Song classification

Nadia Lachetar[1]
[1] Computer Science department,
University 20 aout 1955 Skikda
Skikda, Algeria;
Email nadia_ishak2002@yahoo.fr

Halima Bahi [2]
[2] LabGED Laboratory, Computer Science department,
University Badji Mokhtar Annaba
Annaba, Algeria;
Email bahi@labged.net

***Abstract*—We live in a world full of data. Every day, people handle different types of data from a variety of measurements and observations. Data describes the characteristics of a species of life, depicts the properties of a natural phenomenon; summarizes the results of experiences or records the dynamics of system operation. Thus, the data provides a basis for analysis, reasoning, decision and ultimately, for understanding all kinds of objects and phenomena. The new information technology, at the same time they facilitate the movement and storage of information, contributing to their exponential growth, they are in numbers disproportionate to the human resources to address them. This paper addresses the problem of indexing database containing songs using title and artist particularities to enable their effective exploitation. In this article, we present our experiments in automated song categorization, where we suggest the use of an ant colony algorithm. A Naive Bayes algorithm is used as a baseline in our tests.**

Keywords-componen: Song categorisation; Information Retrieval, Audio indexing; Naive Bayes Algorithm; Ant Colony Algorithm.

I. INTRODUCTION

Because of the difficulty of locating songs in audio archives, a large amount of music will be fallen into oblivion [1]. It is now possible to start building automatic content-based indexing and retrieval tools, which, in time, will make song recordings as valuable as text has been as an archival resource. This paper describes song classification; we propose to construct an automatic system of categorization of songs by theme using metadata. Currently many studies are devoted to the use of acoustic information to detect the theme of songs [2, 3, 4]. Many studies also have been performed to detect the music, speech, or sound features [5, 6]. Very little has been done on the song [7]. Song title briefly describes its theme so we use the title to classify a song. We use also the characteristics of the artist, because every artist has a tendency to compose or sing a particular kind of music. To detect the song theme we use metadata. An ant colony algorithm is proposed for classification of songs.

Section 2 presents the state of the art. Section 3 describes our training data. Section 4 is devoted to categorize songs while section 5 summarizes the naive Bayes algorithm. Section 6 details our approach for indexing songs using the songs title and artist features. In section 7 the obtained results are given, a conclusion is drawn on the performance of the ant colony algorithm and future work is discussed.

II. STATE OF THE ART

Many works are devoted to the extraction of features of a song for the descriptions of its contents. They are generally guided by the acoustic analysis [2, 3, 4]. Knees and others have a pioneering work to build an automatic search that is able to find the music that satisfies arbitrary queries in natural language [8]. Another work described in [9] is based on an automatic segmentation of the soundtrack music or speech, using a technique of segmentation into sentences. The music segments are indexed in a way that allows a search by similarity.

Other jobs using a classification according to the mood of the songs are described in [1, 10, 11]. The classification according to the mood does not seem interesting to apply it to a search engine for music because the mood is subjective metadata where words are short and contain many metaphors that can be understood by humans. Through this work, we introduce a new dimension of classification, considering contextual information about the artist. Thus, each artist sings songs with a specific emotion, thus Eric Clapton often sings sad songs while Bob Marley likes happy songs.

III. CONSTRUCTION OF TRAINING DATA

A great blog site Live Journal (www.livejournal.com) is used, each blog entry is labelled with the theme of the song given by the title of this latter. The song title and artist features can be obtained by simple string matching with the database artist, obtained from open artist got from the music site (www.musicmoz.org). The lyrics may be obtained from (www.lyrics.org).

IV. SONG CLASSIFICATION

Research in the field of automatic categorization remains relevant today since the results are still subject to improvements. For some tasks, the automatic classifiers perform almost as well as humans, but for others the gap is even greater. At first glance, the main problem is easy to grasp. On one hand, we are dealing with a bank of songs and on the other with a set of categories. The goal is to make a computer application which can determine to which category belongs a song based on its contents.

