

國立交通大學

資訊科學所

碩士論文

以人機合作的方式來評斷大量數位作品



**Criticize a Great Number of Subjective Compositions
by the Cooperation between Human and Agents**

研究生: 許仁鴻

指導教授: 孫春在 教授

中華民國九十四年六月

致 謝

在交大度過學生生涯美好的六年時間，接下來要邁向人生另一個階段，因此以這篇論文作為一個紀念並感謝六年來師長們的教誨。首先我要感謝我的指導教授孫春在老師，老師除了在研究上給了我很多意見及方向的指引，也讓我體會了很多比專業技能更重要的處世想法，我很難形容這種感覺，或許套用老師曾經說過的話：做研究最大的樂趣便是能更了解自己一點。我感謝老師讓我能有試著看遠一點的想法，我也希望往後的路我能夠更了解自己。

此外我當然也要感謝一起在實驗室打拼的夥伴們，有大家的扶持、勉勵以及打屁，我過了既充實又獲益良多的研究生生涯。感謝吉隆學長跟崇源學長支持我的研究以及提供寶貴的想法，同實驗室的同學跟我討論並且給我建議，讓我的論文能夠更加的完整以及明確，而且又不厭其煩地作我擾人的實驗並提供我驗證的數據，我真是非常非常的感謝大家。

最後我由衷的感謝我的父母以及妹妹，他們支持給予了我前進的力量，他們的溫暖讓我沒有後顧之憂的作我想做的事，有太多的想法跟話語我無法呈現，只能以這篇致謝小小地感謝所有幫過我的人，謝謝。

以人機合作的方式來評斷大量數位作品

學生：許仁鴻

指導教授：孫春在 博士

國立交通大學

資訊科學研究所

摘要

本研究提出了一個以人和代理人合作來評斷大量主觀作品的模型。主要概念是以演化式計算的方法來訓練一群能夠察覺使用者喜好的代理人，在架構中代理人扮演著中介的角色，以這些訓練過的代理人來逐一評斷物件，省去由使用者親自評斷的大量時間，根據評分的結果進而推薦使用者可能有興趣的作品。此模型可以應用在許多類型的作品評斷之上，像是音樂，圖畫，電影 等。本研究中以此概念實作一個個人化音樂推薦系統。在此系統中使用了八種音樂特徵來表示樂曲的特性，並且藉由互動式演化計算的方式，讓使用者直接訓練一群能符合該使用者喜好的代理人，此外也提出一些機制來減少因人類參與演化所會帶來的問題。最後進行一系列的實驗來證實以此模型實作的推薦系統的確有良好的效能，並提出此模型的其他應用方法。

Criticize a Great Number of Subjective Compositions by the Cooperation between Human and Agents

Student: Jen-Hung Hsu

Advisor: Dr. Chuen-Tsai Sun

Institute of Computer and Information Science

National Chiao-Tung University

Abstract

This thesis presents a model which evaluates a large number of subjective compositions by the cooperation between human and agents. The main concept is to train a group of agents which satisfy the user's taste by evolutionary computing. The agents play the role of an intermediary in the model and evaluate all compositions for the user. For this reason, the user can reduce the time of directly evaluating.

According to the results of the agents' evaluations, the system could further recommend the items the user may be interested in. This model could consider many kinds of compositions, like music, pictures, movies, and so on. We implement a personalized music recommendation system with this framework. The system applies eight kinds of musical features to represent the music items, and let the user directly train a group of agents which fit the user's preference by interactive evolutionary computing. Furthermore, we present some additional mechanisms to reduce the problems which result from the human beings participating in the evolution. Finally, a series of experiments are executed to show that our approach performs well.

目錄

1. Introduction.....	3
2. Related Work.....	5
2.1 Recommendation system	5
2.2 Interactive Genetic Algorithm.....	6
2.3 The user preference model.....	7
3. Music recommendation system.....	9
3.1 Representation Component.....	11
3.1.1 Track selector	11
3.1.2 Feature extractor	12
3.2 Evolution Component	13
3.2.1 Recommendation Agents module	14
3.2.2 Evolution manager	16
3.3 Characteristics of the system	20
4. Experiments	21
4.1 Quality of recommendations :	24
4.2 Convergence Test :	26
5. Conclusion	29
6. References.....	31
Appendix A	34
A.1 Midi 檔案分析.....	34
A.2 音樂特徵抽取.....	35
A.3 統計音程表.....	37
Appendix B	41
B.1 初始分布策略.....	41
B.2 系統演化流程圖.....	43

圖目錄

Figure 1. The system architecture.....	10
Figure 2. The concept of evolution.....	14
Figure 3. The chromosome architecture.....	15
Figure 4. Selection Box.....	18
Figure 5. The user interface.....	22
Figure 6. The demonstrating test.....	23
Figure 7. Precision of recommendation.....	26
Figure 8 Converge Test.....	27
Figure 9. An Example of the user's result.....	28
Figure 10 Professional evaluation system.....	30



1. Introduction

Since the birth of the first web browser, Netscape, in 1994, thousands of surfers have been attracted to “explore” what the story is behind the Internet. The information booms subject to the Internet techniques growing fast within a recent decade. Users feel at a loss about a great deal of information flourishing on the Internet. Therefore, many kinds of the service that aims to filter the info are developed one after another.

The common service is to show some information actively according to the statistical data. For example, the news webpage specially lists the news which most people read in the main page, or the web book store, like amazon.com, shows the books billboard. This service recommends by gathering the statistics of all users' behavior. But it's useless regarding the items about personal feelings, such as music, movie, or other objects of Art because doesn't consider the difference of users' preferences.

Appreciating Art is the subjective doings. For example, songs on the billboard are not favored by everyone. If we ask the person why he/she likes this song, sometimes the person can't explain the reason. Therefore, if the data items are about human beings' subjective judgment, the system must consider from the viewpoint of individual difference and is not suitable to recommend by the statistic info of the mass population.

The personal recommendation service recommend data items that users may be interested in based on users' predefined preference or user's access history. In other words, the main purpose of this service is to help users to pick the interested items. Various items have been considered in these recommendation systems, such as music, WebPages, movies, and books. According to the property of the method, there are two

major approaches of the personalized recommendation system. (1) Content-based filtering — Analyzing the content of the items and find the features which the user might interest in. (2) Collaborative filtering — Grouping the users who have the same interests and sharing what they access in common. These two approaches consider with the opposite aspects. One is concerned with the analysis of the items content, and the other focus on the connections of the users. These two methods also have advantages and shortcoming. The detailed definitions and related works would be introduced in following sections.

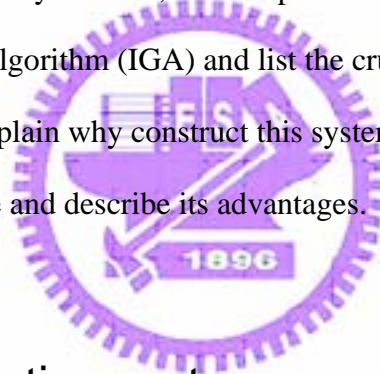
This paper presents a model which criticizes a large number of music items by the cooperation between human and agents. The main concept is to train a group of agents which understand the user's preference by using the method of GA. These agents play the role of an intermediary in the system, and evaluate all music items for the user. For this reason, the user can reduce the time of evaluating directly. According the results of the agent's evaluations, the system will collect the high grades items to recommend.

