Detecting Emotional Scene of Videos from Subtitles

Utsav Sinha, 12775   Rajat Kumar Panda, 12545
Guided by Professor Amitabha Mukherjee
CSE Department, IIT Kanpur

Motivation

With the explosion of multimedia content on the web, it becomes increasingly difficult to find the movie of one's choice. Classification of movies mainly relies on human reviews and ratings.

This project aims to automatically tag a movie and pin-point out the emotional scenes in an unsupervised manner. Finding the highlights of a video helps to get a quick review in advance or to watch the major scenes once again. This can further be extended for genre classification, recommendation systems and for detecting profanity.

Introduction

Multimedia expresses emotions using:

Dialogue Content	Music
Facial Expressions	Way of Articulation
Background Scene	The Context

In this project, we are applying Natural Language Processing (NLP) techniques on video subtitle dialogues to accomplish the task of detecting emotional scenes by classifying emotions into love, happiness, surprise, emotionless, sad, disgust, anger and fear.

With extensive learning on large corpus, the pragmatic inferences along with the semantic context of phrases can be derived. This would be used to automatically annotate sentences which would then be trained on sentiment labeled dataset.

Corpus Creation

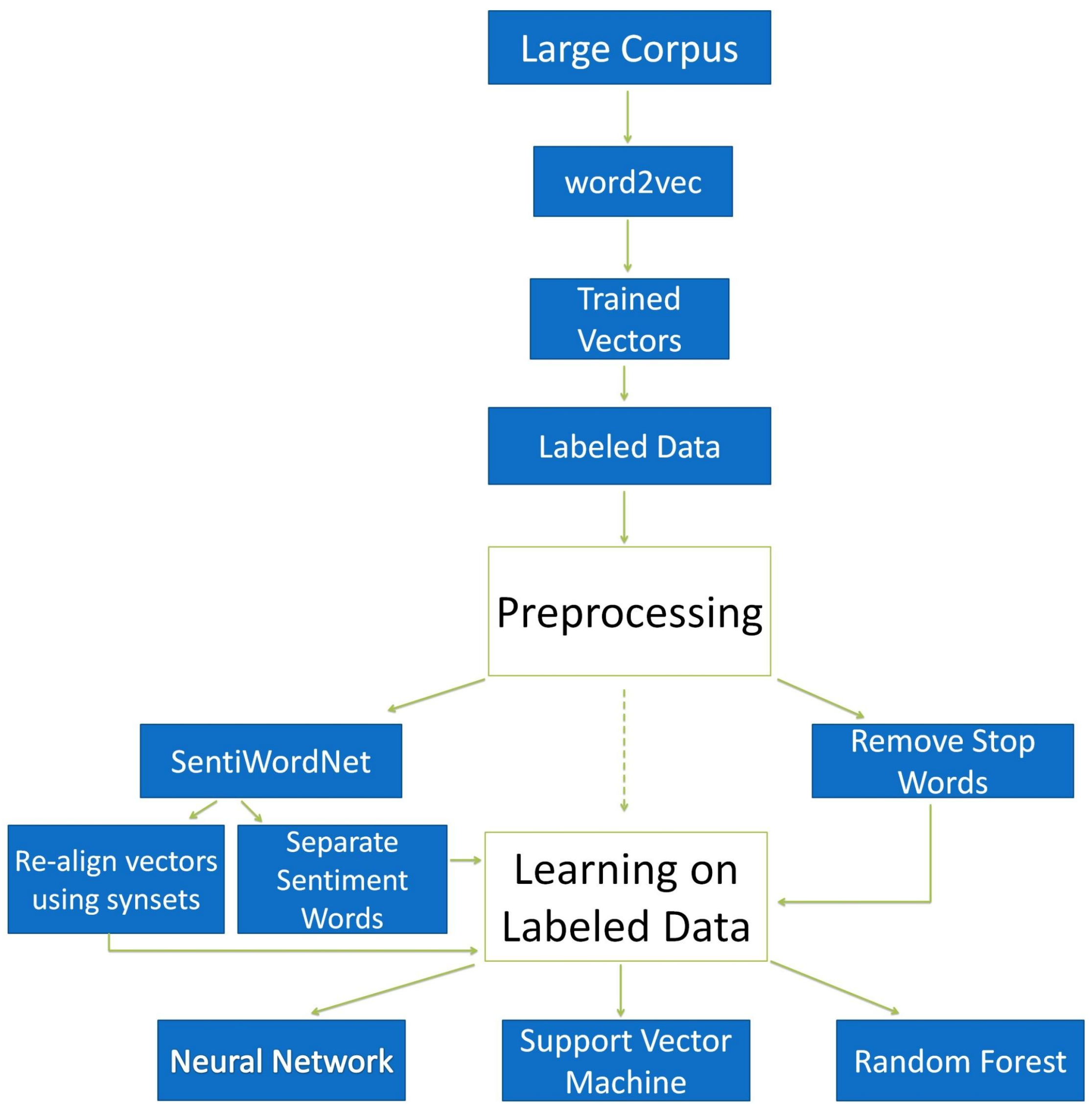
Video subtitles were obtained from [pod] and [tvs] Since subtitles with labeled emotions are not available, so a few TV-episodes and movie subtitles were hand-tagged with emotional labels which formed the ground truth against which our proposed model would learn from and would later be compared with.

