Aspect Based Sentiment Analysis using Bidirectional Residual Gated Units on Electronic Reviews

By:
Group No. 1 –
Syed Muntazar Abbas 059,
Avanish Gupta 014,
and **Gaurav Yadav 023**

Mentor: **Dr. Alok Kumar**

University Institute of Engineering & Technology,
CSJM University,
Kanpur

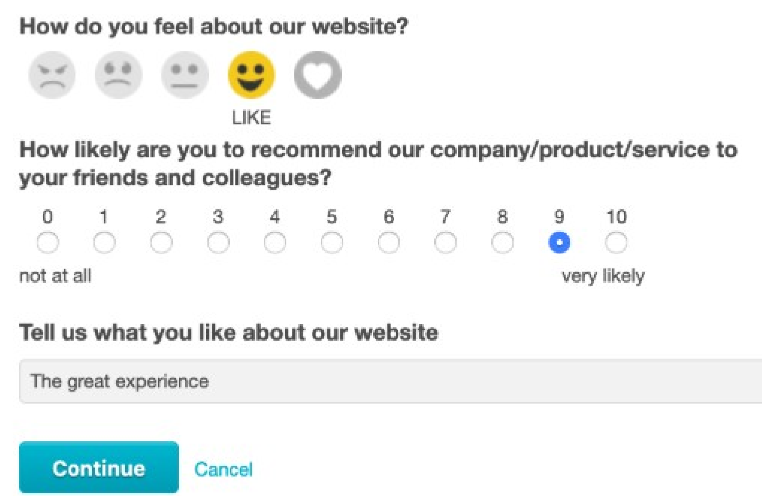
Background

- Corporations and Enterprises rely on Consumer feedback in order to improve their Products and/or Services.

- Feedback forms and Review forums were used to manually collect this information.

- e-Commerce enabled consumers to review products and services online. This was easy for consumers , hence the volume of data became very large.

- This led to the need to develop automated methods of handling reviews and feedback.


How do you feel about our website?

LIKE

How likely are you to recommend our company/product/service to your friends and colleagues?

0 1 2 3 4 5 6 7 8 9 10

not at all — very likely

Tell us what you like about our website

The great experience

Continue Cancel

Problem Statement

- Given a set of Sentences : S= {s(1),s(2),s(3)..s(n)}, for each Sentence s(i):

Part A: To identify the mentioned aspects of a product in the sentence

Part B: To identify the sentiment expressed by the author on each aspect identified in part A.

Example:

Sentence: The battery life of this laptop is really good.

Part A: The **battery** **life** of this laptop is really good
battery {ASP} **life** {ASP}

Part B: The **battery** **life** of this laptop is really good
battery {POS} **life** {POS}

{POS}= Positive Sentiment
{NEG}=Negative Sentiment
{NEU}=Neutral (Ambiguous) Sentiment

{ASP}= Aspect

Related Work

- **Liu et al.** Utilized BiReGU recurrent networks to label both aspects and sentiments.Employing a Cross Shared Unit. **(2019)**

