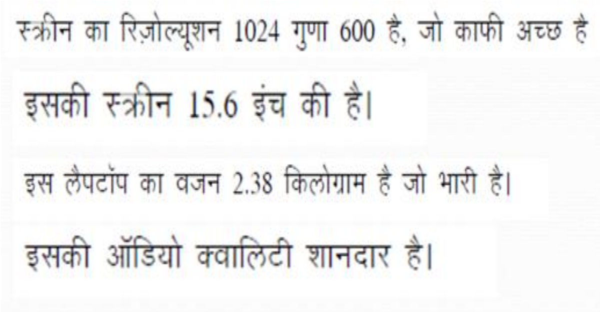
स्क्रीन का रिज़ोल्यूशन 1024 गुणा 600 है, जो काफी अच्छ है
इसकी स्क्रीन 15.6 इंच की है।
इस लैपटॉप का वजन 2.38 किलोग्राम है जो भारी है।
इसकी ऑडियो क्वालिटी शानदार है।

Fig 1. Raw Data Sample

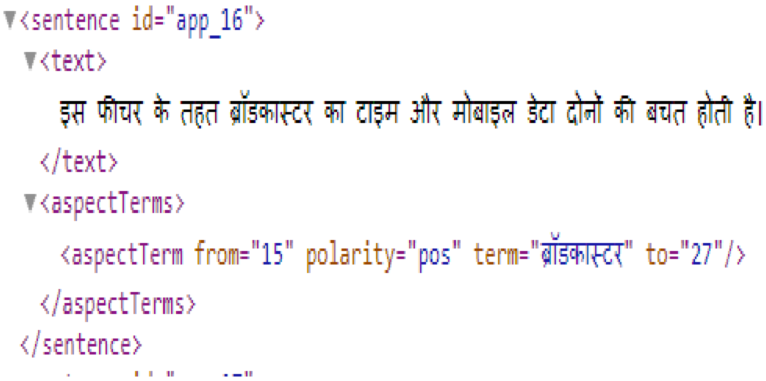
```
▼<sentence id="app_16">
  ▼<text>
     इस फीचर के तहत ब्रॉडकास्टर का टाइम और मोबाइल डेटा दोनों की बचत होती है।
   </text>
  ▼<aspectTerms>
     <aspectTerm from="15" polarity="pos" term="ब्रॉडकास्टर" to="27"/>
   </aspectTerms>
 </sentence>
```

Fig 2. Annotated Data Sample

5.2. Aspect Term Extraction Stage

This is a sequence labeling problem, where for given sequence of tokens, one has to mark the boundary of an aspect term properly. We require all the aspect terms to be identified, including aspect terms for which no sentiment is expressed (neutral polarity). To extract aspect terms, the data needs to be in BIO format. The training data was converted into BIO format which was then tagged using BIOES tagging scheme.
Hindi word embedding provided by fastText group has been used here. These word vectors are trained on Wikipedia using fastText. These are 300 dimensional vectors that were obtained as defined by Bojanowski et al. in 2016 using the skip-gram model.
The main architecture of the neural network is as shown in fig 3. First a CNN is used to compute the character representation for each word. This is then concatenated with the word embedding and fed to the BLSTM network. Once the relationship among the terms is learnt through BLST, this information is fed to Conditional Random Fields (CRF) layer that label the aspect terms.