Related Work

Classification of sentences into binary sentiment classes has been done extensively, but tagging into multiple emotional clusters has not been done in detail. [PYKJ11] outlines ideas based on wordNet.

[KCK09] works towards the same goal but uses Naïve emotion count classifier and Maximum Entropy classifier with Unigram model. Their results:

Movie	Accuracy
Gladiator cd1	72%
Love Actually	48%
Troy	65%
X-men 1	62%



Large Corpus

word2vec

Trained Vectors

Labeled Data

Preprocessing

SentiWordNet

Remove Stop Words

Re-align vectors using synsets

Separate Sentiment Words

Learning on Labeled Data

Neural Network

Support Vector Machine

Random Forest

Figure 1. Implementation Flow Chart

Implementation

Wikipedia corpus is used in training by Word2vec [MTC+13]. This provided a 100 dimensional vector for each word(with at least 5 occurrences) in its training corpus. The skip-gram neural network Word2vec implementation uses context of sentences to build vectors in an unsupervised manner.

The labeled data is now used for sentiment tagging (Figure 1). The vectors of each word in a dialogue are summed and its average is send as the sentence vector. along with the labeled emotion to the learning phase.

But before that, pre-processing is done to remove non sentiment words and to give more emphasis on sentiment words taken from SentiWordNet [BES10].

Reweighing of the vectors of major emotions is also done as a testing parameter[PYKJ11]. This process helps to bring together emotional words such as "pleasant", "delight", "cheerful" closer together to the major emotion of "happiness".

Learning on labeled data is done using:
- Neural Networks
- Support Vector Machine (SVM) Classification
- Random Forest