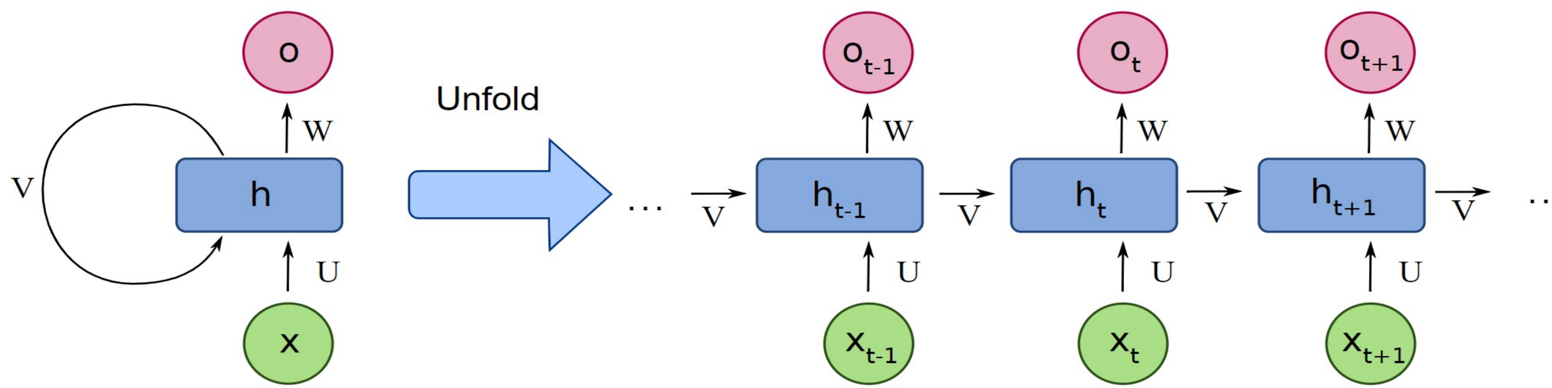
- **Xu et al.** Utilized the BERT language model which is pretrained on a large
english corpus like wikipedia to predict sentences. **(2019)**
- **Wang et al.** utilized LSTM for Twitter sentiment classification by simulating the interactions of words during the compositional process. **(2018)**

- **Huang et al.** proposed to encode the syntactic knowledge (e.g., part-of-speech tags) in a tree-structured LSTM to enhance phrase and sentence representation.

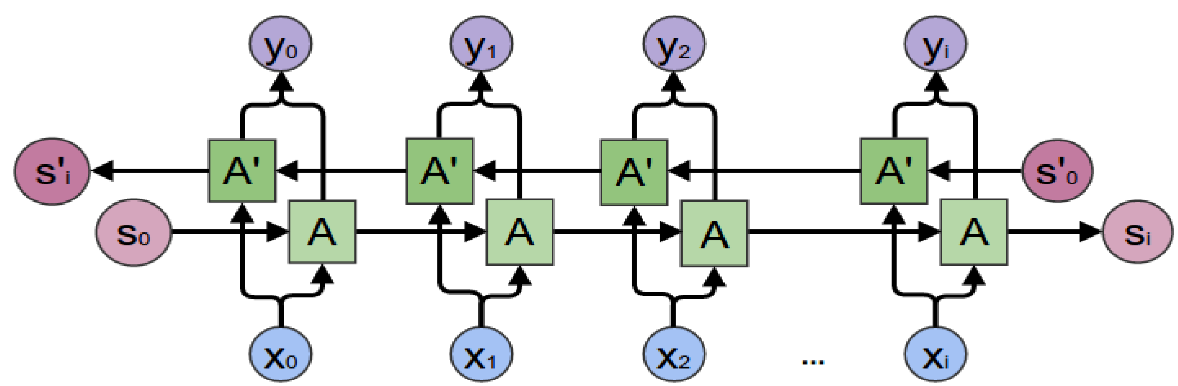
All of these approaches employed Recurrent Neural Networks to analyze sentences as a sequence of words with slight variations.

Methodology: RNNs

- **Recurrent Neural Networks:** a Deep Neural Network that accepts Variable sized sequences and labels each element of the sequence a class label.



- **Bidirectional NNs:** Recurrent Neural Networks that operate along both directions of a sequence.



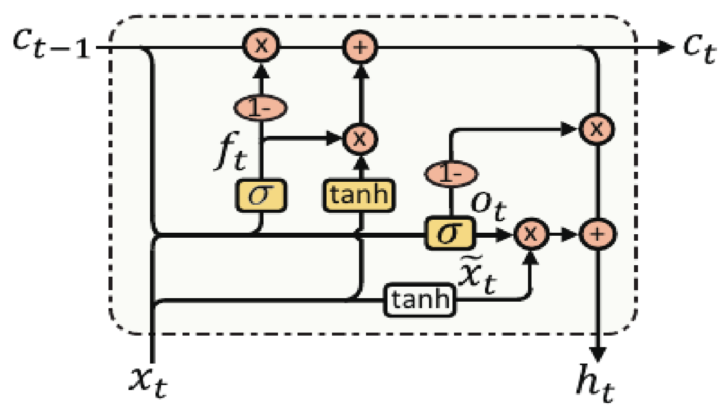
Methodology: ReGUs

- **Recurrent Neural Nets** suffer from vanishing/exploding gradients and tends to forget information of more than 10 elements before.Hence an RNN won't work on a sentence of more than 10 elements.