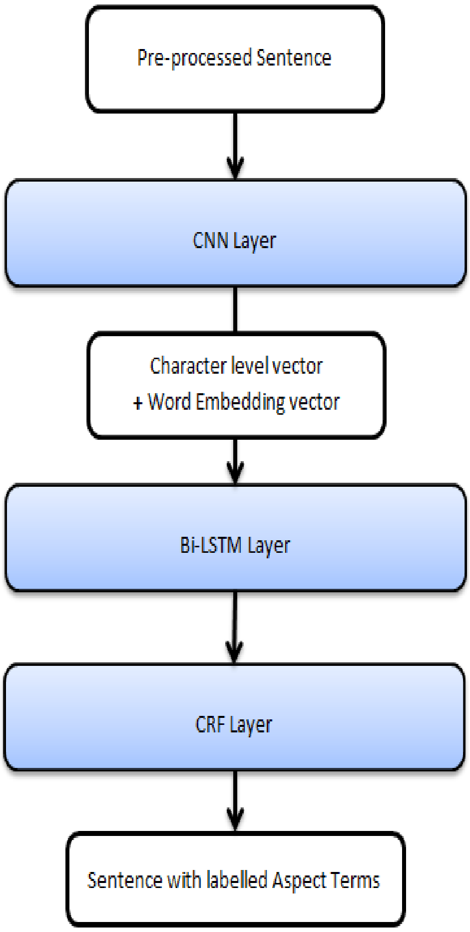

Fig 3. Block Diagram
Neural Network Design for Aspect Term Extraction

5.3. Aspect Term Sentiment Analysis

The task here is to find the sentiment around the given aspect word in the given sentence. This is done by mapping each word into its embedding vector; these words are separated into two parts –aspect vector and context vector. Context word vectors are then stacked and regarded as the external memory. There are multiple computational layers (hops), each of which contains an attention layer and a linear layer. In the first computational layer (hop 1), aspect vector are treated as the input to adaptively select important evidences from memory through attention layer. The output of attention layer and the linear transformation of aspect vector are summed and the result is considered as the input of next layer (hop 2). In a similar way, multiple hops are stacked together and run multiple times, so that more abstractive evidences could be selected from the external memory. The output vector in last hop is considered as the representation of sentence with regard to the aspect, and is further used as the feature for aspect level sentiment classification.

The output vector from the last hop is fed to a softmax layer for aspect level sentiment classification. The model is trained in a supervised manner by minimizing the cross entropy error of sentiment classification.

