2016 International Conference on Information Technology (InCITe) - The Next Generation IT Summit

Music mood classification based on lyrical analysis of Hindi songs using Latent Dirichlet Allocation

Swati Chauhan, Prachi Chauhan
School of I.C.T
Gautam Buddha University
Greater Noida, Uttar Pradesh, India

***Abstract*—For over a decade now, due to the introduction of UTF-8 encoding, the digitization of Hindi content has increased rapidly because of which Hindi-music has accomplished popularity on the web. The focus is to identify the emotion, a person is experiencing while listening to a song track. The aim of this research work is to analyze the lyrics of Hindi-language based songs, in order to detect the mood of the listener. We used unigram and term-frequency as the main features. The songs were reduced to a level where only relevant words will be used for mood-detection. We employ unsupervised machine learning namely topic-modeling (Latent Dirichlet Allocation model) for mining the mood out of every song in the corpus. We created our own dataset of 1900 songs consisting of Bollywood tracks, bhajans (spiritual prayers) and ghazals. A mood taxonomy is used to distinguish songs into Happy or Sad. Data is applied to LDA model to discover the hidden emotions within each song. At the end of experimentation, we compare the results with manually pre-annotated dataset for validation purpose and observe good results**

***Index Terms*—emotion recognition, Hindi lyrics analysis, mood taxonomy, latent dirichlet allocation**

I. Introduction

Humans are inherent with emotions or in other words, emotions are what makes us human. The affective aspect of music also referred as music mood has been recently recognized as an important aspect of organizing and obtaining music information [6]. Music is one of the key human needs for leisure and recreation. Hindi content on the internet is on the edge for a big growth. The Hindi search queries have almost tripled in the last five years. Today, the online Hindi content consumption is rising at 94%. Thus, many blogs and websites are currently being written in Hindi-language, where content such as song lyrics is available in Indic-languages such as Hindi, Marathi, Bengali etc. This has resulted in an increase in the expansion of digital content day by day. Therefore, the need of a new classification system such as mood based will serve the purpose very well. It will be different from classic genre based systems. Our emotions rule our choice of music. This had led to an increase in research work in the field of browsing music based on its affective aspects. The paper is laid as: We review research works related to lyrics analysis or text analysis carried out in various languages. We learn about different approaches and thus observations are made about how precise & accurate these approaches were. Based on this, objectives are set and the methodology used and every stage in song analysis will be explained thoroughly. Lastly, experimentation is discussed and future work is suggested.

II. Related Work

A. Taxonomy

It has been suggested in [6] that emotion models can be generally classified in two categories

1) Parametric Classification model: Emotions can be defined according to one or more dimensions. These parametric models of emotion endeavors to classify human emotions by locating where these lie in two dimensions (valence-arousal) or three dimensions (valence-arousal-intensity).

a) Russell's Circumplex model of affect: This is a well-known two-dimensional model called the Valence-Arousal Model proposed by Russell in 1980. This model assigns one axis to represent the Arousal level indicating the intensity in the form of high (active) and low values (inactive) and the other axis to represent Valence, which is an assessment of polarity ranging from positive (happy) to negative (sad).

b) Thayer's model: This model implements the theory that mood can be derived from two factors: Energy (High/ Low) and Stress (Positive/Negative), thus taxonomy is divided into four clusters: Anxious, Contentment, Depression and Exuberance. The MIREX mood-taxonomy follows Thayer's model of emotion is widely used for emotion analysis in various languages.

2) Categorical Classification model: This categorical model is based on Rusell's model of affect that describes four bipolar dimensions spaced 45° apart-

- Positive Affect (excited/sluggish)
- Pleasantness (happy/sad)
- Negative Affect (distressed/relaxed)
- Engagement (aroused/still)

These quarters can be regarded as two pivots that decide positive and negative affectivity of a person. Many research works for sentiment analysis of Twitter data has also adopted this model

B. Approaches Used

Using the taxonomies described in the previous section, an extensive amount of work has been done on mood classification using audio, lyrics, social tags or the combination of either of these [6]. These features have been selected

978-1-5090-2612-8/16/$31.00 ©2016 IEEE

and accordingly various machine learning approaches were employed to classify the emotions.