The key point of constructing personalized recommendation system is how to adapt the system to the user's preference. Our model adopts the method of *Evolutionary Computing* (EC) to train a group of agents, and let these agents learn to satisfy the user's taste. Different from the common EC model using the defined fitness function to evaluate the agents each round, our model replaces the fitness function with human beings' subjective judgments in order to promote the agents more fitting the user. Due to the participation of humans, some additional problems have to be overcome, for example, the design of the user interface and the methods of reducing the human fatigue. The following section will describe these problems and present the method to figure out them.

In section 2, the related works will introduce and indicate the difference between our model and others. The research implements a music recommendation system. The framework of this system is described detail in section 3. Section 4 will show the result of the system performance and the experiment snapshot. Finally, a conclusion is given in section 5.

2. Related Work

The following paragraphs introduce the recommendation service system and indicate its shortcomings firstly. Second, we will probe into the related researches about Interactive Genetic Algorithm (IGA) and list the crucial points when using this method. Finally, we will explain why construct this system by adopting agents to satisfy the users' preference and describe its advantages.



2.1 Recommendation system

The recommendation system recommends the data items that users may be interested in based on users' predefined preference or user's access history. Various items have been considered in these recommendation systems, such as music[1-3], WebPages[4-6], movies[7, 8], and books[9].

There are two major approaches of the personalized recommendation system. One is the content-based filtering, which analyzes the content of items that the user preferred in the past and recommends the similar items. In other words, this approach recommends according to the connection of users' preference and the content of items. In this approach, the representation of data items and the records of users' preference

are key issues to affect the function of the recommendation system.[10] However, the recommendation systems adopting the content-based filtering approach can only recommend the data items in which the user has indicated his/her interest. Other potential interesting data items cannot be explored in such recommendation systems if the users never access before.[11]

Different from the previous approach, the collaborative filtering approach makes the recommendation by grouping the users who have the same interests and sharing what they access in common. Broadly speaking, the main goal of the collaborative approach is to make the recommendation among the users in the same group. The recommending approach has a high possibility to recommend surprising items by the nature of information sharing, which cannot be achieved by the content-based filtering approach. However, the bootstrapping of this approach may sometimes be hard and take a long time. [1, 12]



2.2 Interactive Genetic Algorithm

In 1975, John Holland referred the mechanism about the evolution of the Nature and proposed *genetic algorithm* (GA), an artificial intelligent system invented for the optimal solution of the problem. Under the construction of GA, the chromosome structure of individuals will be designed according to the problem, and the genes of the chromosome will be generated randomly when the system initializes. The agents evaluate the individual's performance to the unsolved problem by a fitness function and decide which one should be preserved or discarded in next run. The discarded ones will be replaced with new individuals whose genes are got from the preserved ones.

According to this concept of "Survival of the fittest", GA repeats above

procedures of evolution until the optimal solution of the problem is figured out. However, if we would like to solve the problem about Art by GA, such as appreciating music or paintings, it is hardly to define an effective and clear fitness function which can substitute human beings' subjective judgment. This kind of problem which needs human beings' subjective judgment is not only limited in art but in engineering and education, like database retrieval and writing education.[13, 14]

Interactive Genetic Algorithm(IGA) is an optimization method that adopts GA among system optimization based on human evolutionary[15]. In other words, it is simply a GA technique whose fitness function is replaced by a human user.

Because of users' participation, IGA has more limitations than GA. The main factor affecting the evaluation of IGA is human beings' emotion and fatigue. When processing the evaluation of each run, the users cannot make the fair judgment; therefore, the result will be changed in the different occasion due to the people's emotion. Furthermore, people will feel tired and fail to process with large population. Therefore, how to search for a goal with a smaller population size within a fewer number of searching generations is the important problem. Another problem is fluctuation of human evaluation which would result in the inconsistency of different generation. [16-18]

2.3 The user preference model

No matter what kinds of method the personalized recommendation system, the key point is how to adapt the system to the users' preference. According to the previous research[19], the users' preference model can be constructed in two approach as follows[20, 21]:

(1) Implicit - Observing the user's behavior (Machine-learning) or inferring from

domain knowledge or other user information (Knowledge-engineered).

(2) Explicit- Using survey, dialog or any other methods to obtain the user knowledge directly (User-programmed).

Normally, the recommendation systems always belong to the 1st approach. Via analyzing the behavior of the user, like the access history or the category which the user feel more interested in, the system will construct the user preference model automatically and then will make the recommendation based on the user model. In this approach, the users will be unconscious that some software programs are gathering the information when operating the system.

In this study, the user needs to train a group of agents actively to be the intermediary between human beings and data items. Obviously our system belongs to the second approach of the user modeling. Compared with the existing recommendation systems, our system spends more time on training the agents in the beginning, but we can adapt the systems to the user' preference in shorter time, and don't need to waste time searching and collecting the users' information in the accessed history. Furthermore, from the viewpoint of the interaction between human beings and the system, the users directly and actively adapt the system to his/her preference, resulting in our model is also more effective to satisfy the users than the above- mentioned indirect methods.

3. Music recommendation system

In this study, we design a personalized music recommendation system under the concept of the cooperation between human beings and agents, and expect to figure out how to criticize a great number of subjective works well. As shown in Fig.1, the system consists six function blocks, namely, the track selector, the feature extractor, recommendation agents module, evolution manager, user interface, and the database. According to the usage, we classify these function blocks in two components:

(1) Representation Component:

This component exists three function blocks: track selector, feature extractor, and the database. The main purpose of this component is to transfer the conceptual properties in musical items to the useful information with specific values and to store the information in the database for the later procedures. Simply speaking, this is the pre-processing section of the recommendation system.

(2) Evolution Component:

This component is the bridge between the users and musical items. It is formed with recommendation agents module, evolution manger, and user interface. The main function of this component is to satisfy the users' taste by building the GA model and implementing the concept of optimizing the agents during the evolution.

When a new musical object is inserted into the system and processed through the track selector and feature extractor, it specifies the musical features and stores them in

the database in order to let agents access and select. After the user evaluates recommended musical items via user interface, the grades will be dispatched to the agents which recommend the musical items. According to the performance of agents, the evolution manager controls the GA procedure, like selection of the population, crossover, and mutation, and monitors the status of the system.