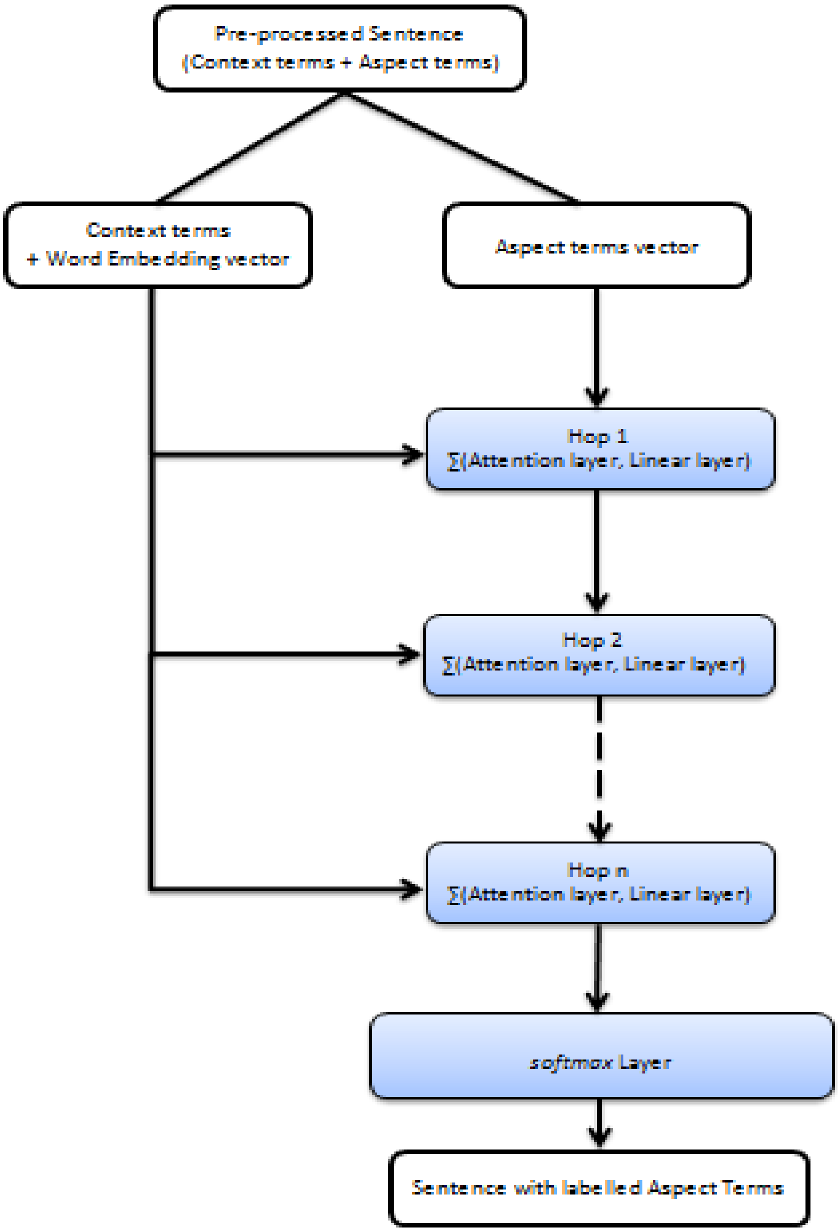

Fig 4. Block Diagram
Neural Network Design for Sentiment Classification

5.4. Mathematical Model

Consider the review text

$V_T = \{v_1, v_2, \ldots, v_t\}$

After pre-processing

$V = \{v_1, v_2, \ldots, v_n\}$

Aspect term:

$A_v = \{a_1, a_2, \ldots, a_m\}$

Where $A_v \subset V$

Aspect term Sentiment:

$S_v = \{s_{va1}, s_{va2}, \ldots, s_{vam}\}$

Where s_{vai} is sentiment towards aspect a_i in the review text and $s_{vai} \subset S$

And S = {'pos', 'neg', 'neu', 'con'}

5.5. Algorithm

For a given review text following steps was defined to extract aspect terms and the sentiment around it.

- Given review text $V_T = \{v_1, v_2, \ldots, v_t\}$ defined as a vector of *t* terms.
- Preprocessing
  - Tokenization
  - Chunking

- Pos tagging

We get $V = \{v_1, v_2, \ldots, v_n\}$, a vector of n terms.

- Aspect identification

$A_v = \{a_1, a_2, \ldots, a_m\}$
With *m* aspect terms
Where $A_v \subset V$

- Sentiment classification

$S_v = \{s_{va1}, s_{va2}, \ldots, s_{vam}\}$
Where s_{vai} is sentiment towards aspect a_i in the review text and $s_{vai} \subset S$
And S = {'pos', 'neg', 'neu', 'con'}

VI. EVALUATION & RESULTS

6.1. Aspect Term Extraction:

Proposed system was compared with existing system by Akhtar et al. [10]. The existing system used only Conditional Random Field (CRF) –CRF++ package to identify aspect terms, whereas in our proposed system we used CNN and BLSTM along with CRF.

6.1.1. Existing system:

Existing system by Akhtar et al.[10] was developed using Conditional Random Fields (CRF++) for aspect term identification. The classifier is trained with the features:

- Word & local context
- Part-of-Speech (PoS)
- Chunk information
- Prefixes and suffixes

The existing system used Conditional Random Field (CRF) –CRF++ package for the aspect term extraction.
To evaluate the performance of the system, researcher used the evaluation scripts made available by the SemEval 2014 shared task organizers. The 3-fold cross-validation of the final evaluation results were as shown in Table 1. Confusion matrices for aspect extraction and sentiment analysis are given in Table 2 & 3 respectively.

