

APPLICATION OF GENETIC ALGORITHM FOR AUDIO SEARCH WITH RECOMMENDER SYSTEM

Sachin Bojewar and Jaya Fulekar

Vidyalankar Institute of technology, Mumbai University, Mumbai, India

*Corresponding author: Email: jaya_fulekar@yahoo.co.in, Tel: +022-28681811

[Received-26/03/2012, Accepted- 12/05/2012]

ABSTRACT:

With growing popularity of the web-based systems that are applied in many different areas, they tend to deliver customized information for their users by means of utilization of recommendation methods. The recommendation techniques: content-based, collaborative and by means of genetic algorithm are used in existing systems. They recommend user those items which are based on past history of the user but user's intention is not focused. This paper proposes Genetic algorithm for recommendation by generating new mutations by providing optimized solution every time and which is based on user's preferences.

Keywords: Recommender system, Genetic algorithm, User Profile

[I] INTRODUCTION

When users browse through a web site they are usually looking for items they find interesting. Interest items can consist of a number of things. For example, textual information can be considered as interest items or an index on a certain topic could be the item a user is looking for. Another example, applicable for a web vendor, is to consider purchased products as interest items. Whatever the items consist of, a website can be seen as a collection of these interest items.

Recommender systems are widely implemented in e-commerce websites to assist customers in finding the items they need. A recommender system should also be able to provide users with

useful information about the item that interest them. The ability of promptly responding to the changes in user's preference is a valuable asset for such systems.

A Recommender system for music data is proposed which assists customers in searching music data and provides result with items resulting in own user preference. This system first extracts unique properties of music like pitch, chord, and tempo from the music file using a CLAM annotator software tool. This extracted data is then stored on the database. [4]

Each stored property is analyzed using content based filtering and interactive genetic algorithm. After acquiring records, the system recommends items appropriate to user's own favorite.

The following sections describes genetic algorithm, the flow of recommendation system for audio search.

[II] GENETIC ALGORITHM

2.1. Introduction of GA

The evolutionary process of a GA is a highly simplified and stylized simulation of the biological version. It starts from a population of individuals randomly generated according to some probability distribution, usually uniform and updates this population in steps called generations. Each generation, multiple individuals are randomly selected from the current population based upon some application of fitness, bred using crossover, and modified through mutation to form a new population.

- Crossover – exchange of genetic material (substrings) denoting rules, structural components, features of a machine learning, search, or optimization problem.
- Selection – the application of the fitness criterion to choose which individuals from a population will go on to reproduce.
- Replication – the propagation of individuals from one generation to the next
- Mutation – the modification of chromosomes

2.2. Genetic algorithm in recommender system

A Recommender system for music data is proposed which assists customers in searching audio data and provides result with items resulting in own user preference. This system first extracts unique properties of music like pitch, chord, and tempo from the music file using a CLAM annotator software tool. This extracted data is then stored on the database.

Each stored property is analyzed using content based filtering and interactive genetic algorithm. After acquiring records, the system recommends items appropriate to user's own favorite.

The genetic algorithm is applied and user is provided with a general list from which users can select the audio tracks, listen to it and give rating, recommendation list and user favorite list where

that user has given highest rating to the audio tracks.

[III] PHASES OF GENETIC ALGORITHM IN RECOMMENDER SYSTEM

3.1. Phases

The following are phases of genetic algorithm are as follows:

1. Selection phase – Music features are extracted using CLAM software. The items which are rated above threshold value are selected rest other items are ignored.
2. Crossover phase – The BLX – alpha crossover algorithm is used since extracted features are real numbers. Hence crossover is performed with this algorithm resulting in new generation.

BLX-a

1. Select two parents $X(t)$ and $Y(t)$ from a parent pool
2. Create two offspring $X(t+1)$ and $Y(t+1)$ as follows:
 3. for $i = 1$ to n do
 4. $d_i = |x_i(t) - y_i(t)|$
 5. Choose a uniform random real number u from interval $\langle \min(x_i(t), y_i(t)) - a d_i, \max(x_i(t), y_i(t)) + a d_i \rangle$
 6. $x_i(t+1) = u$
 7. Choose a uniform random real number u from interval $\langle \min(x_i(t), y_i(t)) - a d_i, \max(x_i(t), y_i(t)) + a d_i \rangle$
 8. $y_i(t+1) = u$
 9. end do

where:

a – positive real parameter

3. Matching phase- This phase finds the similar items stored in database to the newly generated music features. Once similarity is found those items are recommended to the user.

