

## **Abstract**

People often tend to listen to similar songs according to their taste and mood. I intend to classify the songs as “Happy” or “Sad” and provide recommendation based on the lyrics of the songs by using ensemble method which makes use of many state-of-the-art classifiers and take majority voting based on their result labels and then assign the label to the song. In the first phase of the project, I would classify the songs using classifiers where the input to the classifiers would be Bag Of Words. In the second phase of the project, I would use Generative adversarial networks, where the network would generate a new lyrics based song by taking just the tag as an input. I believe training a stable GAN would be challenging. I intend to apply different architectures and mathematical models (such as employing different divergences) for training a stable GAN.

## **Custom Dataset**

1. Get the list of happy songs from trustable site <https://www.quora.com/What-are-some-happy-love-songs>
2. Get the list of Song songs from a trustable site <http://www.worldssaddestsongs.com/page1.html>
3. Prepare a URL of the format [http://lyrics.wikia.com/wiki/Ed\\_Sheeran:Shape\\_of\\_You](http://lyrics.wikia.com/wiki/Ed_Sheeran:Shape_of_You)  
“http://lyrics.wikia.com/wiki/”+ singer\_name:song\_name.
4. Python based Web Scraper script would do a HTTP GET with the above url and the response object received is in the HTML format.
5. BeautifulSoup will be used to parse the HTML content to extract lyrics text.

## **Statistics**

Number of Happy Songs: 692

Number of Sad Songs: 698

Total Number of Songs: 1370

Total Size of the Dataset: 1.072MB

## **Data Preprocessing**

Stem words using NLTK snowball stemmer

Remove Stop words using NLTK (stopwords from english dictionary)

Minimum length of the word = 3

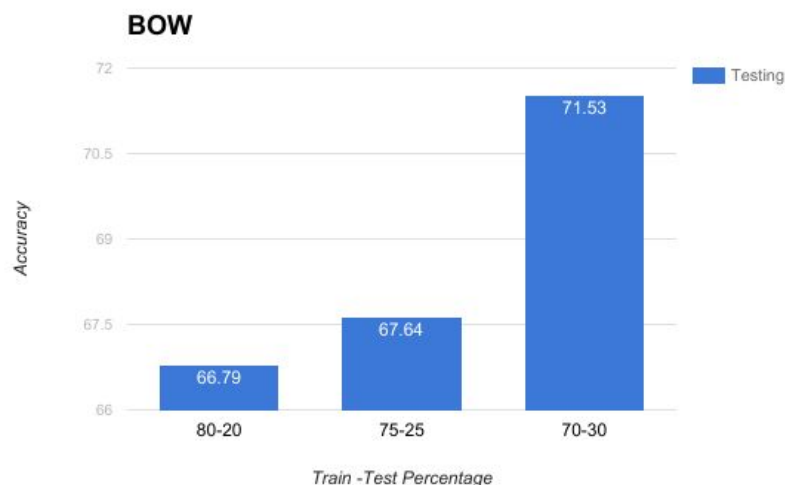
## **Implementation**

Phase1- Combine different classifiers in an ensemble to have the most accurate recommender system. Different classifier would be KNN classifier, Support vector machines, Multinomial Naïve Bayes, Random Forests, Decision Trees, Quadratic Discriminant Analysis, Neural Networks, and Gaussian Naïve Bayes.

Phase2- Train Generative adversarial network with different architectures and mathematical models to generate new lyric based songs.

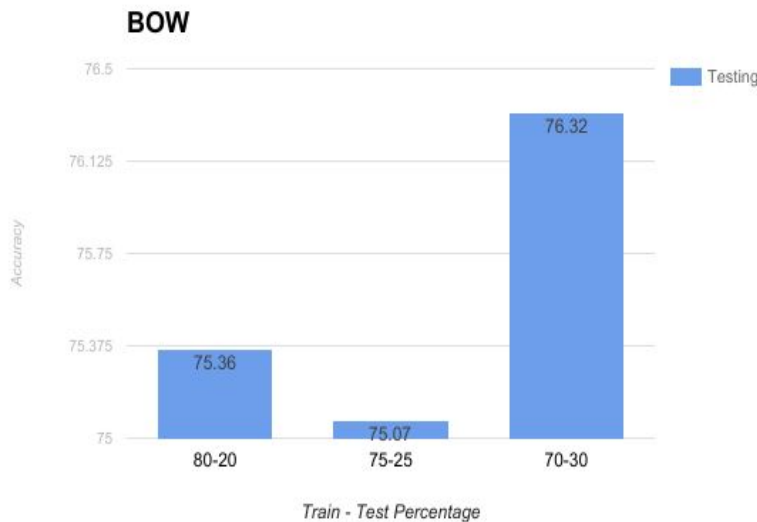
## **Previous Results and Plots**

KNN classifier: K =37, search algorithm: KDTree



Plot1: Accuracy achieved using KNN classifier when train and test dataset ratio was 70-30% was 71.53.

## SVM classifier



Accuracy achieved using SVN classifier is 76.32 when train and test dataset ratio is 76.32

### **Work in Progress and Future**

Phase1 -Using other classifiers such as Multinomial Naïve Bayes, Random Forests, Decision Trees, Quadratic Discriminant Analysis, Neural Networks, Gaussian Naïve Bayes and ensemble method.

#### Phase2-

- Train different architecture models of Generator and Discriminator by applying deep learning techniques and mathematical models.
- Apply KL divergence and Jensen divergence.
- Overcome Mode collapse
- Add Dropout in generative network in both train and test phase
- Add noise to the inputs of discriminator.
- Normalize the inputs

