Inferring song moods from lyrics

Raluca-Elena PODIUC, Diana GRATIE, Octavian VOICU

Abstract. This paper has the purpose to examine and develop a classification of sentic states inferred by lyrics. Also it has the purpose to examine the relevance of the lyrics classification comparing to the mood inferred my the melody itself. The novelty of this experiment is the algorithm used to classify the lyrics using a mixture between data mining algorithms, machine learning and human experts.

Keywords: lyrics, classification, sentic state, data mining, machine learning, human expertise.

1 Introduction

There are several experiments in affective computing area that are called sentic experiences or emotion experiments. These experiments try to find out how can sentiments be identified and tagged correctly and automatically. From day to day life we know that certain emotions can be induced by music or by pictures. This was also demonstrated physiologically by the discovery of an intense activity in the limbic system during these experiences. If emotions can be induced from music what is the percentage of importance of lyrics if there is one? If the listener knows the language in which the lyrics are written than how much do they influence the emotion induced? Can a melody be tagged only by its lyrics? These are some questions that several researchers focus on.

2 Analysis of existing work

Few NLP systems have been developed for the multi-class emotion classification problem. Logan et al. [1] used latent semantic indexing of 15,589 pop lyrics by 399 artists for extracting genre. Polzin and Waibel [2] achieved 46.7% F-measure to classify 5,750 movie dialogue segments into 3 classes (neutral, angry, sad). Devillers et al. [3] reported 67.3% accuracy with 5 categories (anger, fear, satisfaction, excuse, neutral) using unigrams with stemming and compounding. Schuller et al. [4] used Bayesian belief networks to determine whether an automobile-task dialog was emotional, and if so categorized it by 6 primary emotions.

The research reported in the emotion extraction in music lyrics is by Ogihara [5]. He used lyrics for identifying clusters of 45 artists and 55 albums (such as Carly Simon, James Taylor, Joni Mitchell, Suzanne Vega etc). Accuracy of lyrics was comparable with using sound (0.635 vs. 0.685), as was precision (0.572 vs. 0.654) and recall (0.622 vs. 0.714). The F-measure for lyrics (0.602) was comparable to that of sound (0.669).

3 Proposed approach

The novelty of our experiment is the mixture between several models of approaching this matter. As we mentioned before the option we have in mind is to develop a system that has a supervised learning behavior. The classification is done by using data mining algorithms and also can be done by human factor.

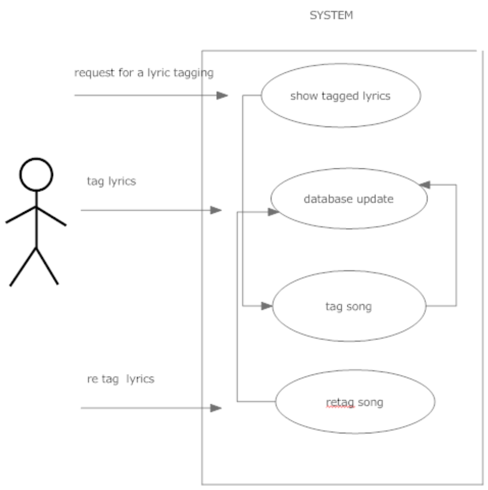

Figure 1: Use case diagram

Lyrical text is distinct from ordinary text in the use of stylistic qualities such as rhyme, poetic form, and figurative language. Song lyrics help to focus the listener's attention on specific emotions. Psychologists have interpreted the affective value of words, based upon empirical surveys and expert judgments. Measurement scales were created to quantify the verbal reports of psychological state according to how many and which dimensions (e.g. intensity, valence, and dominance). A variety of ratings scales for affective words were developed, and documents were rated by summing the ratings of individual words.

Our application tags songs by lyrics using the following steps.

First of all we have to create a corpus of tagged lyrics. In order to tag these lyrics we have to parse the text for each keyword. Each keyword in the training set will be tagged as positive or negative by the number of appearances of that word in positive or negative tagged songs. There are several classifiers for words, prepositions, conjunctions and other links in phrase that are considered to be forever disposable. Words like nouns could or may not have values in the context of our applications. For example, the word “car” can be considered never having a relevant importance because cannot be associated with a mood.

Second step, the tagging of the test data, is done following the next steps.