This phase uses Euclidean distance between two offspring and distance between each feature of the two offspring is calculated, resulting value is used to match the records stored in the database.

Those records are compared with the resulting value which the user has given highest rating to the tracks.

3.2. Flow of the system

The figure 1 explains the flow of the system which can be designed and implemented. The idea is to provide each user with user profile for audio tracks. Each of the records from the data base is displayed on user page and called as general list and indexing can be provided to make it more user friendly. The CLAM software then extracts the features from each of the files which the user has provided rating, the features extracted can be tempo, pitch, etc. These extracted features are stored on to the database. The genetic algorithm is used further to provide optimized solutions for recommending items and recommended list will be always provide newer solutions one which are closer to user's preference.

The genetic algorithm can be made to run infinite times to generate the mutations, hence every time providing new solutions that is, new recommendation every time.

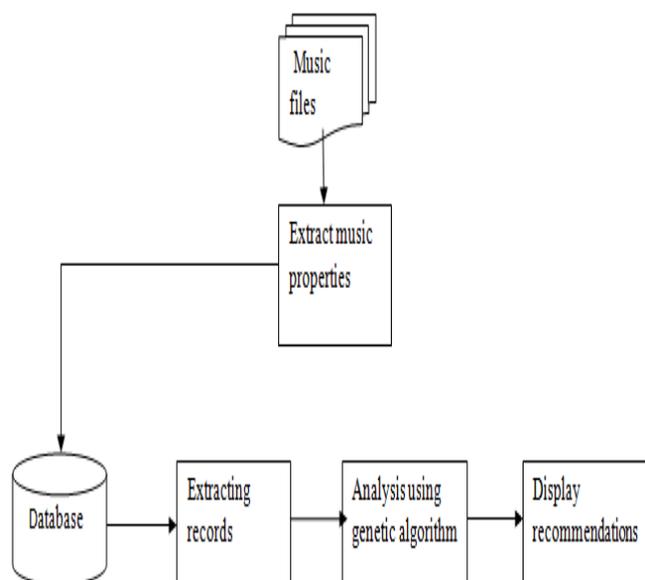


Fig 1. The flow of recommender system

[V] CONCLUSION

In this paper, we proposed real-time genetic recommendation method in order to overcome the existing recommendation techniques are not reflect the current user's intend. With the genetic algorithm newer solutions can be generated providing optimal solution each time when the algorithm is made to run, thus providing mutations. This method can be compared with the existing ones which lack the quality of providing accurate results.

REFERENCES

- [1] Hyun – Tae Kim, Eungyeong Kim, “Recommender system based on genetic algorithm for music data”, 2nd International Conference on Computer Engineering and Technology, 2010.
- [2] J. Ben Schafer, Joseph Konstan, John Riedl, “Recommender Systems in E-Commerce”, 2007.
- [3] D.E Goldberg and J. H Holland , “Genetic Algorithms and Machine Learning”, 2006.
- [4] Paul resnick and Hal .R. varian, “Recommender system”, Vol. 40, No. 3 COMMUNICATIONS OF THE ACM, March 1997.
- [5] Tom V. Mathew, “Genetic algorithm”, http://www.civil.iitb.ac.in/tvm/2701_dga/2701-ga-notes/gadoc/gadoc.html.
- [6] Xiaoyuan Zhu, Yongquan Yu, Xueyan Guo, “Genetic Algorithm based on Evolution Strategy and the Application in Data Mining”, pp-849-852, First International Workshop on Education Technology and Computer Science, 2009.
- [7] Janusz Sobecki, “Implementations of Web-based Recommender Systems Using Hybrid Methods”, International Journal of Computer Science & Applications ,2006 Technomathematics Research Foundation, Vol. 3 Issue 3, pp 52 – 64.