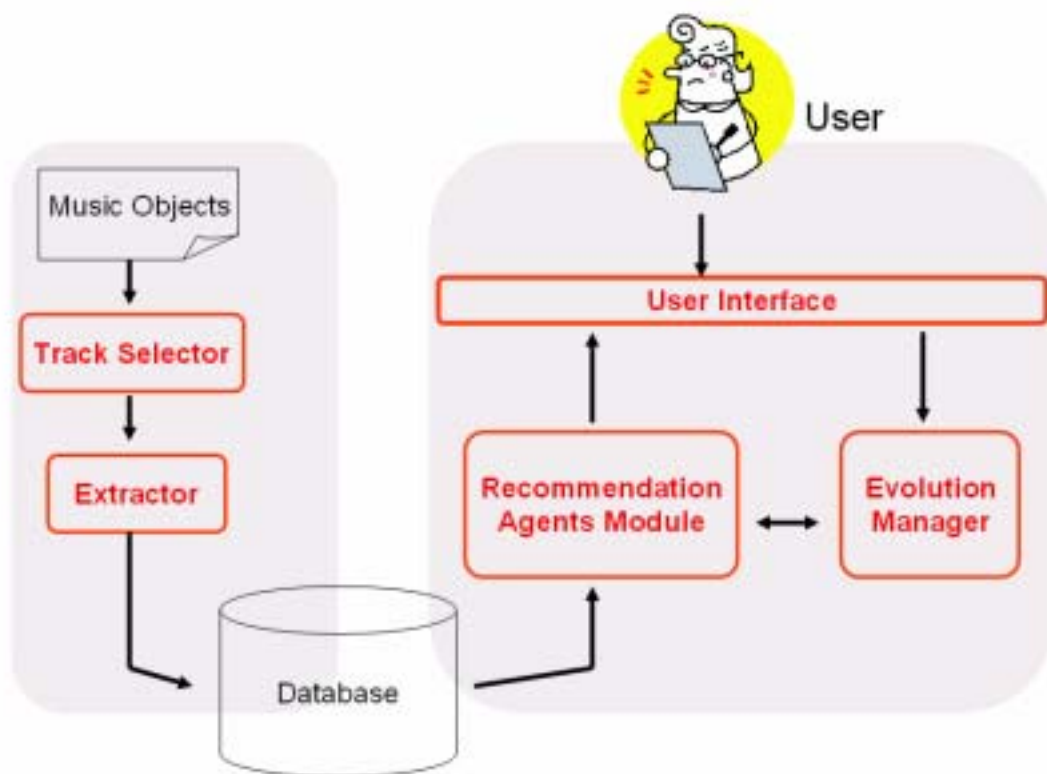


Figure 1. The system architecture

In the following sections, we will explain the concepts and the functions of Function Block and will summarize the characteristics of the personalized music recommendation system. In Chapter 4, how the system really works in the experiment will be described.

3.1 Representation Component

This component could be viewed as the pre-processing stage of the system to analyze and extract features from musical items, **so the system can connect the users' taste with musical items**. In short, the more detailed the item analysis is, the closer the trained agents get to satisfy the user's taste.

3.1.1 Track selector

In the system, the musical items are of polyphonic MIDI format. A Polyphonic musical item usually consists of several tracks- one for melody and the others for accompaniments. The track for melody is regarded as the representative track which contains the most semantics. In the following procedures of feature extracting, the main melody is the focus; therefore, to find out precisely the track of the main melody helps a lot in the accuracy of feature extracting.

The selection of the representative track is made by analyzing the pitch density. The concept is that the track for melody contains much more distinct notes with different pitches than the tracks for accompaniment. According to the research before[3], an 83% correctness rate is achieved by this method. The pitch density of a track is defined as follows:

$$\text{Pitch density} = NP/AP$$

where NP is the number of distinct pitches in the track and AP is the number of all distinct pitches in MIDI standard.

The pitch densities of all tracks of the target music object are computed by above equation. The track with the highest density is then selected as the representative track of a polyphonic music item.

3.1.2 Feature extractor

The purpose of the feature extractor is to extract features from the perceptual properties of the musical items, and transfer into the distinct data. The eight features used in our system are described as follows:

a. Tempo degree

The tempo degree is defined as the average value of the note length which can be derived from MIDI files.

b. Loudness

The feature of loudness is defined as the average value of the note velocities which can be derived from MIDI files.

c. Pitch Entropy

The pitch entropy, derived from Sayood [22], is defined as follows:

$$PitchEntropy = -\sum_{j=1}^{NP} P_j \log P_j$$

where P_j is defined as follows:

$$P_j = \frac{N_j}{T}$$

where N_j is the total number of notes with the corresponding pitch in the main track, T is the total number of notes in the main track.

d. Pitch Density

This definition has been given in last section.

- e. Mean of the pitch values

The definition is the mean of the pitches in all tracks.

- f. Standard deviation of the pitch values

The definition is the standard deviation of the pitches in all tracks.

- g. Number of Channels

This is the number of the channels which appears in the music item.

- h. The catalog of pitch interval

It is known that the pitch interval of varied music types differs from each other, from which we came out ten catalogs of pitch interval as the standard. When inserting a new song, the system will compare the fresh interval with the standard and do the classification.

3.2 Evolution Component

This component constructs the evolution model based on IEC. In this model, the users' appreciation is the source of pressure of evolution to adapt the agents to the users' preference. The flow path of evolution is shown in [Fig.2](#)

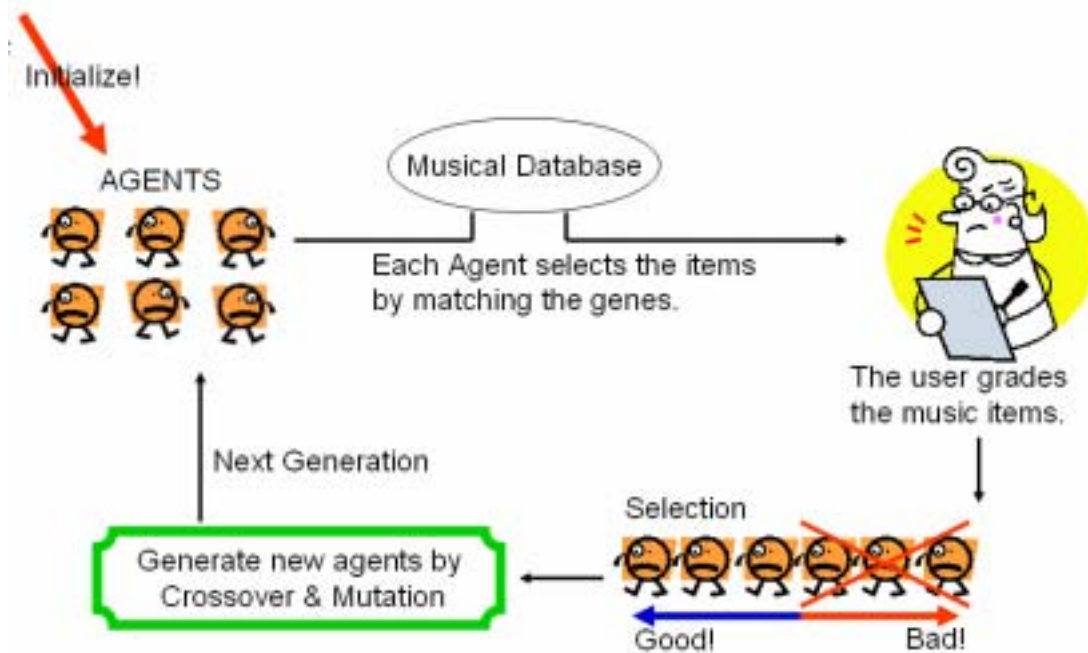


Figure 2. The concept of evolution

3.2.1 Recommendation Agents module

This module contains the design of the agent and the method to pick up songs. In the model of Genetic Algorithm, the first procedure is *chromosome encoding* which design the structure of chromosomes based on the problem solving. An agent can own many chromosomes according to requirement. The procedure of *chromosome encoding* will affect the performance and result of evolution.