The purpose of the automatic song categorization is to learn a machine to classify a song into the correct category based on its content; the categories refer to the topics (subjects).

We may wish that the same song is associated with only one category or it can belong to number of categories. The set of categories is determined in advance. The problem is to group the songs by their similarity.

A. There are two approaches to solve the problem of songs categorization: the information using either acoustic or verbal information. In this paper we focus on the words in the song's title to determine its theme and the characteristics of the artist to determine the kind of music. How to categorize a song?

The categorization process includes the construction of a prediction model that receives in input the title of the song, and as output it combines one or more labels. To identify the category associated to a song, the following steps are required:

Learning includes several steps and leads to a prediction model.

We have a set of labeled songs (for every song we know its class).

From this corpus, we extract the k descriptors (t_1,.., t_k) which are most relevant in the sense of the problem solving.

We have then a table of "descriptors x individuals" and for every word of the title of the songs we know the value of descriptors and its label.

The classification of words of title for a new song d_x includes two stages

Research and weighting the instances t_1, .. t_k of terms in title of song to classify d_x[12]. Note that the k most relevant individuals (t_1, ..., t_k) are extracted during the first phase by analyzing the titles of songs of the training corpus. In the second phase, the classification of a new song, we simply seek the frequency of these k descriptors (t_1, ..., t_k) in the title of song to be classified.

B. Representation and coding of a song

Prior coding of song is necessary because there is currently no method of learning which can directly handle unstructured data in the model construction stage, or when used in classification.

For most learning methods, we must convert all texts in a PivotTable "individuals-variables".

Individual is a song d_j, labeled during the learning stage, it will be classified in the prediction phase.

Variables are descriptors (terms) t_k which are extracted from data of the song.

The contents of table w_{kj} represent the weight of term k in title of song j.

Different methods are proposed for the selection of descriptors and weights associated with these descriptors. Some researchers use the words as descriptors, while others prefer to use the lemmas (lexical roots) or even Stemme (deletion of affix) [12].

C. Approaches for songs representation

Learning algorithms are not able to treat texts and more generally unstructured data such as images, sounds and video clips. Therefore a preliminary step called representation is required. This step aims to represent each song by a vector whose components are such words of title in the song to make it usable by the learning algorithms. A collection of songs can be represented by a matrix whose columns are the songs [12].

Many researchers have chosen to use a vector representation in which each song is represented by a vector of *n* weighted terms. The *n* terms are simply the *n* different words of titles in the songs.

In song categorization, we transform the title of the song into a vector $d_j = d_j\ (w_{1j}, w_{2j}, ..., w_{|T|j})$, where T is the set of terms (descriptors) that appear at least once in the corpus (the collection) learning. The weight w_{kj} correspond to the contribution of terms t_k to the semantics of title of song d_j [1].

Once we choose the components of the vector representing the song j, we must decide how to encode each coordinate of the vector d_j. There are different methods to calculate the weight w_{kj}. These methods are based on two observations:

More the term t_k is frequently in a title of song d_j, more it is relevant to the subject of this song.

More often the term t_k is in a collection, unless it is used as discriminating between songs.

The Coding terms frequency x inverse document frequency and Coding terms TFC are the most used.

We add to the vector of features once standardized the characteristics of the author (the type of music and the theme he sings in general).

V. NAÏVE BAYES ALGORITHME

In machine learning, different types of classifiers have been developed to achieve maximum degree of precision and efficiency, each with its advantages and disadvantages. But, they share common characteristics [13].

Among the learning algorithms we cite: Naive Bayes which is the most known algorithm, Rocchio method, neural network, method of k nearest neighbors, decision trees and support vector method [13].

Naive Bayes Classifier is the most commonly used algorithm, this classifier based on Bayes theorem for calculating conditional probabilities. In a general context, this theorem provides a way to calculate the conditional probability of a case knowing the presence of an effect.

When we apply the naïve Bayes for a song categorization task, we look for the classification that maximizes the probability of observing the words of titles of the songs.