As seen [1] worked upon English-song dataset which was classified into emotion categories based on a weighted combination of extracted lyrical and audio features. Using Russells Model of Affect, the features that classify songs were placed along two axes. Social tags from last.fm were collected and distributed amongst 9 mood categories. Feature weighting was performed as audio features were included too. The kNN classifier is improved as the Euclidean distance formula was modified to incorporate weights for each feature. This resulted in an accuracy of 83% in test-set. One song can be classified into any number of classes but it must have a sufficient number of neighbors in those classes hence fuzzy classification was also applied. For a test set of 795 songs, accuracy of 83.20% was observed by comparing the output to the classes derived from social tags.

As seen from [2], described a mood classification system. They developed a supervised system to recognize the sentiment of the Hindi songs based on lyrical features. They opted for Russells Circumplex model of affect and created five mood categories. LibSVM algorithm was trained by the dataset and it performed better than other classifiers such as SVM, SMO and KNN algorithm. Two systems were developed and compared. In the first system, lyrics were classified into five coarse-grained moods classes. In the second, lyrics were classified based the polarities (positive or negative) of textual content. In order to get reliable accuracy, a 10-fold cross validation was performed for both the systems achieved the maximum average F-measure of 68.30% for classification system and 38.49% for mood classification system.

Work in [3] developed a computational model to discover the moods from Hindi songs using several audio features like intensity, timbre, rhythm. Each songs audio was with a time frame of 30 seconds. They employed a decision tree classifier on these selected features and achieved an accuracy of 51.5% on a dataset of 230 songs consisting of five clusters of moods with three subcategories of mood within each of these.

As seen, [6] conducted a study on English songs based on hybrid features i.e. lyrics, audio and social tags. Audio and lyrics were used to build the classifiers while social tags for giving ground truth labels to the dataset. Using last.fm tagset, tags were collected and filtered to obtain 18 relevant tags i.e. 18 mood categories. Songs with only English lyrics were selected. SVM classifier was used on lyric-only, audio-only and hybrid features (late fusion and feature concatenation) and compared. It was seen that among all systems, the hybrid system using late fusion accomplished the best performance outperformed audio-only system by 9.6%.

III. Methodology

A. Topic Model

Topic model is a statistical model used for identifying soft clusters in a corpus of data. It is basically a method for tracking the cluster of words in large bodies of text. Topic is defined as a repetitive pattern of co-existing words. Here the topics will represent the emotions i.e. happiness, sadness, excitement or anger etc.

B. Latent Dirichlet Allocation (LDA)

Before LDA many probabilistic and nonprobabilistic methods have been proposed for text-analysis. The goal of these methods is to express each document in descriptions that can preserve fundamental statistical relationships. This is essential for document classification or summarization and similarity [7].

1) Terminology:

- Word: The central unit of text, represented as $\omega \in \{1,...,V\}$ [7], [4].
- Document: It is a collection of N words represented by $W = \{\omega_1, \omega_2, ..., \omega_n\}$ [7], [4].
- Corpus: A collection of M documents, represented by $D = \{W_1, W_2, ..., W_m\}$ [7], [4].
- Topic: This a probability distribution over the set of V words, denoted as $z \in \{1,...,K\}$. The generative process predicts that each word within a document is produced by its own topic, so $\mathbf{z} = \{z_1, z_2, ..., z_n\}$ denotes the sequence of topics over all words in a document [7], [4].
- α the concentration parameter of Dirichlet prior on per document-topic distribution.
- β the concentration parameter of Dirichlet prior on per topic-word distribution.
- Θ_m topic distribution for document m.
- φ_k word distribution for topic k
- z_n topic for n^{th} word in m^{th} document.
- ω_n a specific word i.e. the n^{th} word in m^{th} document.