In our system, each agent has a certain chromosome where each gene relatively represents each musical feature. The value of genes stands for the preference to the eight musical features. In our design, the gene is divided into three by the storage of information- integer, range, and the attribute of “Don’t care”. An example of the chromosome has been shown in [Fig.3](#).

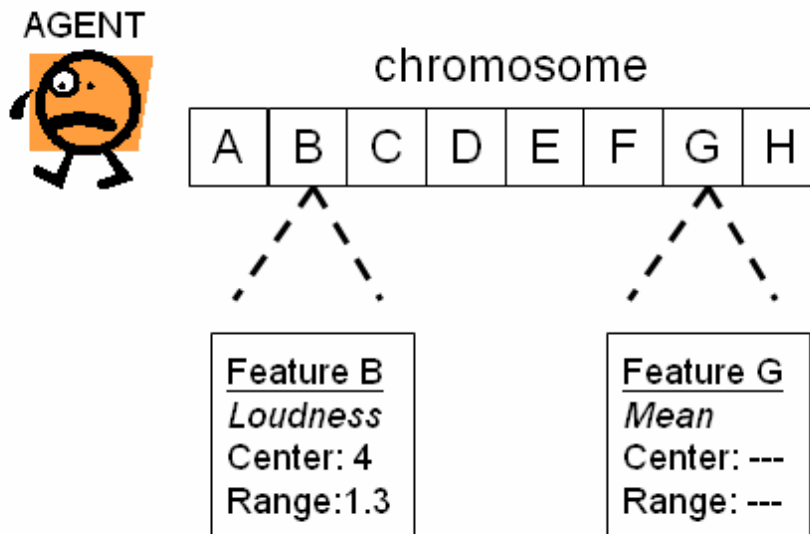


Figure 3. The chromosome architecture

In most of GA systems, the genes in the chromosome are initialized randomly when initializing the GA system. In the case of our system, the random agents probably cannot find the musical items to fit their genes because the distribution of the extracted features from musical items is not balanced. Therefore, the initialization of the agents would refer to the corresponding musical feature statistics in the database, and limits the range of initial value by an appropriate proportion. The procedure avoids that the preferences of initial agents are too strange to find musical items. The agents with strange taste can't get the grades from the users because they fail to recommend the users any musical item. From the viewpoint of evolution, these agents will be discarded some time or other. Consequently, the speed of evolution would increase by excluding these agents in the beginning.

The agents use the chromosome to compare the consistency with all musical items in the database, and pick up the most similar ones that are recommended to the user. We present a mechanism for the users to score the musical items in three aspects; namely, *melody*, *style*, and *originality*. The main purpose of this mechanism is to

enhance the variety of recommended musical items, and to let the users evaluate more precisely. For example, if the user doesn't feel the agent A's recommended musical items melodious but creative, this agent won't be eliminated because of getting the lower score in a single aspect. It's pity to eliminate the potential genes from the agents; therefore, we adopt the mechanism of scoring in the multiple aspects to increase the living opportunity of the creative agents in order to avoid a certain agent from being "the dictator" in the system.

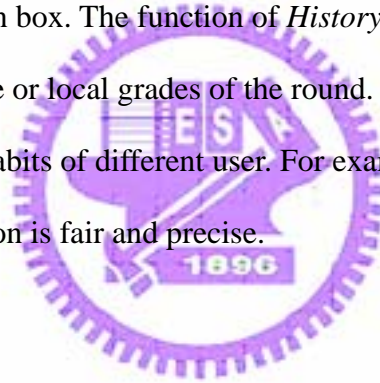
3.2.2 Evolution manager

The most important issue in the model of GA is how to preserve good genes for generating the better and more effective offspring. The common procedure is to select the top agents as *parent generations* to breed new individuals by mixing their genes to replace the eliminated agents

This method is reasonable and effective, but not suitable in our model of optimizing the evolution by the personal subjective evaluation. The *human fluctuation* is an important problem in the system based on IEC, and results in evaluating unfair in every round. In other words, the criterion of the user's evaluation is instable in different rounds. The outstanding agents in previous rounds probably get low grades because of the human fluctuation, and this unexpected failure will cause the good ones to be discarded. Furthermore, there would be an error when agents pick up the musical items by their "intuition". That is to say the recommended musical items sometimes are not enough to stand for the agent's judgment of good taste. For this reason, the problem of discarding the wrong agents will be enlarged in our model.

In order to solve the above-mentioned problems, we propose an idea of *agent*

fame value. The agents get a fame value according to the previous behavior. The higher fame values are, the more possible agents survive. Fig.4 shows the example of the status after the users' evaluation in the round, and then will run the GA selection procedure which picks up the preserved parent generations as well as the discarded agents. As shown in Fig.4 , each agent owns two kinds of attributes; namely, the agent fame value and the fitness values of this round. *Selection Method* in the system determines which agents would be discarded or recombined according to the result of weighted computing agents fame value and local grades of this round. After the selection, the local grades of this round also would be merged into the agent fame value for usage of next round. In addition, the *history ratio* parameter can affect the computation in the selection box. The function of *History ratio* is to lay particular stress on agents' fame value or local grades of the round. We can modulate *history ratio* to adapt the scoring habits of different user. For example, *history ratio* could be raised if the user's evaluation is fair and precise.



