Recommender system for music data
using genetic algorithm

Namdeo Badhe, Divya Mishra, Chanchal Joshi, Neha Shukla

Thakur College of Engineering,
Mumbai, Maharashtra.

ABSTRACT
In this paper we try to develop a recommender system for music data that will predict the ecommerce users music according some information filtering algorithms. This personalized approach to each users will help the users to download their desired music. Here we are using genetic algorithm. Genetic algorithm are general purpose optimized algorithm that predicts effectively to changes in users preference unlike other different algorithms available.

Keywords: Recommender system, Genetic algorithm, User Profiling, Content Based filtering

1. INTRODUCTION
The amount of information on ecommerce sites are increasing day by day. It becomes difficult for ecommerce users to choose the desired product from such a bulk of information. recommender systems are an effective solution for it. Recommender System are normally an information filtering technique that predicts the user items according to users personalized information obtained and from results of algorithm. In case of music website different categories of music may be available recommendation in this type of application will include recommending every user music according to the rating of the song and user profile and preferences.

2. SCOPE OF THE PROJECT
The proposed system has great scope since this system can be used in almost ecommerce sites for music. We have chosen music since unlike other products one cannot just view and select the product. Listening to all music may be a tedious task. So this system can be of great use in such cases. Since this system is a dynamic one. Recommendation results for each user changes with the user preference. One growing area of research in the area of recommender systems is mobile recommender systems. With the increasing ubiquity of internet-accessing smart phones, it is now possible to offer personalized, context-sensitive recommendations.

3. PROPOSED METHODOLOGY
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. The final step after applying genetic algorithm is displaying the items that are closest to the items which the user has given the highest rating. Using Euclidean distance formula the nearest possible music features which are matching with the one generated by crossover step of genetic algorithm are matched and given as out for recommended items. Here a separate recommendation page is displayed where the top ten similar records matching which the two offsprings generated is displayed.

Genetic algorithm procedure:

1. Choose the initial population of individuals
2. Evaluate the fitness of each individual in that population
3. Repeat on this generation until termination (time limit, sufficient fitness achieved, etc.):
4. Select the best-fit individuals for reproduction
5. Breed new individuals through crossover and mutation operations to give birth to offspring.

6. Evaluate the individual fitness of new individuals.

7. Replace least-fit population with new individuals.

The very first phase in any system developing life cycle is preliminary investigation. The feasibility study is a major part of this phase. A measure of how beneficial or practical the development of any information system would be to the organization is the feasibility.

The feasibility of the development software can be studied in terms of the following aspects:

1. Operational Feasibility.
2. Technical Feasibility.
3. Economical feasibility.
4. Motivational Feasibility.
5. Legal Feasibility.

**OPERATIONAL FEASIBILITY**

The site will reduce the time consumed to maintain manual records and is not tiresome and cumbersome to maintain the records. Hence operational feasibility is assured.

**TECHNICAL FEASIBILITY**

Minimum hardware requirements:

- At least 166 MHz Pentium Processor or Intel compatible processor.
- At least 256 MB RAM.
- 14.4 kbps or higher modem.
- A video graphics card.
- A mouse or other pointing device.
- At least 1 Gb free hard disk space.
- Microsoft Internet Explorer 4.0 or higher.

**ECONOMICAL FEASIBILITY**

Once the hardware and software requirements get fulfilled, there is no need for the user of our system to spend for any additional overhead. For the user, the web site will be economically feasible in the following aspects:

- The web site will reduce a lot of paper work. Hence the cost will be reduced.
- Our web site will reduce the time that is wasted in manual processes.
> **MOTIVATIONAL FEASIBILITY**

The users of our system need no additional training. Visitors do not require entering password and are shown the appropriate information.

> **LEGAL FEASIBILITY**

The licensed copy of the required software is quite cheap and easy to get. So from legal point of view the proposed system is legally feasible. Software Development Model Used.

Software process model deals with the model which we are going to use for the development of the project. There are many software process models available but while choosing it we should choose it according to the project size that is whether it is industry scale project or big scale project or medium scale project. Accordingly the model which we choose should be suitable for the project as the software process model changes the cost of the project also changes because the steps in each software process model varies. This software is build using the waterfall mode. This model suggests work cascading from step to step like a series of waterfalls. It consists of the following steps in the following manner

### 6. FLOW GRAPH

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 favourite.

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 favourite list where that user has given highest rating to the audio tracks.

### FLOW OF SYSTEM

#### 7. DESIGN

The design stage takes as its initial input the requirements identified in the approved requirements document. For each requirement, a set of one or more design elements will be produced as a result of interviews, workshops, and/or prototype efforts. Design elements describe the desired software features in detail, and generally include functional hierarchy diagrams, screen layout diagrams, tables of business rules, business process diagrams, pseudo code, and a complete entity-relationship diagram with a full data dictionary. These design elements are intended to describe the software in sufficient detail that skilled programmers may develop the software with minimal additional input.
Waterfall model design

**Analysis Phase**

To attack a problem by breaking it into sub-problems. The objective of analysis is to determine exactly what must be done to solve the problem. Typically, the system’s logical elements (its boundaries, processes, and data) are defined during analysis.

**Design Phase**

The objective of design is to determine how the problem will be solved. During design the analyst’s focus shifts from the logical to the physical. Data elements are grouped to form physical data structures, screens, reports, files and databases.

**Coding Phase**

The system is created during this phase. Programs are coded, debugged, documented, and tested. New hardware is selected and ordered. Procedures are written and tested. End-user documentation is prepared. Databases and files are initialized. Users are trained.

**Testing Phase**

Once the system is developed, it is tested to ensure that it does what it was designed to do. After the system passes its final test and any remaining problems are corrected, the system is implemented and released to the user.

8. **CONCLUSION**

Thus the proposed system combined the strengths of content-based filtering and interactive genetic algorithm.

The proposed system initially extracts the unique features of each item using the content-based filtering. As inputs to genetic algorithm, the individuals are composed of the extracted features of the data set. The system then requests the user to evaluate the fitness of each item. Next, genetic algorithm operates on the evaluated items to discover the most appropriate items for recommendation while the data grouping technique is applied to search for the candidate items more rapidly.

9. **RESULTS**

We proposed real-time genetic recommendation method in order to overcome the existing recommendation techniques that are not reflecting in 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.

10. **ACKNOWLEDGEMENT**

The Authors will like to thank Mr. Namdeo Bade, Department of Information Technology for their guidance and throughout support.

11. **REFERENCES**