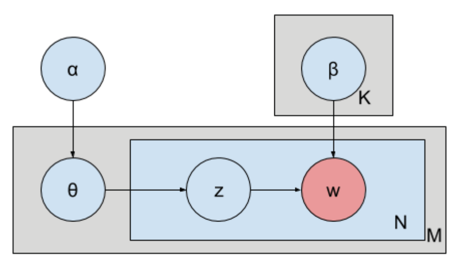

Fig. 1. Plate notation shows joint distribution on entire corpus

2) The Generative Process: Consider a corpus D, for each document i.e $m \in \{1,...,M\}$ consisting of length N_i

- Choose a K-dimensional topic weight vector Θ_m from the distribution $p(\Theta \mid \alpha) = Dirichlet(\alpha)$
- Choose a V-dimensional word weight vector from φ_k the distribution $p(\Theta \mid \beta) = Dirichlet(\beta)$
- For each word with index $n \in \{1,...,N\}$ in a document:
  1) Choose a topic $z_n \in \{1,...,K\}$ from the distribution $p(z_n = k \mid \Theta_m) = \Theta_k^m$
  2) Given this topic z_n, select a word ω_n from the probability distribution $p(\omega_n = i \mid z_n = j, \beta) = \varphi_{ij}$ [7], [4].

The generative process defines a joint distribution for each document W [7], [4]. as shown in fig1. If the parameters α and β are known, the joint distribution over the topic mix Θ and the set of K topics **z** is,

$$p(\Theta, z, \omega \mid \alpha, \beta) = p(\Theta \mid \alpha) \prod_{n=1}^{N} p(z_n \mid \Theta) p(\omega_n \mid z_n, \beta) \quad (1)$$

The Bayes rule says-

$$p(\Theta, z \mid \omega, \alpha, \beta) = \frac{p(\Theta, z, \omega \mid \alpha, \beta)}{p(\omega \mid \alpha, \beta)} \quad (2)$$

Therefore, LDA works in the following fashion: Considering the joint probability-

$$p(\Theta, z \mid \omega, \alpha, \beta) \propto p(\Theta, z, \omega \mid \alpha, \beta) = p(\omega \mid z, \beta) p(z \mid \Theta) p(\Theta \mid \quad (3)$$

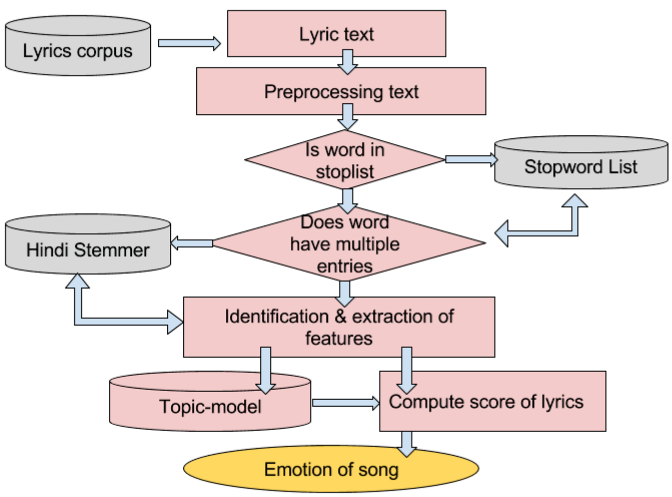

Fig. 2. Topic Model considered for song analysis

$p(\omega \mid z, \beta)$ Means the probability of few laudable words per topic will increase if is converted into a sparse matrix.
$p(z \mid \Theta)$ Means if Θ has concentrated components, the probability of topic occurring in a document will increase.
$p(\Theta \mid \alpha)$ Means using a small α will increase the probability of few relevant topics in a document i.e. what we conclude from above three points is that LDA will form separate word clusters , and assign a small number and most relevant topics possible for each document [4]. For this research work, the methodology in fig2 is considered for extracting mood labels or emotions from lyrical data of songs.

IV. Experimentation

We have used Pythons tools and libraries for performing lyrics-analysis. These include Natural Language Toolkit (NLTK), and machine learning library Scikit-learn. These provide a lot of options to conduct experiments on datasets and perform text, audio or image analysis but here we are concerned with text-part of the song.