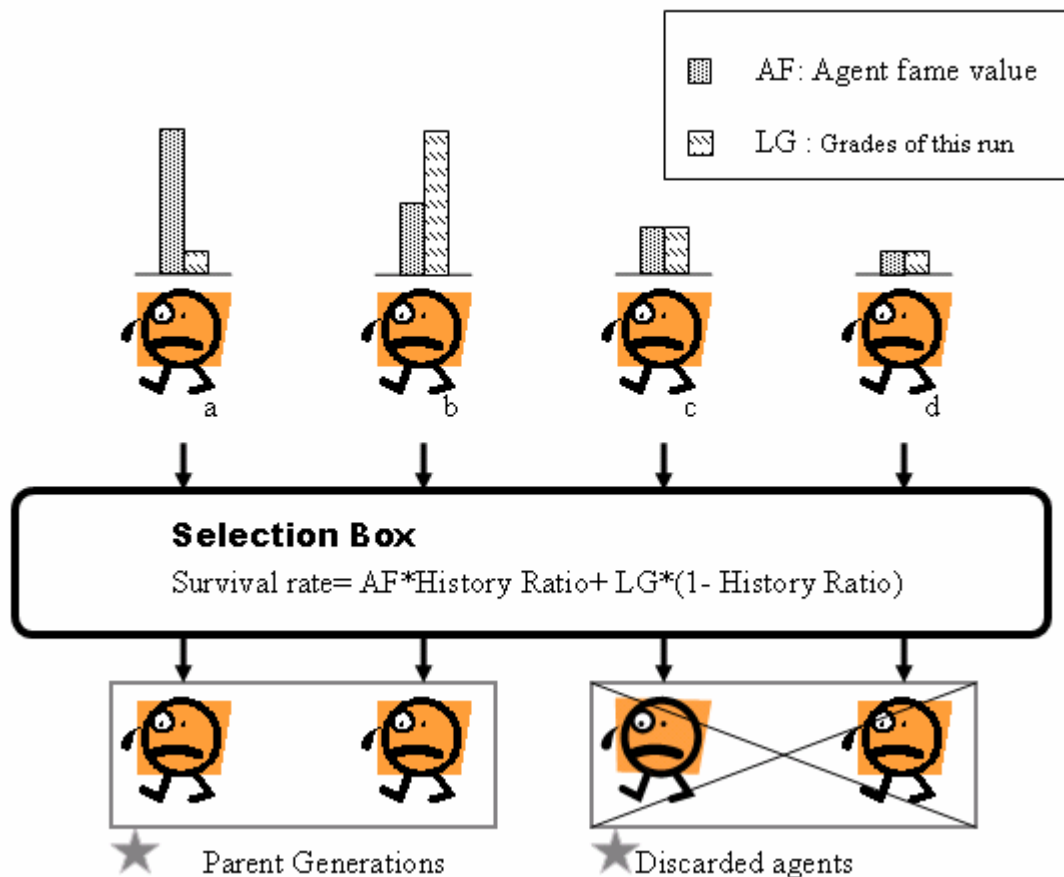


Figure 4. Selection Box

To determine the time of stopping the evolution and system converging is an important issue in the systems based on GA. Generally speaking, the methods of determining the time are to observe whether the system learning curve has ceased moving or the result of evolution has achieved the expected objective. However, as we have described, we make use of the human beings instead of the fitness function to solve the problem involving subjectivity without criterion. It is for sure that there is no way to define the criterion of judging Art and relatively, there is not an impartial solution to verify our system has converged or the agents have been trained completely, either. Therefore we propose another solution of determining the converged time by using the agent fame value as before. Just as the public relationship in our society, the personal fame would increase or decrease by the

behavior. Usually the human would become a consultant if he/she holds good fame for a long time. We take advantage of this concept to our system for the usage of determining the converging time. In our system, each agent's fame value varies in every round, and the system will monitor the agent population to find which agent usually maintains high fame values during a period. If the agent of high fame values with good behavior can past the examination of the time threshold, this agent will be allowed to enter the V.I.P pool. The agents in this pool would not evolve but still keep sharing the genes with the others agents in the circle of evolution. When gathering enough stable agents in the V.I.P pool, the system will terminate the evolution and take these stable agents as the final population for recommendation.

In the process of fundamental GA, the genes of agents with good behavior sometimes would be broken because of ongoing crossover and mutation. That is to say some agents had already missed the most proper timing of stopping the evolution. So selecting the agents with good behavior can preserve the good genes of the agents in the evolution process and avoid the good structure of genes from being destroyed in the overly evolution.

Besides, this procedure can make the agent population varied. There is always a direct answer for the GA questions and leads to the similar agents population when system converged. But in our case, we hope to train the agents with various styles in order to fit the user's taste. For this reason, this system adopts the procedure as described before to collect agents, and then "the last survivors" in the final round can recommend multiple kinds of musical items.

3.3 Characteristics of the system

In this system, the agents grade all musical items individually, and choose the one with the highest scores as the recommended item; consequently, the recommended group are not only affected by the single agent but blended with the different agent's taste. So the agents may recommend surprising musical items. For example, assuming that the agent A gives the JAZZ style high grades; the agent B prefers the CLASSIC type, and then the musical items recommended by these two agents may be blended with the two styles, or be totally different from the original.

The content-based filtering type of the recommendation system recommends items based on the connection between the user's preference and content of the items. Therefore, the representation of the data items; namely, the definition of the extracting features will influence the recommendation. After the process of feature extracting, some systems would have an additional pre-process procedure to classify the data items by the specific features for accelerating the later recommendation procedure. For example, to classify the music items by the tempo feature, if the user just likes the musical items of quick tempo, the system can find the recommended ones in a short time by the early classification. Furthermore, it is able to classify by multiple features. However, this procedure has an important problem, which is that the classified rules must be decided before the process of recommendation. The user's taste is different from the other, so it's difficult to define the general classified rules.

The chromosome structure in our system is also a kind of classified rules from a different point of view. That is to say that varied chromosomes classify the different music groups according to the property or the range of the genes oneself. Differing from the pre-defined classified rules belonging to the pre-process of the system, the agents of our model adapt to the user's taste gradually in the learning stage of the

system. In other words, our agent system based on GA classified dynamically, which discovers the proper rules of the user by the evolution procedure. Comparing with the method using fixed rules, our method is more flexible and precise.

As the other recommendation systems, our system also needs the detailed definitions of musical features to represent the property of the musical items to make the agents recommend more precisely. But some obscure features are permitted to exist in our model. The design of chromosome encoding added a property of “Don’t Care”; namely, to ignore this feature. If the feature B is meaningless to the user, it could be disregarded gradually during the evolution. Consequently, in our system, the features can be defined loosely, and it is not necessary to worry that some useless features will cause the incorrect recommendations.

4. Experiments



We implemented a music recommendation system according to the thesis as described before, and invite 25 persons to perform the system. Fig.5 shows the user interface of the system. Some parameters set in the experiment are shown as follow : The size of the midi database is 1036. The initial population consists of 6 random agents. And the number of the agents could recommend each generation is 12.