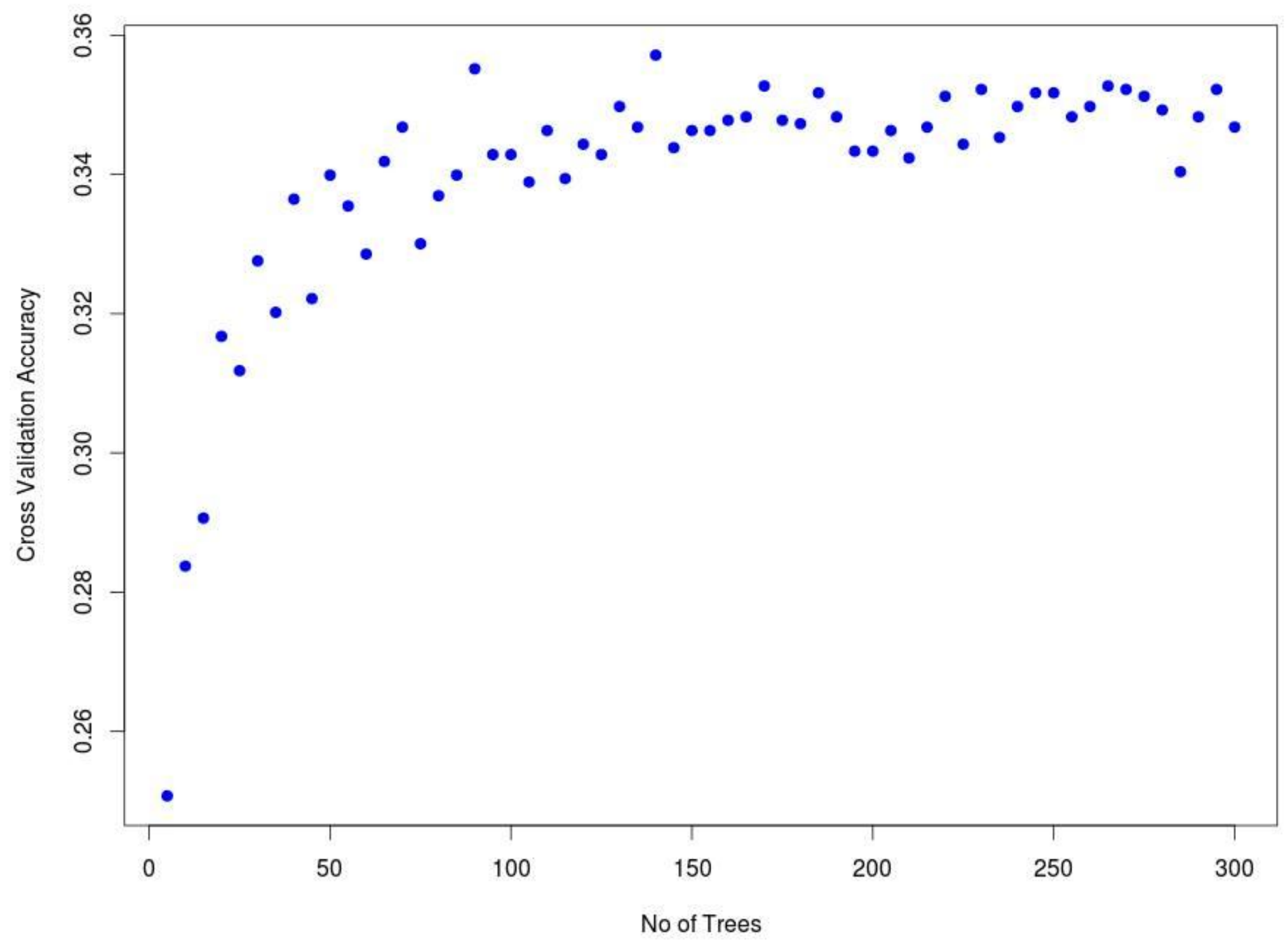


Figure 2. Plot of Cross Validation Accuracy vs No of Trees in Random Forest

Results

10-fold cross validation is used to calculate accuracy of models. Results are tabulated below which shows accuracy scores in percentage. The second row represents accuracy calculated after removing "emotionless" while first row results includes it. Figure 3 shows the accuracy of different emotions using SVM.

Video	Raw word2vec	Neural Network	SVM	Random Forest
Titanic	32.34 20.32	38.21 27.80	39.74 34.70	36.46 24.80
Walking Dead S01E01	33.20 24.78	36.63 29.14	38.93 31.52	34.72 26.41
Friends S05E14	23.31 17.78	27.56 21.33	32.10 25.31	29.76 22.92
Overall	29.82 21.44	34.67 30.04	37.65 32.90	33.58 25.27