A. Taxonomy In Use

For the purpose of this research, we consider Russell's Circumplex model of emotion. A song can display varying emotions. During collection of the corpus, it was seen that most of the songs display positive emotion such as excited, pleased, relaxed etc and also negative emotions such as afraid, frustrated, depressed etc. For simplicity, we divided this taxonomy into two classes. Class_H which represents the happy mood class consisting of all positive emotions and Class_S, which is the sad mood class consisting of all negative emotions as in fig3

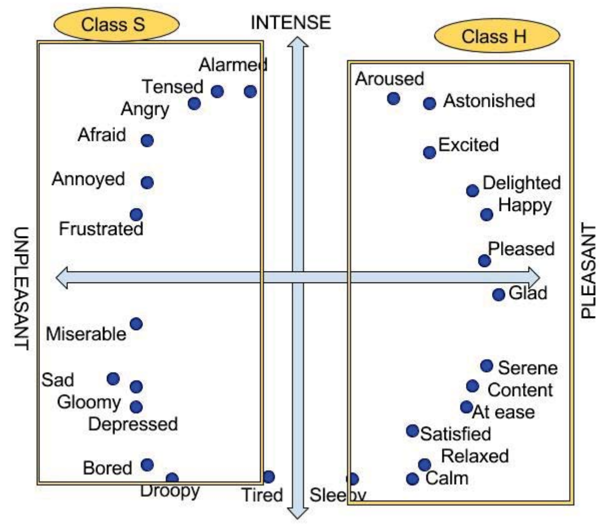

Fig. 3. Russell's Circumplex Model of Affects divided into two mood classes

B. Corpus

The major problem that we faced was the availability of a song-corpus in the Hindi language. No such data or API is readily available on the web. The following steps were followed for building a raw-dataset: The lyrics are present in HTML format and most of this information included in the underlying website was of no use to us. So we have used Beautiful-Soup to extract HTML-free data out of various websites. We then employ an automatic tool import.io, for crawling multiple links and scrape data out of these. This resulted in a table with 1910 songs. The repetitive songs were identified and removed manually. The corpus was left with a remaining total of 1894 songs. For validation purposes later after the analysis, this corpus is annotated with mood labels Happy/Sad, manually by Hindi language expert and student annotators. Lyrics of each song are read and moods are allocated accordingly. The dataset is stored in a database in the form of tables so that it can be easily retrieved whenever required in future.

C. Stop word Removal

Stop words or function words are the words that appear too frequently in a text. Because of the repeated presence of such words, they do not contribute much in text-analysis. Thus, such words can be removed without hesitation.

D. Stemming

Stemming improves the analysis by reducing morphological variants into root words. The words are evaluated and reduced due to the effect of stemmer. The simplest approach for performing stemming includes eliminating the suffix part of terms by using suffix list.

E. Tokenization

The tokens are considered to be those words that were relevant in predicting the emotions of the song. Every sentence in the lyrics of each song is divided into a list of tokens. Since we are dealing with a Hindi-corpus of songs tokenizers available in NLTK cannot identify tokens, as these are compatible for English-language. Some tools like Indic NLP are available for preprocessing and tokenization of words in Indian native languages. But we used a Hindi-language tokenizer designed in python language with the help of different modules.

F. LDA

1) Document-Term Matrix: The very first step for generating a LDA model is to find how frequently each term occurs within each document-term. So the song corpus is converted into a matrix form where rows represent songs and columns represent words and the values in the matrix represent the frequency of each word occurring in a song.

2) Fitting the model: Next step is to fit the LDA model. To achieve this we have to choose the number of topics, in our case 10, which were easy to distinguish into two mood classes.

3) Topic-word distribution: We can now use the fitted LDA model to compute topic-word distribution. From the size of the output, it was visible that each of 10 topics have a distribution over the 11250 words in the corpus.

4) Document-Topic distribution: This means that we can now get the probability distribution of each topic in each document i.e. how the topics are distributed over each song in the corpus. From the size of the output, it is quite visible that each of 1894 songs have a distribution over the 10 topics.

V. Result

We use the old school method of comparing the output with annotated corpus of songs to see LDAs performance. As seen in figure 4(a), in the annotated dataset 530 songs belong to Class_S and 1360 songs belong to Class_H. Whereas the output, fig 4(b) shows that 1530 songs are positive and 370 songs depict negative emotion.