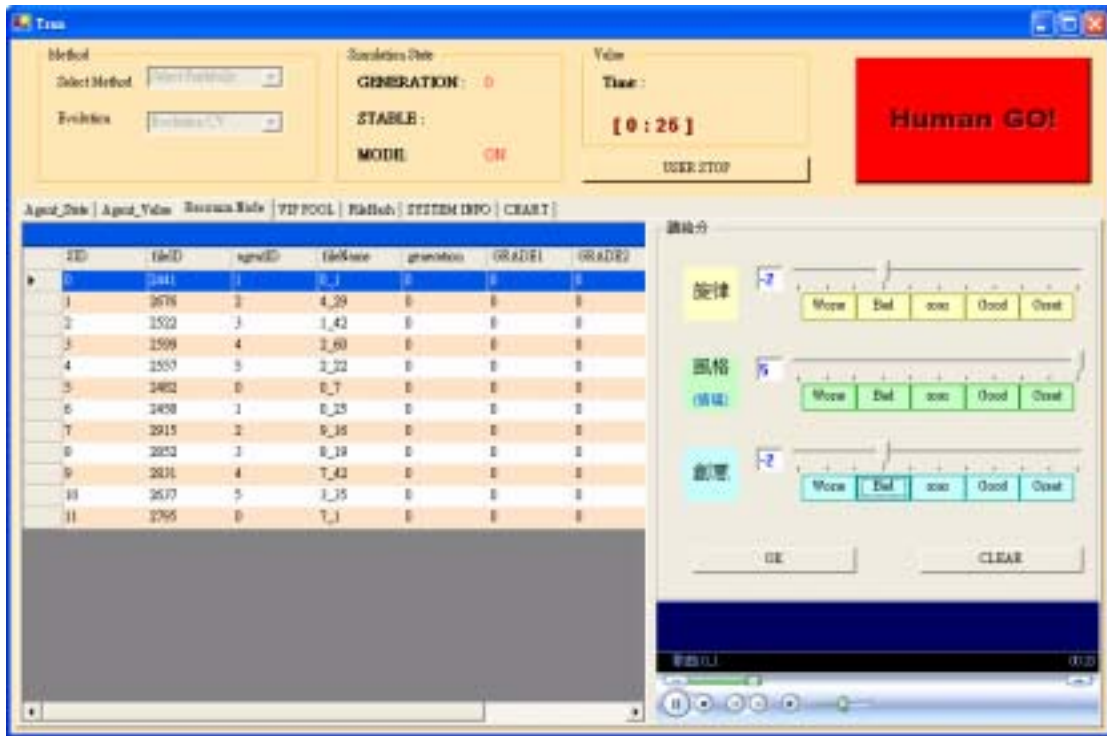


Figure 5. The user interface



The procedures of the experiment divide into two stages:

1. Training phase:

In this phase, the user would train a group of agents which satisfy his/her taste.

At the beginning of each round, the system presents some music items to the user.

As the user gives recommended music items fitness value by preference, the system generate a new population in the next generation using the GA method.

This procedure is repeated until the system determines that the evolution is converged.

2. Validating phase:

After the training phase, the system has adapted the user's taste. And then the user performs an additional test for verifying that this system recommends

correctly. Among the demonstrating test, the system let the agents which satisfy the user's taste grade all music items. We select 20 music items of highest grades as the experimental group and collect another 20 music items randomly as the control group, and let the user evaluates these two groups individually. Fig.6 shows the interface of the demonstrating test.

First we will analyze the results of the demonstrating test to explain whether the system archives the target. Second the convergence test of the training phase would be discussed. Moreover, the rest part of this chapter describes an observation about the result of the experiment.



Figure 6.The demonstrating test

4.1 Quality of recommendations :

The quality of recommendation is measured in two aspects, namely, precision rate and weighted grade. The definitions are described as follow:

Precision

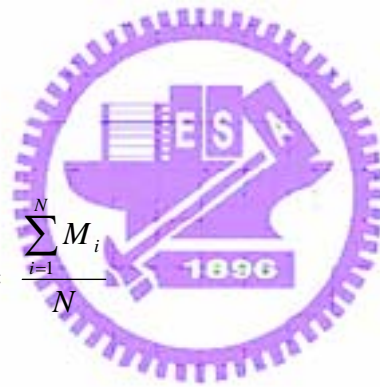
$$\text{precision rate} = \frac{N_s}{N}$$

(Eq. 1)

Where N_s is the number of the successful samples, and N is the number of total music items.

Weighted grades

$$\text{weighted grades} = \frac{\sum_{i=1}^N M_i}{N}$$



(Eq. 2)

Where M_i is the grades of music items, and N is the number of total music items.

The demonstrating test provides six degrees to the user to evaluate the music item. The definitions of six degree are shown in Table.1. The meanings of upper three degrees all indicate the user interests in this song. Form the viewpoint of recommendation, the songs which evaluated with these three degrees can be considered as the successful samples. On the contrary, the songs which evaluated with the lower three degrees are viewed as the failed examples.

Degree	Grade	Precision of recommendation
Great	10	Success
Good	8	Success
Not Bad	6	Success
Not Good	4	Fail
Bad	2	Fail
Terrible	0	Fail

Table. 1. The six evaluating degree table

The user evaluates the experimental group and the control group. The agents in the experimental group evaluate the songs according to their evaluation function trained by human, but the agents in control group only evaluate the songs randomly. After the user finished the test, the system calculates the precision rate and weighted grades by the Eq2 and Eq3.

The Fig.7 shows the results of the demonstrating test. The songs recommended by trained agents have an 84% precision rate, and get the weighted grades of 7.38. Because the music items of the control group are just selected randomly, the random method only gets about 50% precision rate, and 5~6 weighted grade. Comparing with the control group, the experimental group obtains the better results without respect to the precision rate of recommendation or weighted grades. The result shows that our system certainly achieves the target of personalized recommendation.

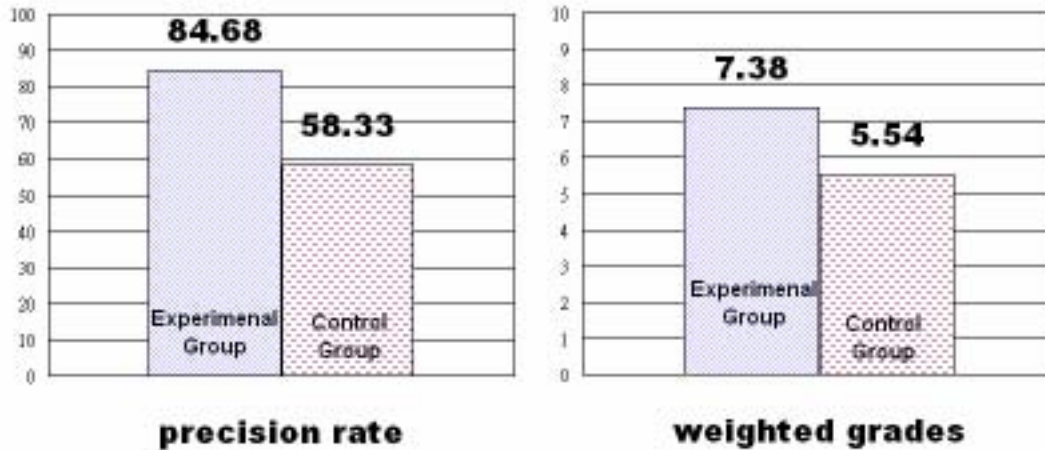


Figure 7. Precision of recommendation

4.2 Convergence Test :

This section analyzes the learning condition and convergence of evolution. In the common case, the model based on GA should perform a large number of generations to observe the converged condition. But it is opposite of the idea that reduces the human fatigue because our model based on IGA needs human being to evaluate in each rounds. For this reason, it is pretty hard to prove the convergence of the IGA. Toward this goal we have attempted to show the change of fitness in following Figure.8 according to the generation and compare the result of the experimental group with the control group.

In order to show the change of learning condition, we let user perform above demonstrating test every round. But the number of the listed items decreases to 10 for fear that the user spends too much time. Therefore, we take the change of the weighted grades got from demonstrating tests in each round as the learning trend of the system.

Fig.8 shows the change of average weighted grades of all users according to the generation. Curve A represents experimental group and Curve B represents control group. We can observe that the Curve A is effectively increased by the user's evaluation, and express the converged condition after 8-th rounds. As we expected before, the curve B can't show the trend of increasing progressively because the control group selects items randomly without training treatments.