- **Residual Gated Units:** Each RNN cell employs gates that learn to selectively remember information of previous elements in the sequence.

- **e.g. a particular sentiment word only addresses an aspect word in a review sentence.**

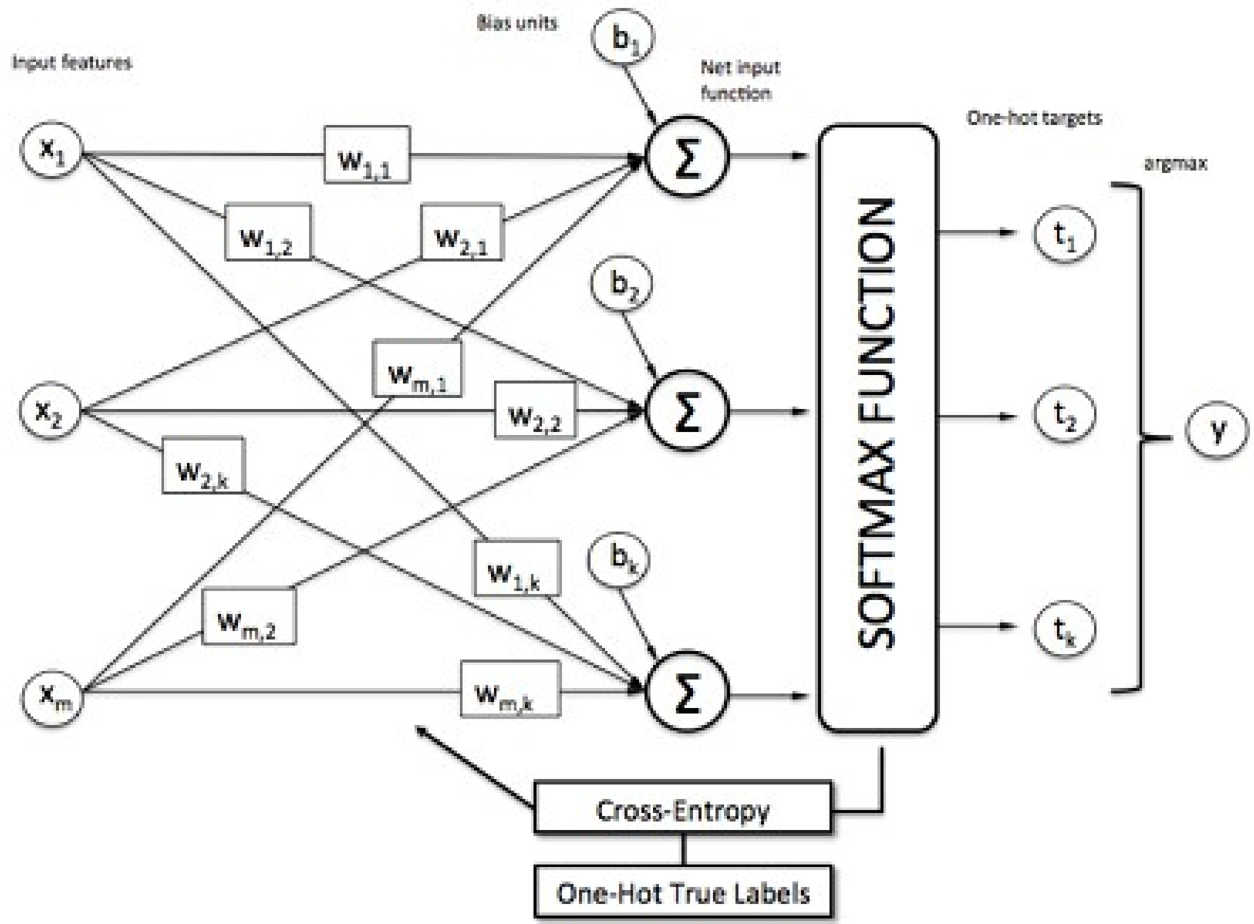
- They use **skip connections, forget & output gates** to encode information of larger sequences efficiently.