VI. Conclusion & Future Work

In this research paper, we cover how unsupervised machine learning namely LDA has been used for performing lyrics-analysis to recognize the mood-sentiment from Hindi lyrics. Previous work in mood classification involves use of supervised machine learning approaches where music tracks are

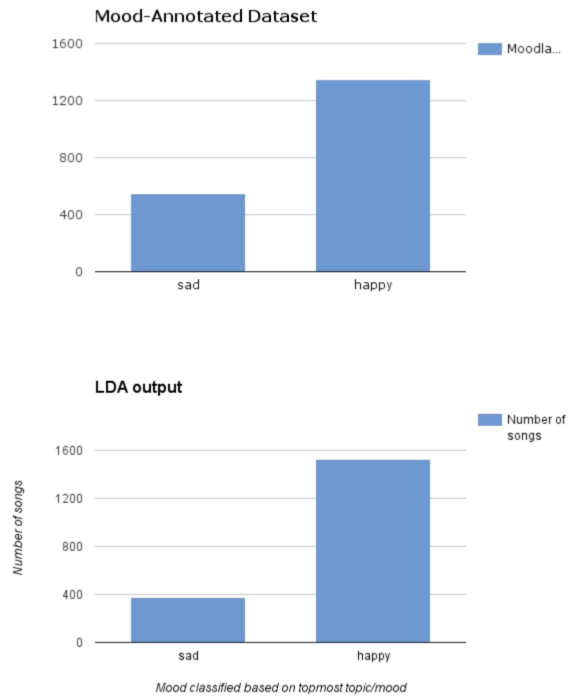

Fig. 4. (a) Bargraph for annotated dataset (b) Bargraph for LDA output of moods

classified first according to set moods and later this information is fed to discover the moods in new data. Advantage: There is no need to provide a training data. The LDA model itself discovers hidden structure of the song. Finally it is determined which topic is most prominent in song, and thus, songs were eventually categorized into Class_H or Class_S. One of the limitations noticeable in LDA is that there is no existing dataset or results to evaluate the model output. In future, evaluation schemes such as cosine similarity metrics can be used. Here taxonomy was divided into two moods for simplicity, but other mood labels can also be considered. Other techniques such as Pachinko Allocation, Hidden Markov Model can also be explored. With some more research LDA model can be used by music recommendation systems to classify Hindi songs according to Hindi lyrics. Every music system wants to expand commercially by expanding its users. The approach to classify music based on fine division of moods, either by clustering moods similar to each other, or otherwise, can be reliably predicted using the lda approach.

References

[1] A. Jamdar, J. Abraham, K. Khanna, R. Dubey, *"Emotion analysis of songs based on lyrical and audio features"*, Int. J. of Artificial Intell. Applicat. (IJAIA) Vol. 6, No. 3, 2015.

[2] B. G. Patra, D. Das, S. Bandyopadhyay, *"Mood classification of Hindi songs based on lyrics"*, 2014.

[3] B. G. Patra, D. Das, S. Bandyopadhyay, *"Automatic music mood classification of Hindi songs"*, Proc. of 3rd Workshop on Sentiment Analysis where AI meets Psychology, pp. 2428, Nagoya, Japan, Oct 14, 2013.

[4] C. Reed, *"Latent Dirichlet Allocation: Towards a deeper understanding"*, unpublished.

[5] G. Sharma, M. N. Murty, *"Mining sentiments from songs using Latent Dirichlet Allocation"* 2011

[6] Xaio Hu, *"Improving music mood social tags"*, PhD Dissertation, Univ. of Illinois, pp. 2428, Urbana-Champaign, 2010.

[7] D. J. Hu, *"Latent Dirichlet Allocation for text, Images, and Music"*, 2009.

[8] Y. Xia et al., *"Sentiment Vector Space Model for lyric-based song sentiment classification"*, 2006.

[9] C. Doig (2015) *Introduction to Topic Modeling in Python [Online]*. Available: http://chdoig.github.io/pygotham-topic -modeling.

[10] A. Ridell (2013) *Topic modeling with Latent Dirichlet Allocation using Gibbs sampling [Online]*. Available: https://github.com/ariddell/lda

[11] C. Strelioff (2014) *Getting started with LDA in Python [Online]*. Available: http://chrisstrelioff.ws/sandbox/