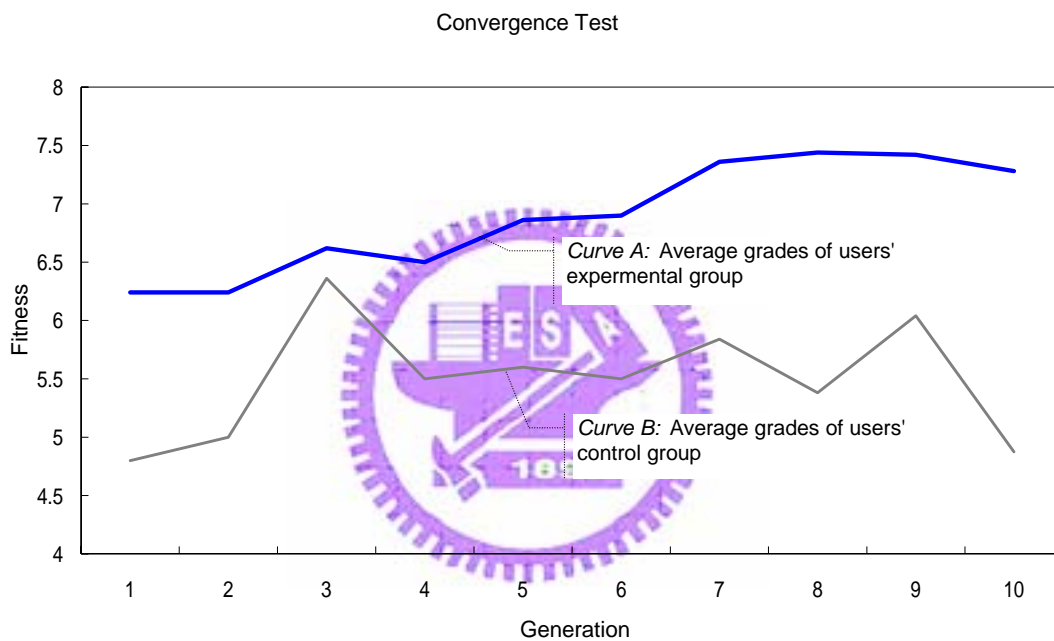


Figure 8 Converge Test

Except for the quality of recommendation and convergence test, we further attempt to compare the music items which are in the experimental group and the control group, and analyze whether the music items of the experimental group are obviously different from the other group. In other words, we try to demonstrate the items which selected by the agents are unique, and to show the agents truly converge to some aspects. In order to achieve this target, we extract the features of all songs in the experimental group and compare with the control group.

To take one user's result for example. As shown in the following Fig.9, the experimental group is obviously different from the control group in *Tempo* feature. The same condition also appears in *Entropy* feature. The example explains the trained agents have converged some unique preferences, and not select items blindly.

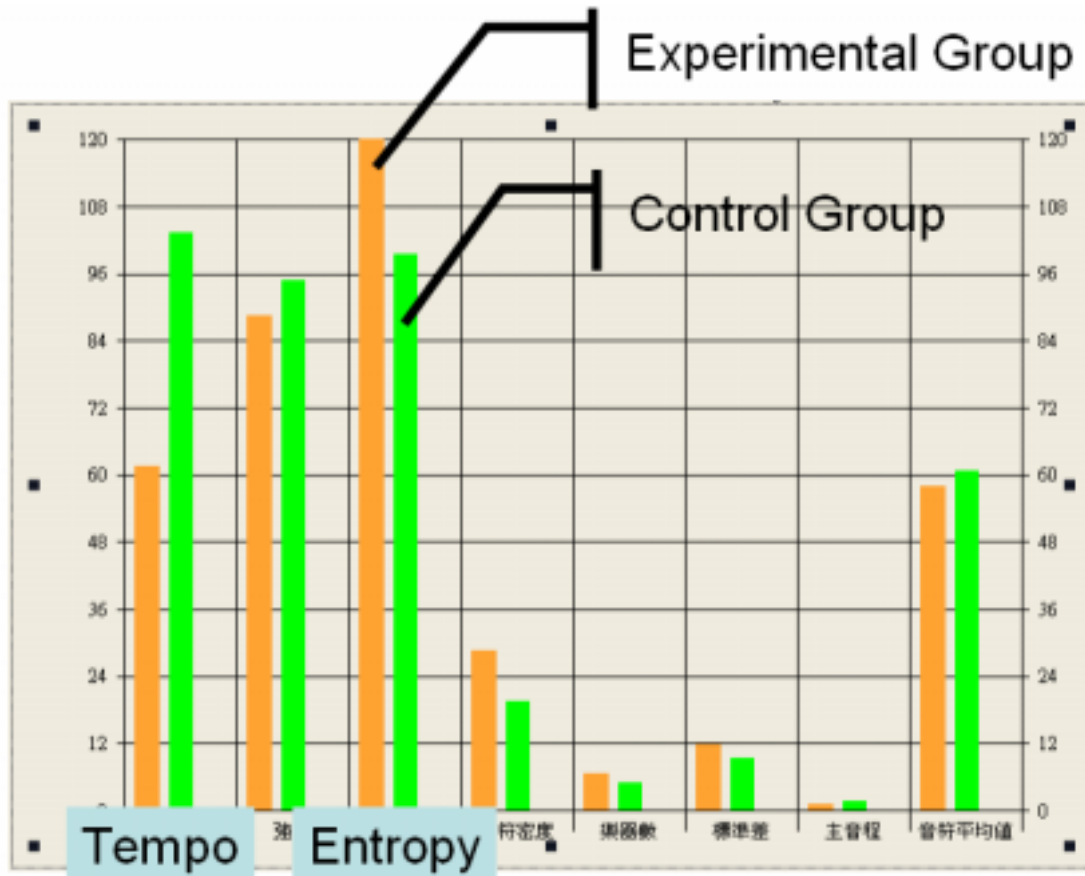


Figure 9. An Example of the user's result

To validating this conclusion more certainly, we also adopt the Chi-Square test of statistics to explain the two groups are independent according to the all features. By compiling the statistics of all users' result, the ratio of the users who could pass the Chi-Square test with 5% alpha level could achieve 46%.

5. Conclusion

This paper presents a model of evaluating subjective objects by the cooperation between human and agents, and implements a personalized recommendation system to demonstrate that the model is workable. In our recommendation system, the user needs to directly train an exclusive group of agents which understand the user's taste. And then the user can use these agents to evaluate a large number of music items.

In order to train the agents of fitting the user's preference, we adopt the procedure of IGA. And this paper presents a concept of the agent fame value to decrease the problem of human fluctuation, which derived from using subjective human evaluation as the fitness function in IGA. Furthermore, the agent fame value is also used as one of the converged conditions to promote the agents population varied and preserve the fine genes in the evolution.

This music recommendation system can be regarded as a basic framework. This system applies to evaluate the other kind of objects by replacing or modulating some function blocks. To take the graph recommendation for example, the system developer can replace the feature extractor with another procedure which can extract information of the graph, like brightness, contrast values, RGB values, and so on. Furthermore, the developer properly modulates the parameters of the evolution module according the properties of the system. Therefore, our recommendation system is also flexible.