A Residual Gated Unit

Methodology: Classification

- **Softmax Classifier:** A multi class classifier that uses the exponential Function to compute a distribution over all the possible class labels From the output of an RNN unit at a particular position.



Methodology: Word Embeddings

Neural Networks need a numeric representation of words:

- **One Hot Encoded Vectors** : too high in dimensionality + contain no Semantic or syntactic knowledge.

- **Word Embeddings:** less dimensions, more semantic & syntactic knowledge.

  –Unsupervised training on a general corpus
  Examples: Word2Vec, Glove.

Rome Paris word V

```
Rome   = [1, 0, 0, 0, 0, 0, ..., 0]
Paris  = [0, 1, 0, 0, 0, 0, ..., 0]
Italy  = [0, 0, 1, 0, 0, 0, ..., 0]
France = [0, 0, 0, 1, 0, 0, ..., 0]
```

Word Embeddings

Rome = [0.91, 0.83, 0.17, …, 0.41]

Paris = [0.92, 0.82, 0.17, …, 0.98]

Italy = [0.32, 0.77, 0.67, …, 0.42]

France = [0.33, 0.78, 0.66, …, 0.97]

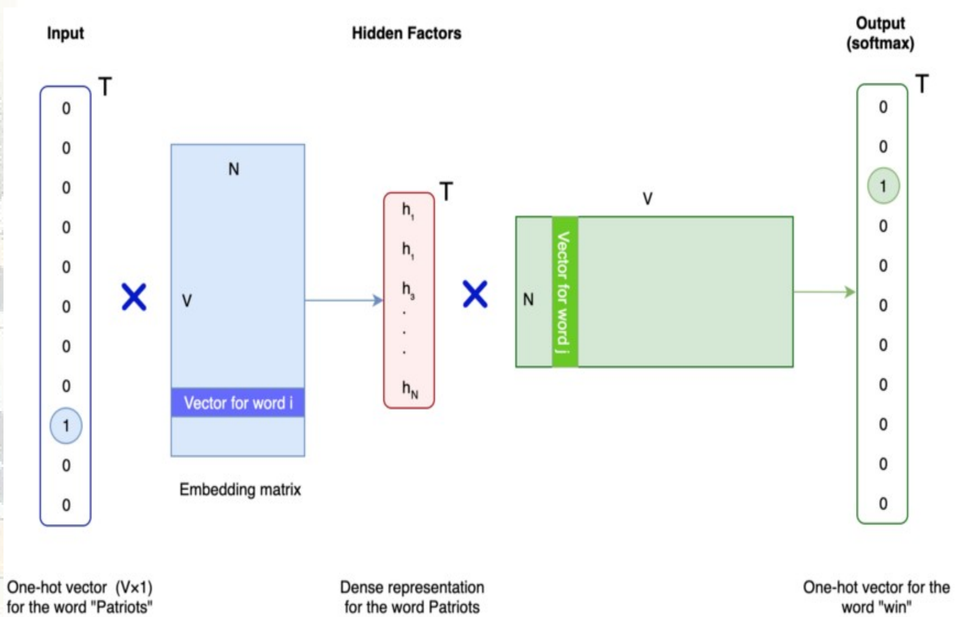
Methodology: Glove

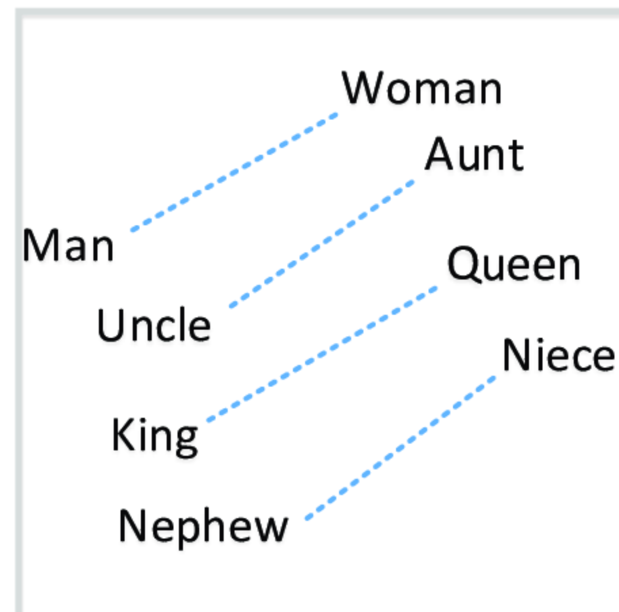
Glove: Global Word Vectors:

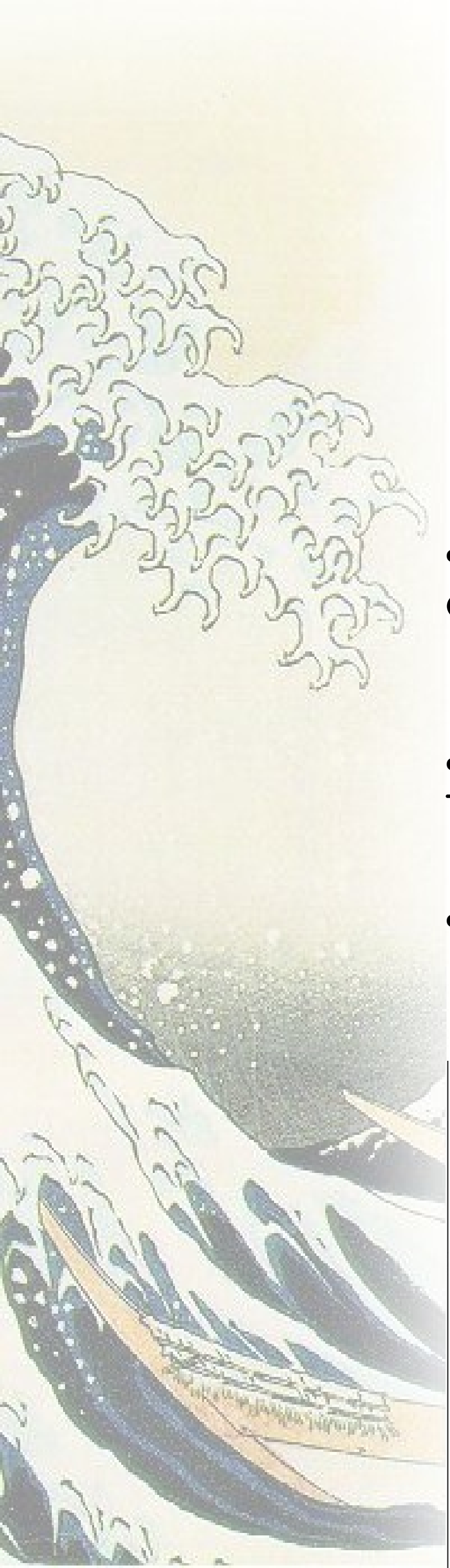
- Based on the idea of **distributional Semantics.**

Unsupervised training Algorithm:

- Calculate Word **Co-Occurrence matrix** from a given corpus.
- Use a neural network to determine the co-ocurrence probability of 2 input words each from the corpus.(calculated before)
- After training: **one hot enc vector * weight matrix = word embedding**