Table 1. Accuracy of Different Test Data on various Models

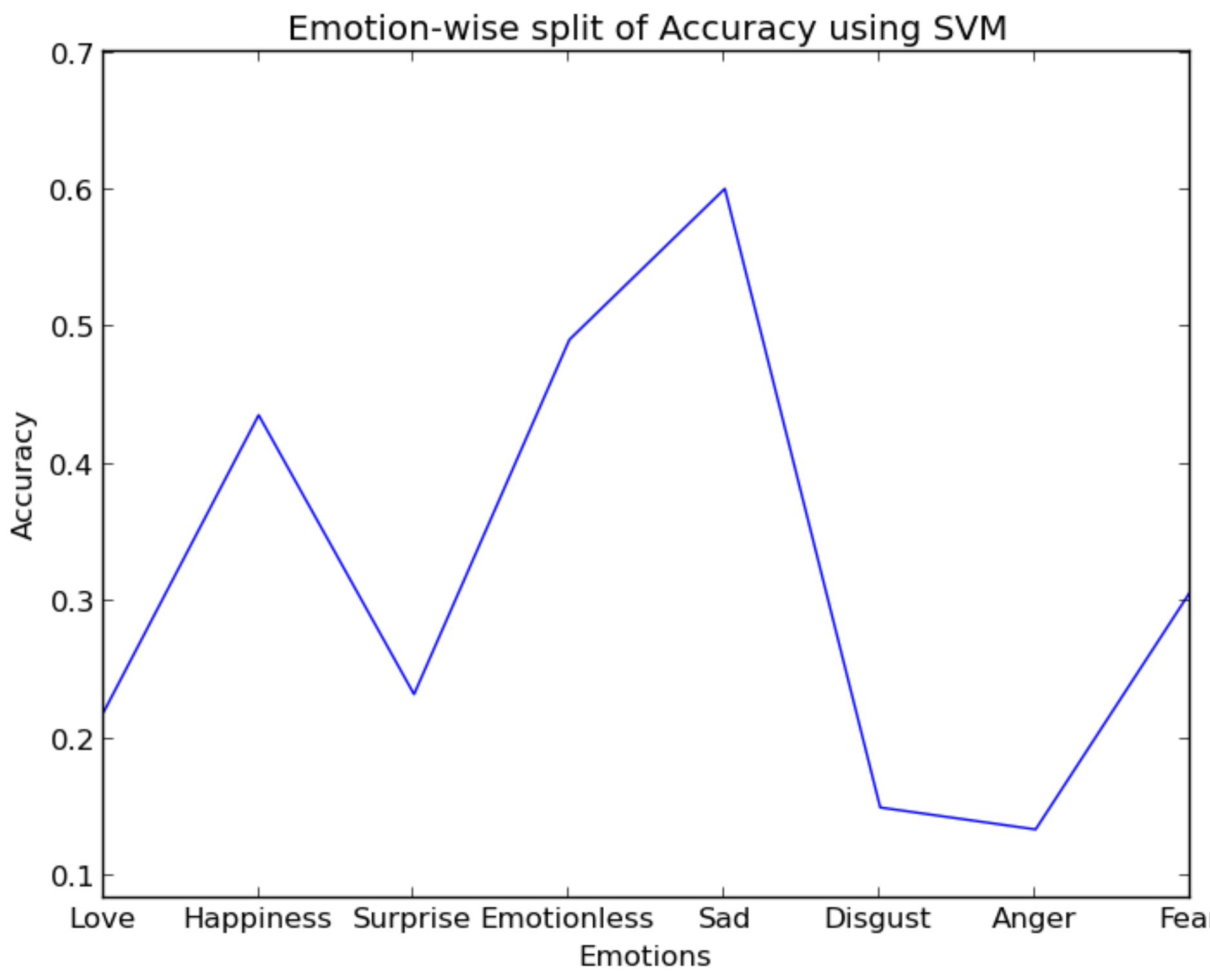


Figure 3. Accuracy of SVM on movie Titanic

Conclusion

We did not use Maximum Entropy Model as learning similarity between words is done by Word2Vec whose skip-n gram model captures context more accurately.

Our accuracy as compared to [KCK09] is lower since we tagged emotion of each dialogue instead of a movie scene (group of 20-25 sentences). Also, we used eight major emotion clusters instead of six.

Lower performance on comic video "Friends" is due to dominance of negative emotions in training data and selection of major emotions.

Future Improvements

More labeled data could be used for training
New metric can be used for calculating sentence vector [LM14]
Kernel Functions can be used to improve SVM result

References

[BES10] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Senti-wordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining 2010
[MTC+13] Mikolov, Tomas and Chen, Kai and Corrado, Greg and Dean, Jeffrey. Efficient estimation of word representations in vector space, 2013
[KCK09] Kalyan, Chetan, and Min Y. Kim. "Detecting emotional scenes using Semantic Analysis on Subtitles." 2009.
[PYKJ11] Seung-Bo Park, Eunsoon Yoo, Hyunsik Kim, and Geun-Sik Jo. Automatic emotion annotation of movie dialogue using wordnet. 2011
[LM14] Quoc Le, Tomas Mikolov Distributed Representations of Sentences and Documents, Google Inc 2014
[pod] podnapisi.net Subtitles datasource
[tvs] Tvsubtitles.net Subtitles for TV Shows

Poster Template Designed by Genigraphics ©2012
1.800.790.4001 www.genigraphics.com