Parsing step
The parsing is done following the next criteria:
First of all one key word is more like a sequence of keywords because there are certain ads that can change the polarity of it. These words are called modifiers.

Refrain

Every refrain is parsed and the keywords found in it are considered to be the main keywords of the song. These words should have a polarity determined in the first step of the algorithm. The associated polarity will have a greatest importance of all tagged words. Also repeated words have a great importance.

Third step - KNN clustering

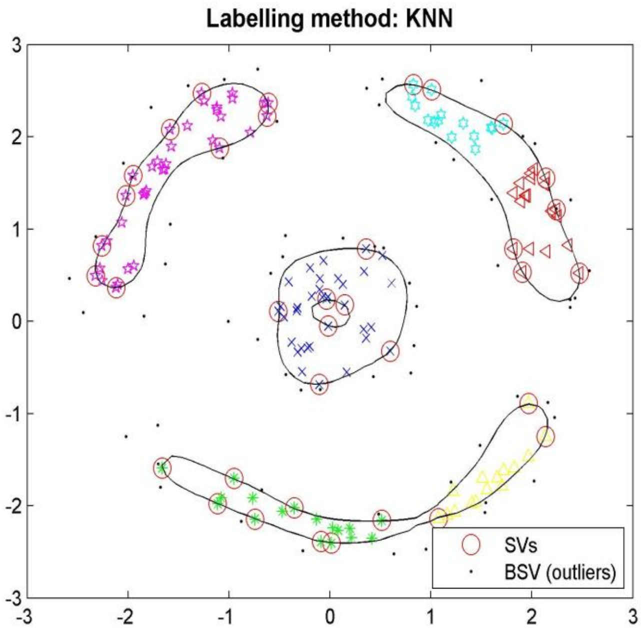

Figure 2: KNN clustering

The *k*-nearest neighbor's algorithm (*k*-NN) is a method for classifying objects based on closest training examples in the feature space. *k*-NN is a type of instance-based learning, or lazy learning where the function is only approximated locally and all computation is deferred until classification. The *k*-nearest neighbor algorithm is amongst the simplest of all machine learning algorithms: an object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its *k*-nearest neighbors (k is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of its nearest neighbor.

The same method can be used for regression, by simply assigning the property value for the object to be the average of the values of its k nearest neighbors. It can be useful to weight the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. (A common weighting scheme is to give each neighbor a weight of $1/d$, where d is the distance to the neighbor. This scheme is a generalization of linear interpolation.)The neighbors are taken from a set of objects for which the correct classification (or, in the case of regression, the value of the property) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.

In this application test the centroids of the cluster are considered to be the keywords with the most pronounced positive or negative polarity. Every keyword is grouped in one of the two clusters. The one with the largest magnitude would be the one that tags the song.

Special cases

There are certain cases when our algorithm can only partially decide, like in the case represented bellow:

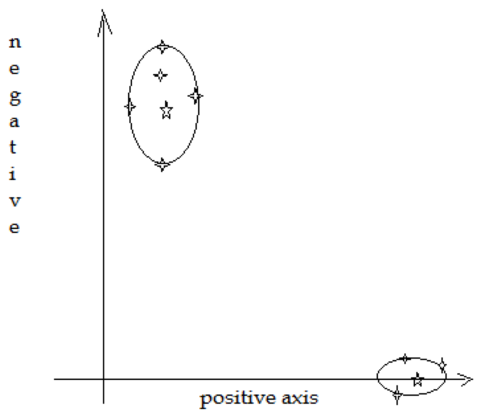

Figure 3: Pathological case

In these case there can be considered two cases: the cluster that is the nearest to the axis or the one that is larger. Each one of the cases can be considered correct. If the tagging fails the user can adjust it by retagging.

4 Software Architecture

The software architecture is the simplest we could think of. First of all we have chosen the safest but also slowest language nowadays, Java. Also we consider that portability should be very important for this application so Java has another plus. Next to it there are several structures of data in java that match our application. Our intention is also to integrate an R language module but this is just a proposal. In the first release of this application we will consider only java modules.