Aspect Term Extraction	F-Measure	41.07%
Aspect Term Sentiment	Accuracy	54.05%

Table1. Evaluation results of the existing system

	B	**I**	**O**
B	1210	128	2477
I	142	754	2105
O	1065	887	91511

Table2. Confusion matrix of Existing System's Results - Aspect Term Extraction

	positive	**negative**	**neutral**	**conflict**
positive	1416	30	540	0
negative	376	51	142	0
neutral	917	27	970	0
conflict	30	4	6	0

Table3. Confusion matrix of Existing System's Results - Sentiment Classification

6.1.2. Proposed System:

Given dataset was first divided into three sets –training, dev and test sets. Training set was used to train the model created by our system. The trained model was then validated against dev set and finally was tested using the test set. This system was executed twice, results of which are as shown in Table III. As system is designed on deep neural network, it needs more than one epoch to get optimum results. During the first run 15 epochs were executed and for the second run, it was increased to 50 epochs. But after a few epochs, the system started to over fit the data resulting in drop in accuracy. Snapshot of the output is shown in Fig6 and the results of best epoch are shown in Table4 and graphical representation of the output in Fig5.

	Precision	Recall	F-score
Train	0.77	0.7	0.73
Dev	0.59	0.49	0.54
Test	0.58	0.52	0.55

Table4. Final result –Aspect Term Identification

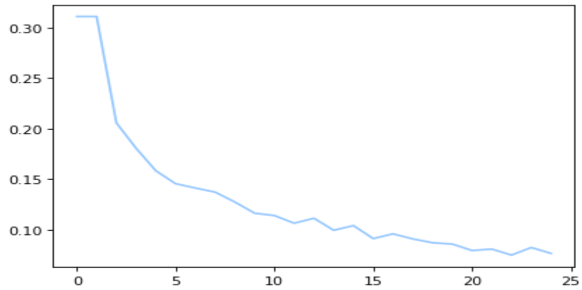

Fig 5: Graphical representation of the Output –Aspect Term Identification

```
Train: Precision: 0.6846032992930087
Train: Recall: 0.5890503548496113
Train: new_F-Score: 0.6332425068119891 best_F-Score: -1.0
Dev: Precision: 0.5777414075286416
Dev: Recall: 0.4644736842105263
Dev: new_F-Score: 0.5149525893508388 best_F-Score: -1.0
Saving Model to ./models/HSA_260219
Test: Precision: 0.560105680317041
Test: Recall: 0.4732142857142857
Test: new_F-Score: 0.5130066545674531 best_F-Score: -1.0
14000 : 0.1371920165081936
16000 : 0.1273468564080748
18000 : 0.11631312409697891
20000 : 0.11401667205677239
22000 : 0.10640154655156096
24000 : 0.11136474268574592
26000 : 0.09944312095474112
Train: Precision: 0.7713968136346795
Train: Recall: 0.7036160865157147
Train: new_F-Score: 0.735949098621421 best_F-Score: 0.6332425068119891
Dev: Precision: 0.5968503937007874
Dev: Recall: 0.4986842105263158
Dev: new_F-Score: 0.5433691756272401 best_F-Score: 0.5149525893508388
Saving Model to ./models/HSA_260219
Test: Precision: 0.5841708542713567
Test: Recall: 0.5189732142857143
Test: new_F-Score: 0.549645390070922 best_F-Score: 0.5130066545674531
```

Fig 6: Sample Output – Aspect Term Identification

6.2. Aspect Term Sentiment Analysis:

Existing system by Shad Akhtar used SVM for sentiment classification whereas our proposed system input data is passed through multiple computational layers and the output of the final layer is fed to softmax to extract the sentiment.

6.2.1. Existing sytem:

Support Vector Method (SVM) for sentiment classification. The classifier is trained with the following features:

- Target aspect term and local context
- Word Bigrams
- Semantic Orientation

The existing system used Conditional Random Field (CRF) –CRF++ package and Support Vector Machine (SVM) – TinySVM package for the aspect term extraction and sentiment classification tasks respectively. Results of the existing system are shown in Table1-3 above.