During the training phase, the classifier calculates the probability that a new song belongs to this category based on the proportion of training songs belonging to this category. It calculates the probability that a given word is present in a title of the song, knowing that this song belongs to this category.

Then as a new song should be classified, we calculate the probability that it belongs to each class using Bayes rule and the probabilities calculated in the previous step.

The likelihood to be estimated is:

$$p(c_j|a_1,a_2, a_3, ..., a_n) \quad (1)$$

Where c_j is a category and a_i is an attribute

Using the Bayes theorem, we obtain:

$$p(c_j|a_1, a_2, .., an) = \frac{P(a_1,a_2,...,a_n\backslash c_j)*P(c_j)}{P(a_1,a_2,...,a_n)} \quad (2)$$

$$p(a_1, a_2, a_3, ..., a_n\backslash c_j) = \prod_{i=1}^{n} p(a_i\backslash c_j) \quad (3)$$

To estimate the probability $P(a_i\backslash c_j)$, we could calculate directly in the titles of songs driving the proportion of those belonging to class c_j that contain the word a_i.

In the extreme case where a word is not met in a class its probability of 0 dominates the others in the above product and would void the overall probability.

To overcome this problem, a good way is to use the m-estimate calculated as well.

$$\frac{nk + 1}{n + |vocabulairy|} \quad (4)$$

Where,

n_k is the number of occurrences of the word in class c_j

n is the total count of words in the training corpus.

|Vocabulary|: the number of keywords.

VI. Application of ant colony algorithm for song categorization

A. *Introduction*

The originality of our approach is on adapting an algorithm of ant colony to song categorization.

The algorithm of ant colony optimization is inspired by the behavior of ants searching for food. Its principle is based on the behavior of individual ants; they are able to determine the shortest path between their nest and a food source using the pheromone which is a substance that ants lay on the floor when they move. When an ant has to choose between two directions, it chooses with higher probability [14].

B. *Principe of the algorithm*

It relies on the specific behavior of ants, and determines the shortest path between the nest and a food source overall progress algorithm.

1) *Iteration and moving of an ant*

Iteration corresponds to the movement of ants. To get from one node of the graph to another, each ant will need a number of iterations depending on the size of the edge to go. This mode of iteration will also emphasize the shortest path as the ants will need less iteration to reach the end.

2) *Life of ant*

Each ant must know the list of nodes it has visited and the nodes still to go. In addition it must measure the time she spends in exploring the solution. At each node the ant will consider the possible edges in observing their corresponding levels of pheromone. It has only to choose randomly, favouring arcs strongly pheromone. Once at destination, the ant knows the total length of the solution, it can reconstruct the path in reverse to mark the path with its pheromones and increases the collective knowledge of the colony.

3) *Deposition of pheromone*

The pheromone is a substance that ants lay on the floor when they move. The heuristic pheromone deposition can significantly change how the convergence of the algorithm. From a naive point of view, we can completely remove the same amount of pheromone on each path. Ants engaged in long paths will filed less than pheromones because they can try fewer paths. And instead engaged ants on the shortest paths will soon try other paths. In the shortest paths will be found more pheromone than others. We may also use other methods of depositing pheromones. An interesting idea is to deposit more pheromone as the solution is good.

C. *For song categorization*

For the construction of the graph, the nodes represent titles of songs. The pheromone is a measure of similarity between titles of songs which may be the distance between these documents. The choice of distance is an important parameter.

1) *Calculate the distance between a given song and songs constituting the graph*

For our approach we use the cosine similarity between two songs a and b defined by

$$\sum_{t\in T} \frac{p_t(a) * p_t(b)}{\sqrt{\sum_{t\in T} p_t(a)^2 * \sum_{t\in T} p_t(b)^2}}$$

Where:

T is the set of attributes.

$p_t(a)$ is the weight of term t in title of song a.

$p_t(b)$ is the weight of term t in title of song b.