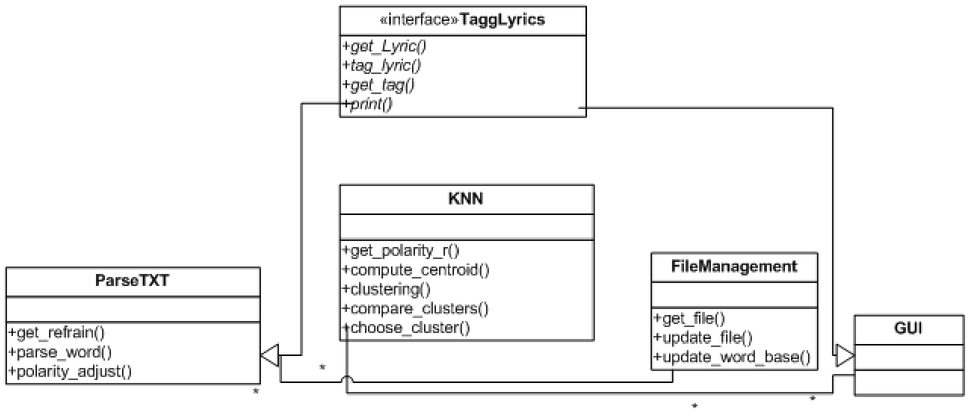

Figure 4: Class diagram

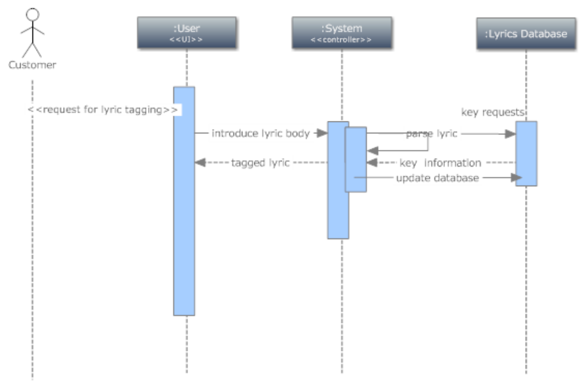

Figure 5: Sequence diagram

For better understanding of the work flow we consider also a sequence diagram shown above.

The corpus is stored in independent txt files. At each parsing step the output of the parsing is stored to a central file called word_base.txt that is structured following the bellow layer.

Key word	Positive polarity	Negative polarity	Tagged songs

The test phase is constructed using the same parsing class but with an adjustment. The importance of two aspects are underlined here and implemented if applicable. First of all a great importance is given to the refrain. We consider that, if the refrain exists than these contain the main idea of the song and the mood should be induced according to it. In this case an extra weight is applied to the keywords by adding an extra polarity to it.

Another very important aspect is the words found in front of the key word. These can change the chosen polarity. If for example we have a negation the polarity taken in consideration would be the negative one from the word base. If also an augmentation is found than the polarity will get an extra weight by doubling it.

Next step is to choose the centroids of two clusters representing the negative percentage and the positive one. These will be chosen from the refrain preferably but also, if not found in it can be found in the corpus of the lyric.

In the class KNN all the wards that are parsed and are considered to be relevant are grouped in the two clusters. We obtain two different clusters from which we consider the most polar one if the magnitude of it is greater than the other or if the two are almost of same polarity ratio we consider the one with the larger density. Examples are shown bellow.