6.2.2. Proposed system:

The task here is to find the sentiment around the given aspect word in the given sentence. This is done by mapping each word into its embedding vector; these words are separated into two parts –aspect vector and context vector. Context word vectors are then stacked and regarded as the external memory. There are multiple computational layers (hops), each of

which contains an attention layer and a linear layer. In the first computational layer (hop 1), aspect vector are treated as the input to adaptively select important evidences from memory through attention layer. The output of attention layer and the linear transformation of aspect vector are summed and the result is considered as the input of next layer (hop 2). In a similar way, multiple hops are stacked together and run multiple times, so that more abstractive evidences could be selected from the external memory. The output vector in last hop is considered as the representation of sentence with regard to the aspect, and is further used as the feature for aspect level sentiment classification. The output vector from the last hop is fed to a softmax layer for aspect level sentiment classification. The model is trained in a supervised manner by minimizing the cross entropy error of sentiment classification.
Output of a 7hop, 4 epoch run is shown in Table5 and sample output is as shown in Fig7.

	Accuracy
Training set	0.73
Test set	0.74

Table5. Final result –Aspect Term Sentiment Analysis

```
epoch 3...
loss -  0.48569223
loss -  0.68007904
loss -  0.4481433
loss -  0.04522133
loss -  1.2407494
loss -  0.041370206
loss -  1.8887969
loss -  0.1154233
train-loss=0.61;train-acc=0.78;test-acc=0.74;
epoch 4...
loss -  0.07336266
loss -  1.3282754
loss -  0.15538462
loss -  0.03814653
loss -  0.16189197
loss -  1.2266076
loss -  0.03604766
loss -  1.918244
train-loss=0.57;train-acc=0.80;test-acc=0.71;
Best Epoch: 0
Train-loss=0.85;Train-acc=0.73;Test-acc=0.74
```

Fig 6: Sample Output – Aspect Term Sentiment Analysis

VII. CONCLUSION

In this paper we have proposed a system for Aspect Based Sentiment Analysis for Hindi language using latest neural network techniques. Our system consists of 2 stages, first aspect term extraction stage, where our model was trained by passing the data through CNN to obtain character level representation; this along with word embedding was given to BLSTM layer that understood the relationship among the terms, this information when fed to a CRF extracted the aspect terms in the sentence. During second stage, aspect term sentiment analysis was performed by passing the input data through multiple computational layers and the output of the final layer is fed to softmax to extract the sentiment. We obtained an F-score of 0.55 or 55% for Aspect term extraction against 41.07% achieved by existing system and an accuracy of 0.74 or 74% as compared to 54.05% of the existing system.

VIII. ACKNOWLEDGMENTS

My sincere thanks to Md Shad Akhtar, Asif Ekbal and Pushpak Bhattacharyya for providing data set and base of this research.

REFERENCES

[1] Pang, Bo, and Lillian Lee. ”A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts.” Proceedings of the 42nd annual meeting on Association for Computational inguistics. Association for Computational Linguistics, 2004.

[2] Liu, Bing. ”Sentiment analysis and opinion mining.” Synthesis lectures on human language technologies 5.1 (2012): 1-167.

[3] Aditya Joshi ,Balamurali A R, Pushpak Bhattacharyya ,A Fall-back Strategy for Sentiment Analysis in Hindi: a Case Study , Proceedings of ICON 2010: 8th International Conference on Natural Language

[4] Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews Peter D. Turney (National Research Council of Canada) (Submitted on 11 Dec 2002).

[5] Mukherjee, Subhabrata & Malu, Akshat & R, Balamurali & Bhattacharyya, Pushpak. (2013). TwiSent: A multistage system for analyzing sentiment in Twitter. ACM International Conference Proceeding Series. 10.1145/2396761.2398684.

[6] Mittal, Amit Kumar, Yusuf Chisti, and Uttam Chand Banerjee. ”Synthesis of metallic nanoparticles using plant extracts.” Biotechnology advances 31.2 (2013): 346-356.