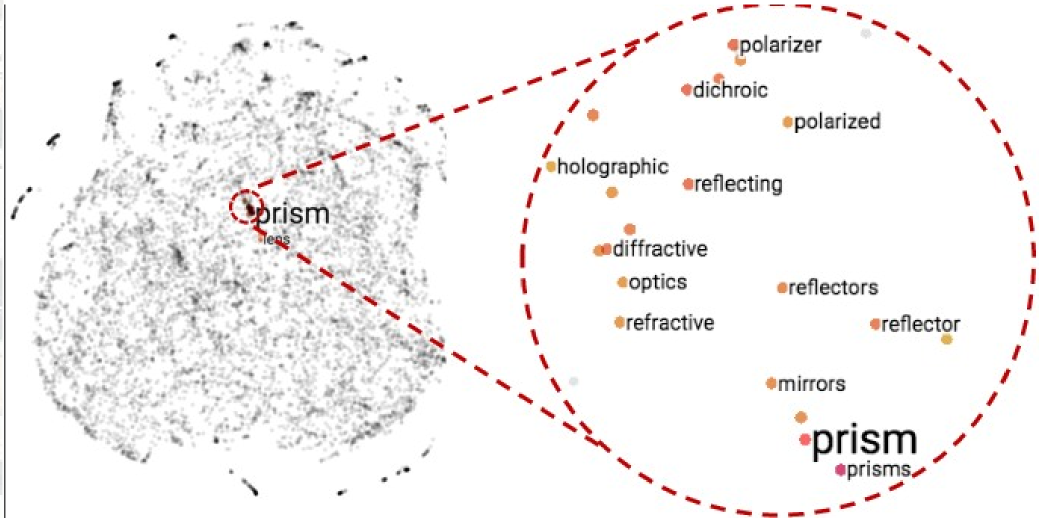
Methodology: Double Embeddings

To recognize and differentiate domain specific aspect words, we need Domain specific embeddings.

- Word Embeddings described before are trained on a general English Corpus e.g. Wikipedia.

- Amazon pretrained embeddings on the 'Electronic' Domain were used for This purpose.(Size = 100)

- The extra embedding was concatenated with the original word embedding For each word. Effectively increasing size of each word vector to **400.**



Methodology: Sentiment Lexicon

- A Sentiment Lexicon: Set of words in a particular language annotated by A specific sentiment value each.

- Each word embedding is concatenated with it's sentiment value

- -1 for negative , 0 for neutral and 1 for positive sentiments respectively.

Lexicons Used in the Project:

* Opinion Lexicon (a list of 6800 positive and negative words) by
Hu and Liu ,KDD-2004

* Hindi SentiWordnet by **A. Das and S. Bandyopadhyay.(SentiWordNet for Indian Languages), 2010**

Categorical		Numerical	
word	sentiment	word	sentiment
nice	pos	nice	2
beautiful	pos	beautiful	3
amazing	pos	amazing	4
ugly	neg	ugly	-3
stupid	neg	stupid	-2

Methodology: Labeling Method

- Each word in a Sentence is labelled with 'B'<Begin>, 'I'<Inside> or 'O'<Outside> for each aspect.