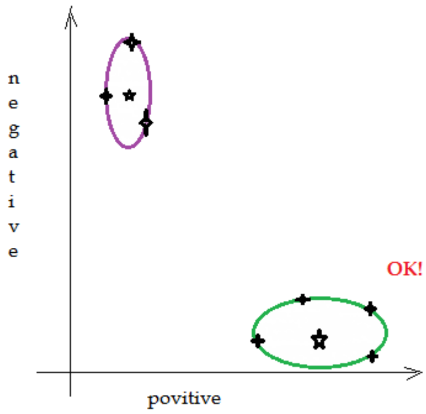

Figure 6: First case o clustering

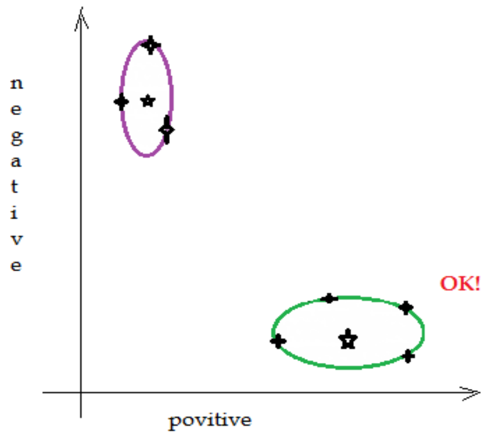

Figure 7: Second case o clustering

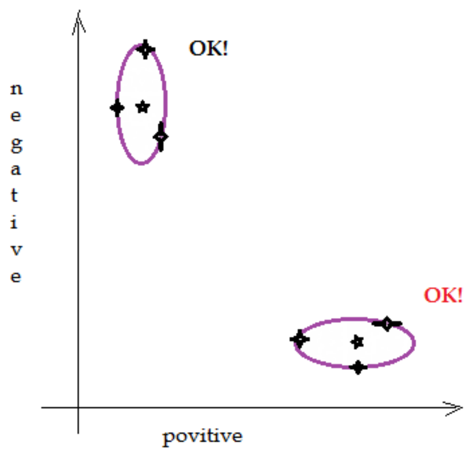

Figure 8: Third case o clustering (pathological case)

Remark 1. In the above case the melody is tagged according to the gender polarity established by the user or by the training set.

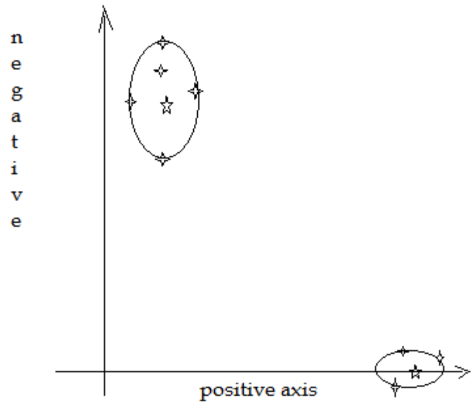

Figure 9: Another pathological case

Remark 2. In the case shown in the above clustering diagram it cannot be decided. By implementation we take the greatest polarity but we consider that the correction, if necessary wil be given by human expertise.

3 Results and Future work

We expect in the worst case to obtain results at least as good as the result described in the paper "Sentiment Vector Space Model for Lyric-based Song Sentiment Classification"[] that is an accuracy of 0.78 % but much efficient than in that case because our algorithm has a greater performance/cost ratio comparing to the solutions proposed until now.

In the future this experiment will have a greater accuracy by mood classification according to Russel's model of mood. Also the application will be supported on web in order to become a collaborative application.

A very interesting comparison that this project has as goal is to compare the accuracy of the melody compared to the lyrics.

4. References

[1]B. Logan, D. P.W. Ellis, and A. Berenzweig. Toward evaluation techniques for music similarity. In Proceedings of The 4th International Conference on Music Information Retrieval, pages 81–85, 2003.
[2]POLZIN, T. AND WAIBEL, A. 2000. Emotion-sensitive human-computer interfaces. In Proceedings of the ISCA-Workshop on Speech and Emotion.
[3]DEVILLERS, L., VASILESCU, I., AND LAMEL, L. 2002. Annotation and detection of emotion in a task-oriented human-human dialog corpus. In Proceedings of ISLE Workshop.
[4]SCHULLER, B., RIGOLL, G., AND LANG, M. 2004.Speech emotion recognition combining acoustic features and linguistic information in a hybrid support vector machine –belief network architecture. IEEE International Conference on Speech, Acoustics, and Signal Processing 2004.
[5]OGIHARA, M. 2004. Data mining for studying large databases: Course presentation Onlinelocation: www.cs.rochester.edu/u/scott/200/2004-04-08_ogihara_slides.pdf.
[6]Alena Neviarouskaya, Helmut Prendinger, and Mitsuru Ishizuka – "EmoHeart: Conveying Emotions in Second Life Based on Affect Sensing from Text"
[7] http://dusanwriter.com/index.php/2010/04/10/ifeel_im-robot-adds-sensations-to-second-life/
[8] Valery A. Petrushin - Emotion recognition in speech signal: experimentaly study, development, and application
[9] Lyric-based song emotion detection with affective lexicon and fuzzy clustering method -Yajie Hu, Xiaoou Chen and Deshun Yang Peking University
[10] Music Emotion Identification from Lyrics Won-Sook Lee School of Information Technology and Engineering University of Ottawa