ML and Neural Networks for Sentiment Analysis

MENTOR: Drushti Apoorva Gopikrishnan

TEAM: Anandita Dhasmana, Kanishka Joshi, Kritik Mathur, Subrito Haldar

Problem Statement

To employ and compare different Ml and NN techniques for hindi song lyrics and make a robust system that gives substantially accurate results.

Motivation

Extracting the emotion of a song has an important application in auto playlist generation, song recommendation etc.

Audio features have been tapped for this purpose but they are often not consistent with the emotion expressed by the lyric. So we looked into lyrical features to evaluate their significance in making a multimodal system.

Challenges

- No fixed syntactic structure to a sentence
- Non-sentimental phrases that repeat several times
- Extensive use of metaphors
- A song doesn't necessarily have just one emotion

तिरकिट ताल से लो चली कहानी
पनघट काल से लो चली कहानी
हो सरपट दौड़ती है फ़क्त जुबानी
छूट-पूत आशिकी में ढली कहानी

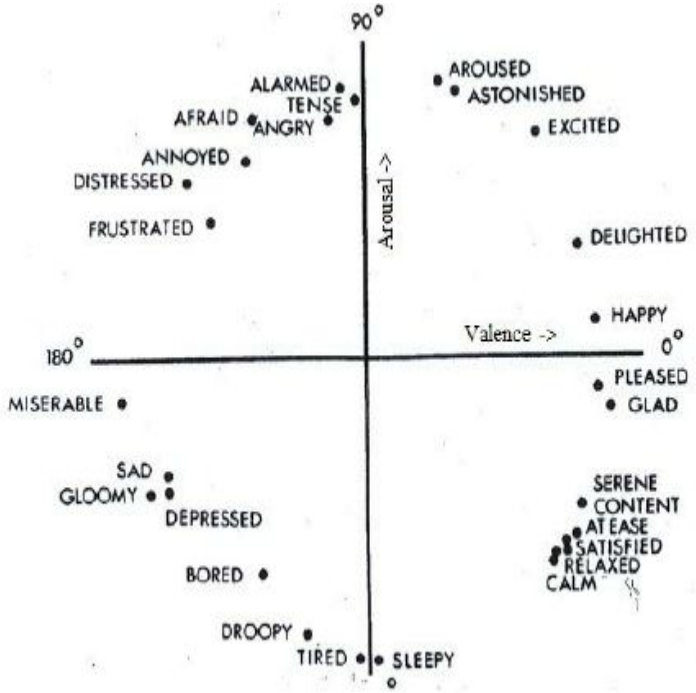
बिरहा का दु:ख काहे हो बांकिये
दिखे मोहे तू ही जो जिया में झांकिए

पल पल गिनती हूँ आठों ही पहर
कितने बरस हुए मोहे हाँ किये
नैना निहारों मोरे भोरे से झरे
प्रीत मोरे पिया बातों से ना आंकिए
मैं ही मर जाऊं या मारे दूरियां

Dataset

Two sets of lyrics:

- A collection of 1055 hindi songs lyrics in Devanagari script (2.6MB, 9937 tokens)
- A collection of 2553 hindi songs lyrics in Roman script



Approach

Taxonomies experimented with:

- 2-class taxonomy (positive / negative)
- 4-class taxonomy
  - class_E (Excited, Astonished, Delighted...)
  - class _A (Angry, Alert, Afraid, Annoyed...)
  - class_C (Calm, Relaxed, Satisfied…)
  - class_D (Depressed, Sad, Gloomy…)

Approach (contd..)

Features experimented with:

- Bag of Words
- Tf-idf

$$w_{ik} = \frac{tf_{ik} * \log(N / n_k)}{\sqrt{\sum_{k=1}^{n} \left(tf_{ik} * \log(N / n_k)\right)^2}}$$

Bag of Words

2-class	fvalue	Accuracy
k-NN (n_neighbours = 5)	0.62	62.86
Gaussian NB	0.75	79.05
Multinomial NB	0.73	72.38
SVM	0.68	78.10
Multi-Layer Perceptron	**0.79**	**80.00**

4-class	fvalue	Accuracy
k-NN (n_neighbours = 5)	0.28	36.89
Gaussian NB	0.58	58.25
Multinomial NB	**0.60**	**60.78**
SVM	0.32	49.02
Multi-Layer Perceptron	0.55	57.28

TFIDF

2-class	fvalue	Accuracy
k-NN (n_neighbours = 5)	0.74	75.24
Gaussian NB	0.75	77.36
Multinomial NB	0.72	80.19
SVM	0.67	77.14
Multi-Layer Perceptron	**0.81**	**80.95**

4-class	fvalue	Accuracy
k-NN (n_neighbours = 5)	0.50	49.51
Gaussian NB	0.55	57.84
Multinomial NB	**0.56**	**60.78**
SVM	0.31	47.57
Multi-Layer Perceptron	0.50	54.37

Conclusion

- Multinomial Naive Bayes performs the best with Bag of Words (Accuracy 60.78%, F1 Score 0.60)
- Multinomial Naive Bayes performs the best with TFIDF (Accuracy 60.78%, F1 Score 0.56)

Future Work

- Transliterate Roman data to Devanagari and make a larger dataset
- Use SentiWordNet for Hindi to get better results
- Consider modifiers to emotion words (कुछ, बहुत, थोड़ा etc.)