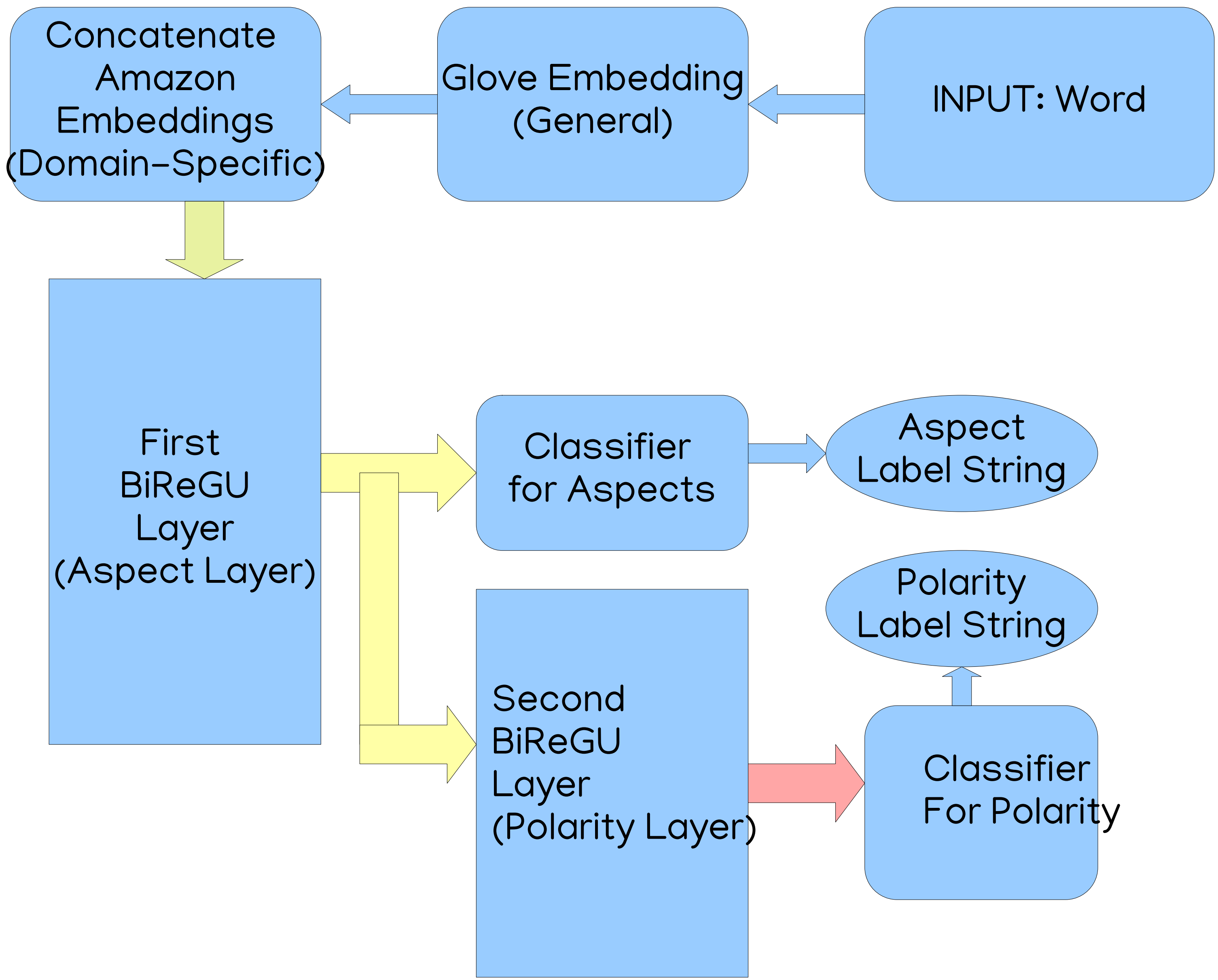
- This generates an aspect label String:

Example:	The	**battery**	**life**	of	this	laptop	is	really	good.
Output:	O	B	I	O	O	O	O	O	O

- Polarity Labelling is done in the same way, with labels : 'P' <Positive> , 'N' <Negative>, 'E' <Neutral> and 'O' <Other>

Example:	The	**battery**	**life**	of	this	laptop	is	really	good.
	O	P	P	O	O	O	O	O	O

Methodology: Overview



Alternative Approach: SVM

- The aim of implementation of this model is to explore other usable techniques to check whether they can be used to improve the overall performance.
- This system is based on supervised learning using Support Vector classifier.
- Performance of the system is also compared with other classifiers.

Alternative Approach: SVM implem.

- Import important libraries and various classification models.
- Set path to the dataset XML file and ‘JAVAHOME’ path.
- Applying all predefined functions to create dataframe .
- Sort the data frame according to aspect's name and separate data(X) and
- Sorted uneven datafields and operated to make it even and operable.
- Created various models and predicted test data.