This measure allows comparing titles of songs of different lengths by normalizing their vector.

We use the cosine similarity between each title of song “a” of the graph of songs and the input title of song “b” to be classified.

The following algorithm computes the cosine similarity based on relevant attributes for the various couples forming the nodes of a graph and the input title of song. It takes as input the graph of songs and the document to classify and returns as output a similarity matrix based on relevant attributes.

```
Algorithm Cosine_Similarity
Begin
    Input:song_Graph,song_class //graph of
    songs,Classified song;
    Output: Mat_Sim
    // similarity matrix based on the relevant
    attributes
    Mat_Sim ← 0;
    Begin
    For each node of song_Graph
    // Extract set of attribute nodes of the graph
    SIM=Calcul_Sim (node, song_class);
    Mat_Sim=Mat_Sim+Sim(node,song_class);
    Return Mat_Sim
    End.
End.
```

Figure1. **Cosine_Similarity**

2) Ant colony optimization

To find the song category, we adopt the algorithm of ant colony optimization (ACO), proposed in [14]. Although the ant colony algorithm is originally designed for the travelling salesman problem, it finally offers great flexibility. Our choice is motivated by the flexibility of the metaheuristics which makes possible its application to different problems that are common to be NP-hard. Thus the use of a parallel model (colonies of ants) reduces the computing time and improves the quality of solutions for categorization.

Formalization of the problem: In our context, the problem of classifying a song reduces the problem of subset selection [14], and we can formalize the pair (S, f) such that:

S contains all the cosine similarities calculated between the graph of songs and the song to classify. It's "matrix similarity" mat_sim.

F is defined by the function score, the score function is defined in [14] by the formula.

$$score(s') = f(\text{song\_Graph} \cap \text{song\_class} - g(slits\ s')). \quad (6)$$

Splits (S') is the set of nodes in the graph which are more similar to the song to classify. So the result is a consistent subset S' of nodes, as the score function is maximized

3) Construction of the graph of songs

To adapt our approach, a direct graph G(V,E) is drawn, where V is a set of vertices and E is a set of possible edges between these vertices as shown in figure 1 in this graph a number of ants is managed for tmax iteration.

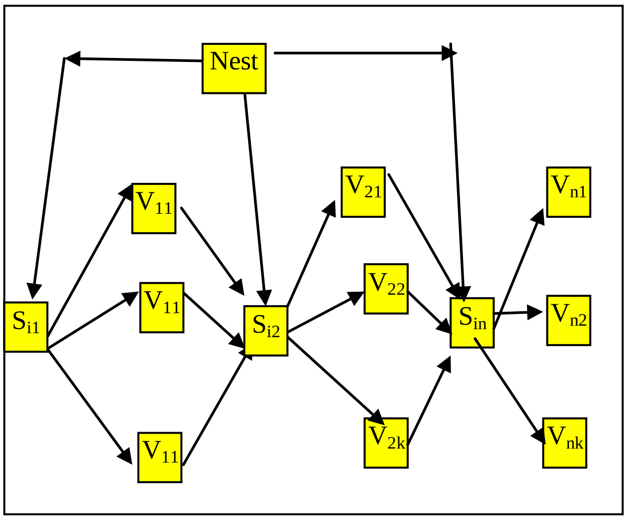

Figure 2. **Representation of song categorization**

A graph G(V,E) is automatically generated from a text file that contains the problem's data.

4) Description of the algorithm

At each cycle of the algorithm, each ant constructs a subset. Starting from empty subset, ants at each iteration add a couple of nodes from the similarity matrix. S_k chosen among all couples not yet selected. The pair of nodes to add to S_k is chosen with a probability which depends on the trail of pheromones and heuristics. One aims to encourage couples who have the greatest similarity and the other is to encourage couples who are most increase the score function. Once each ant has built its subset, a local search procedure start to improve the quality of the best subset found during this cycle. Pheromone trails are subsequently updated based on the subset improved. Ants stop their construction when all pairs of candidate nodes are decreased the score subset or when the three latest additions failed to increase scoring.