[7] Sharma, Richa, Shweta Nigam, and Rekha Jain. ”Opinion mining of movie reviews at document level.” arXiv preprint arXiv:1408.3829 (2014).

[8] Das, Amitava, Sivaji Bandyopadhyay, and Bjrn Gambck. “Sentiment analysis: what is the end user’s requirement?.” Proceedings of the 2nd International Conference onWeb Intelligence, Mining and Semantics. ACM, 2012.4

[9] Akhtar, Md Shad, Asif Ekbal, and Pushpak Bhattacharyya. ”Aspect based Sentiment Analysis in Hindi: Resource Creation and Evaluation.” LREC. 2016.

[10] Akhtar, Md Shad, Asif Ekbal, and Pushpak Bhattacharyya. ”Aspect Based Sentiment Analysis: Category Detection and Sentiment Classification for Hindi.” International C onference on Intelligent Text Processing and Computational Linguistics. Springer, Cham, 2016.

[11] Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., and Manandhar, S. (2014). Semeval-2014 task 4: Aspect based sentiment analysis. In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, August.

[12] Shad Akhtar, Md & Sawant, Palaash &f Sen, Sukanta &f Ekbal, Asif &f Bhattacharyya, Pushpak. (2018). Solving Data Sparsity for Aspect Based Sentiment Analysis Using Cross-Linguality and Multi-Linguality. 572-582. 10.18653/v1/N18-1053

[13] G. P. Zhang, ”Neural networks for classification: a survey,” in IEEE Transactions on Systems, Man, and Cyberneti11cs, Part C (Applications and Reviews), vol. 30, no. 4, pp. 451-462, Nov. 2000. doi: 10.1109/5326.897072.

[14] Rao, Delip, and Deepak Ravichandran. "Semi-supervised polarity lexicon induction." Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, 2009.

[15] Bakliwal, Akshat, Piyush Arora, and Vasudeva Varma. "Hindi subjective lexicon: A lexical resource for hindi polarity classification." Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC). 2012

[16] Arora, Piyush. "Sentiment analysis for hindi language." MS by Research in Computer Science (2013).

[17] Sharma, Raksha, and Pushpak Bhattacharyya. "A sentiment analyzer for hindi using hindi senti lexicon." Proceedings of the 11th International Conference on Natural Language Processing. 2014.

[18] Prasad, Sudha Shanker, et al. "Sentiment classification: an approach for Indian language tweets using decision tree." International Conference on Mining Intelligence and Knowledge Exploration. Springer, Cham, 2015.

[19] Jha, Vandana et al. “HOMS: Hindi opinion mining system.” 2015 IEEE 2nd International Conference on Recent Trends in Information Systems (ReTIS) (2015): 366-371.

[20] Kumar, S. Sachin, et al. "AMRITA_CEN-NLP@ SAIL2015: sentiment analysis in Indian Language using regularized least square approach with randomized feature learning." International Conference on Mining Intelligence and Knowledge Exploration. Springer, Cham, 2015.

[21] Kumar, Ayush, et al. "IIT-TUDA: System for sentiment analysis in Indian languages using lexical acquisition." International Conference on Mining Intelligence and Knowledge Exploration. Springer, Cham, 2015.

[22] Mishra, Deepali, Manju Venugopalan, and Deepa Gupta. "Context specific Lexicon for Hindi reviews." Procedia Computer Science 93 (2016): 554-563

[23] Akhtar, Md Shad, et al. "A hybrid deep learning architecture for sentiment analysis." Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. 2016.

[24] Garg, Komal, and Preetpal Kaur Buttar. "ASPECT BASED SENTIMENT ANALYSIS OF HINDI TEXT REVIEW." International Journal of Advanced Research in Computer Science 8.7 (2017).

[25] Ma, Xuezhe, and Eduard Hovy. "End-to-end sequence labeling via bi-directional lstm-cnns-crf." arXiv preprint arXiv:1603.01354 (2016).