Alternative Approach :SGD Classifier

- Linear classifier that uses Stochastic Gradient Descent for training.
- Works with data represented as dense or sparse array of floating point values.
- It uses convex loss function such as SVM or logistic regression.
- SGD is much faster than gradient descent while dealing with large dataset

Alternative Approach: OneVsRest Classifier

- It consist in fitting one classifier per class.
- It is a heuristic method using binary classification for multiclass classification.
- It is computationally efficient and easily interpretable.
- $P(C)=max(P(c1),p(c2)\ldots\ldots\ldots P(cn))$
- $P(c1)=fc1(X)$

Alternative Approach: Multinomial Naïve Bayes Classifier

- Suitable for classification with discrete features.
- It normally requires integer feature counts.
- Gaussian Distribution is used.
- P(f1,f2….fn|c)=Π P(fi|c)   where i=1,2…..n
- P(c|f1,f2…fn) $\propto$ P(c).p(f1|c)…….p(fn|c)
- Here p(fi|c) is a multinomial distribution.

Tools Employed

- **PyTorch:** a fast deep learning library written in python , developed by facebook

PyTorch

- The **gensim** library contains multiple Word embedding algorithms for Natural Language Processing

gensim

Datasets

- English:

Semeval 2014 Laptop Review Dataset;

No. of Sentences containing Aspect words: 1488

```
<text>
I charge it at night and skip taking the cord with me because of the good
battery life.
</text>
<aspectTerms>
<aspectTerm term="cord" polarity="neutral" from="41" to="45"/>
<aspectTerm term="battery life" polarity="positive" from="74" to="86"/>
</aspectTerms>
```

- Hindi:

Hindi Electronics Review Dataset with annotated aspects and polarity labels By **Md Shad Akhtar, Asif Ekbal, Pushpak Bhattacharyya; Aspect Based Sentiment Analysis in Hindi: Resource Creation and Evaluation, 2016**

No. of Sentences with Aspect Words: 3318

```
<text>
फेसबुक का सिक्योरिटी चैकअप फीचर पॉपअप की तरह यूजर्स को दिखाइ देगा।
</text>
<aspectTerms>
<aspectTerm from="10" polarity="neu" term="सिक्योरिटी चैकअप फीचर" to="31"/>
</aspectTerms>
```

Experimental Setup

- **LEARNING RATE:** 0.1
- **REGULARIZATION:** L2 (parameter: 0.001)
- **LOSS FUNCTION:** Cross Entropy Loss
- **OPTIMIZATION FUNCTION:** Stochastic Gradient Descent
- **TRAINING ALGORITHM:** Back Propagation Through Time
- **EPOCHS:** 100

DATASET BREAKUP

English	Hindi
Train: 1000 sentences	Train: 1000 Sentences
Test: 488 sentences	Test: 500 Sentences

Results

Primary Approach:

The F1 Score has been used as a metric for evaluation.

	English	Hindi
Aspect Term Labeling-	F1: 79.35%	F1: 67.36%
Polarity Detection-	F1: 67.34%	F1: 53.47%

Alternative Approach:

The recall score has been used as a metric for evaluation.

Support vector classifier: 69.3%

Results from other classifiers:

Linear support vector classifier : 66.7%
SGD classifier : 55.73%
Multinomial naïve bayes : 49.47%

Conclusion

In this project we experimented with the best Performing methods for natural language processing available and applied it to the problem using datasets of multiple languages implementing in a relatively new deep learning framework. The challenges of high computation were offset by employing Google Collab and other cloud based services when at times the computation became time consuming. Albeit there are better performing methods available, but they often sacrifice time as well as space complexity. . The application of this method in Hindi was very experimental and new, yet it performed in line with our expectations. Overall this project has been an incredible learning experience for us, and will help us to further navigate the world of AI and NLP with ease and a sense of familiarity.

Future Scope

We would like to explore the possibility of using attention vectors and sentiment lexicons in order to boost polarity detection. Better representation of technical terms is needed in order to increase aspect detection performance. Better domain specific word embeddings are needed for the domain in which we intend to work. I used only general hindi text embeddings for hindi version of this model, domain specific embeddings for hindi would greatly increase it's performance for both the subtasks. Hence in the future I would like to make these additions to the currently presented model.