Construction of a solution by an ant: The following code describes the procedure followed by ants to construct a subset. The first object is selected randomly.

Algorithm Construction_subset
Begin
Input:SS_problem(S,$s_{consistant}$,f)and an associated heuristic function: S*P(S)$\rightarrow$ IR^+; heuristic pheromone; and an phenomenal factor δ and 2 heuristic factors φ_1 and φ_2;
Numeric parameter α, β_1 and β_2
Output: a consistent subset S'$\in$ S
Initialize pheromone trails to τmax
Repeat
For each ant k in 1 .. nbAnts, construct a solution S_k as follows:
1. Randomly select the first node $O_i \in S$
2. $S_k \leftarrow \{o_i\}$
3. Candidat $\leftarrow \{o_i \in S / S_k \cup \{oj\} \in S_{consistent}\}$
4. While Candidates $\neq \emptyset$ do
5. Choose a node $o_i \in$ candidat with probability
6. $p_{oi} = \sum_{t\in T} \frac{p_t(a)*p_t(b)}{\sqrt{\sum_{t\in T} p_t(a)^2 * \sum_{t\in T} p_t(b)^2}}$
song_graph.
// p_t (b) is the weight of term t in song to be classified
7. $s_k \leftarrow S_k$ union $\{o_i\}$
8. Remove o_i from Candidates
9. Remove from candidates each node o_j as $S_k \cup \{o_j\} \in S_{consisting}$
10. End while
11. End for
Update pheromone trails according to $\{S_1, ..., S_{nbAnts}\}$
If a pheromone trail is less than τ_{min} then set it to τ_{min}
Else If a pheromone trail is greater than τ_{max} then set it to τ_{max}
Until maximum number of cycles reached or solution found.

VII. Results , Conclusion and Future Work

To evaluate performances of our suggestion, we make some experiments using two corpus one for the training and the other for the test. We also use the Naïve Bayes classifier as baseline one.

Table I. Classes of corpus

Class	Nb of songs in training stage	Nb of songs in test stage
national anthem	12	10
Loves songs	30	26
Religious songs	13	12
Sport songs	4	4
Learning songs	8	6

The results of classification stage are reported below for ant colony algorithm and naïve Bayes algorithm.

Table II. Results of tests with ant colony algorithm

Clas	Nati Ant	Lov son	Reli Son	Spor son	lear son	Total
Nati Ant	9	0	1	0	0	10
Lov Son	0	24	1	1	0	26
Reli Son	0	3	8	0	1	12
Spor Son	0	1	0	3	0	04
lea son	0	1	0	0	5	06

Table III. Results of tests with naïf bayes algorithm

Clas	Nati Ant	Lov son	Reli son	Spor son	lear son	total
Nati Ant	7	2	1	0	0	10
Lov Son	0	22	1	1	2	26
Reli Son	1	3	7	0	1	12
Spor Son	0	1	0	3	0	04
lea son	1	1	0	0	4	06

Precision and recall are the most used measurements to evaluate information retrieval systems, they are defined as follow:

Table IV. Contingency table based evaluation of the classifiers

	Song belonging to the category	Song not belonging to the category
Song assigned to the class by the classifier	A	b
Song rejected by the classifier	C	d

According to this table, we define:

Precision=a/(a+b), the number of correct assignments over the total number of assignments.

Recall=a/(a+c), the number of correct assignments over the number of assignments that should have been made.

When evaluating the performance of a classifier, precision or recall is not considered separately. So the F1 measure is defined which is used extensively by the formula:

F1 = 2*r*p/(p+ r) (r is the recall, and p is the precision).

It is a function which is maximized when the recall and precision are close.

Table 5 and table 6 present performances of ant colony and naïve Bayes in terms of recall, precision and F1

Table V. Recall, precision, F1 for each class (ant colony algorithm)

Class	Recall	Precision	F1
national anthem	90.00	100.00	94.73
Loves songs	92.30	82.75	87.26
Religious songs	66.66	80.00	72.72
Sport songs	75.00	100.00	85.71
Learning songs	83.33	71.42	73.68

Table VI. Recall, precision, F1 for each class (naïve bayes algorithm)

Class	Recall	Precision	F1
national anthem	70.00	77.77	73.68
Loves songs	84.61	75.86	79.99
Religious songs	58.33	77.77	66.66
Sport songs	75.00	75.00	75.00
Learning songs	66.66	57.14	61.53

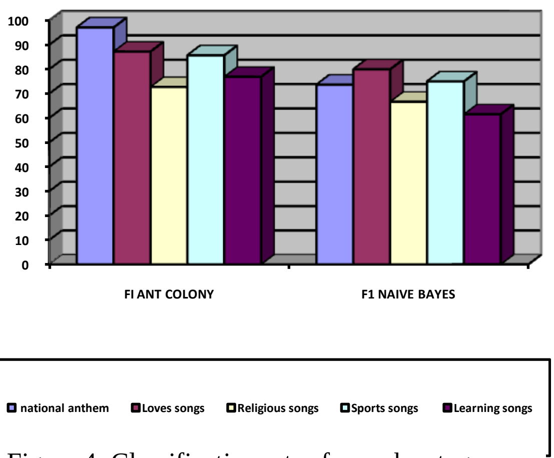

Figure 4: Classification rates for each category and both classifiers

Conclusion

An ant colony algorithm for song classification using metadata has been described; it offers an efficient tool for categorization. The results show that the suggested ant colony algorithm is better than Naïve Bayes algorithm in terms of recall and precision because ant colony algorithm handle graphical representation (characteristics of artist are taken into account during the construction of the graph).

Since performing better categorization need more contextual information, we propose to use the chorus which is a refrain of a song: a verse repeats at least twice with none or little difference between repetitions and tend to explicitly show the main theme of the song.

REFERENCES

[1] Dang, T. T., Shirai, K., 2009. Machine learning approaches for mood classification of songs toward music search engine.

[2] Li, T., Ogihana, M., 2004. Content-based music similarity search and emotion detection.

[3] Chua, B. Y., Lu, G., 2004. Determination of perceptual tempo of music. *In CMMR.*

[4] Liu, D., Lu, L., Zang, H.J., 2003. Automatic mood detection from acoustic music data.

[5] Scheirer, E., Slaney, M., 1997. Construction and evaluation of a robust multifeatures speech/music discriminator. *In ICASSP. IEEE*, vol.2, pp. 1331-1334.

[6] Karjalainen, M., Tolonen, T., 1999. Multi-pitch and periodicity analysis model for sound separation and auditory scene analysis. *In ICASSP.IEEE*, vol. 2, pp. 929-932.

[7] Arroabarrem, M. Zivanovic, X., Rodet, Carlosena, A. 2003. Instantaneous frequency and amplitude of vibrato in singing voice. *In ICASSP.IEEE*, vol. 5, pp. 537-540.

[8] Knees, P., Pohle, T., Schedl, M., Widmer, G., 2000. A music search engine Built upon Audio-based and Web-based similarity Measures. *In SIGIR.*

[9] Harb, H., Chen, T., 2003. Cyndi: an indexing engine from the soundtrack by segmentation and semantic extarction keywords. LIRIS, Lyon

[10] Kaners. PW.M., 2009. Automatic mood classification for music.

[11] Laurier, C., Grivolla, J., Herrera, P., 2008. Multimodal music mood classification using audio and lyrics. *ICMALA, pp 668-693*

[12] Jalame, R., 2003. Machine learning and multilingual tex classification. Lyon 2 university.

[13] Sebastiani, F., 2002. Automated text categorization tools, techniques and application. Renne France,.

[14] Solnon, C., 2005. Contribution to the practical problem-solving combinatorial-graph and the ants. Thesis for habilitation research, Claude Bernard University Lion.