Except the personalized recommendation system, our model of cooperation between human and agents can expand to the professional evaluation system according to the purpose of the users. As shown in Fig.10, each user needs to judge the other user's agents besides trains the own ones. Therefore, the system would

recommend the items which the mass population are interested in, no longer fit the single user's taste.

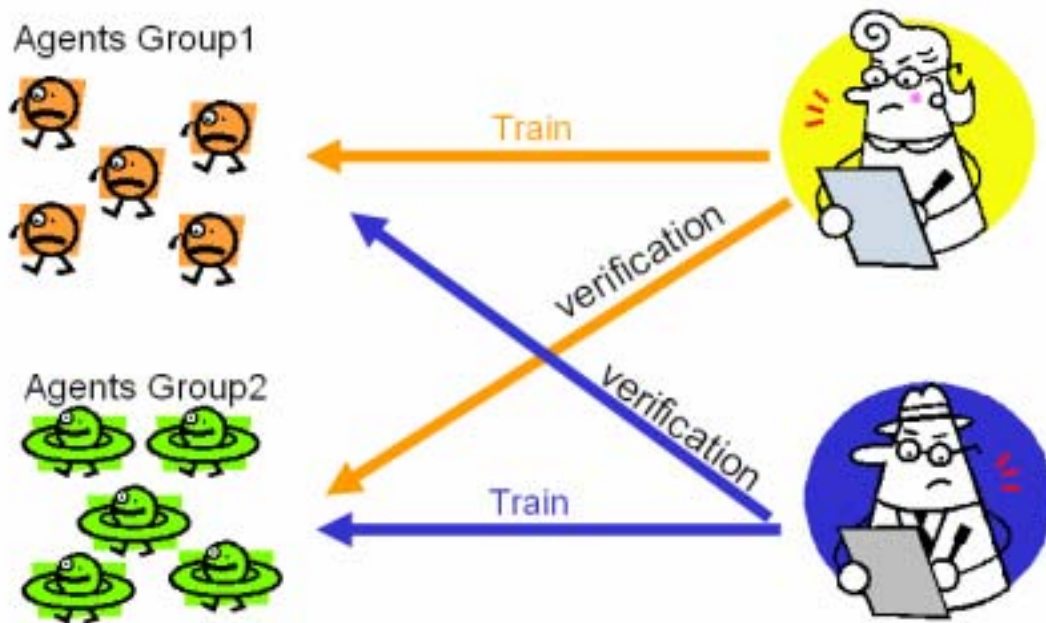
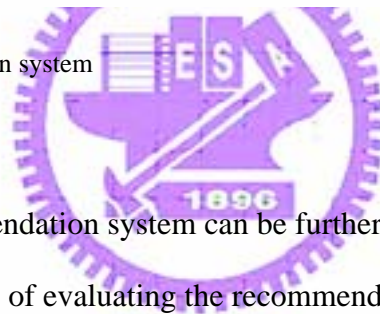


Figure 10. Professional evaluation system



Our personal recommendation system can be further enhanced in some ways. For example, to reduce the time of evaluating the recommended music items, the summarization of music items may be necessary. The system needs more definitions of the features to represent the complex semantics in the music items. Furthermore, the system parameters, like the number of recommended items in each run, the number of agents, or the threshold of agent fame value, these parameters may need more experiments to find appropriate values. Or presenting a method which can modulate the parameters to fit the user reduces the users' time of training agents automatically.

6. References

- [1] U. Shardanand and P. Maes, "Social information filtering: algorithms for automating \"word of mouth\",\" in *Proceedings of the SIGCHI conference on Human factors in computing systems*. Denver, Colorado, United States: ACM Press/Addison-Wesley Publishing Co., 1995, pp. 210-217.
- [2] F.-F. Kuo and M.-K. Shan, "A personalized music filtering system based on melody style classification,\" presented at Data Mining, 2002. ICDM 2002. Proceedings. 2002 IEEE International Conference on, 2002.
- [3] H. C. Chen and A. L. P. Chen, "A music recommendation system based on music and user grouping,\" *Journal of Intelligent Information Systems*, vol. 24, pp. 113-132, 2005.
- [4] M. Balabanovic and Y. Shoham, "Fab: content-based, collaborative recommendation,\" *Commun. ACM*, vol. 40, pp. 66-72, 1997.
- [5] J. Chaffee and S. Gauch, "Personal ontologies for web navigation \" in *Proceedings of the ninth international conference on Information and knowledge management McLean*, Virginia, United States ACM Press, 2000 pp. 227-234
- [6] J. H. Chiang and Y. C. Chen, "An intelligent news recommender agent for filtering and categorizing large volumes of text corpus,\" *International Journal of Intelligent Systems*, vol. 19, pp. 201-216, 2004.
- [7] D. Fisk, "An application of social filtering to movie recommendation,\" *Bt Technology Journal*, vol. 14, pp. 124-132, 1996.
- [8] R. Mukherjee, E. Sajja, and S. Sen, "A movie recommendation system - An application of voting theory in user modeling,\" *User Modeling and User-Adapted Interaction*, vol. 13, pp. 5-33, 2003.
- [9] R. J. Mooney and L. Roy, "Content-based book recommending using learning for text categorization \" in *Proceedings of the fifth ACM conference on Digital libraries*

San Antonio, Texas, United States ACM Press, 2000 pp. 195-204

[10] L. F. Tian and K.-W. Cheung, *Learning User Similarity and Rating Style for Collaborative Recommendation*, 2633 ed, 2003.

[11] M. J. Pazzani, "A Framework for Collaborative, Content-Based and Demographic Filtering," *Artif. Intell. Rev.*, vol. 13, pp. 393-408, 1999.

[12] T. Hofmann, *Learning What People (Don't) Want*, 2167 ed, 2001.

[13] S.-B. Cho and J.-Y. Lee, "A human-oriented image retrieval system using interactive genetic algorithm," *Systems, Man and Cybernetics, Part A, IEEE Transactions on*, vol. 32, pp. 452-458, 2002.

[14] S.-B. Cho, "Emotional image and musical information retrieval with interactive genetic algorithm," *Proceedings of the IEEE*, vol. 92, pp. 702-711, 2004.

[15] H. Takagi, "Interactive evolutionary computation: Fusion of the capabilities of EC optimization and human evaluation," *Proceedings of the Ieee*, vol. 89, pp. 1275-1296, 2001.

[16] P. Maes, "Agents That Reduce Work and Information Overload," *Communications of the Acm*, vol. 37, pp. 31-40, 1994.

[17] D. A. Norman, "How Might People Interact with Agents," *Communications of the Acm*, vol. 37, pp. 68-71, 1994.

[18] J.-Y. Lee and S.-B. Cho, "Sparse fitness evaluation for reducing user burden in interactive genetic algorithm," presented at Fuzzy Systems Conference Proceedings, 1999. FUZZ-IEEE '99. 1999 IEEE International, 1999.

[19] W. Chai and B. Vercoe, "Using User Models in Music Information Retrieval Systems," presented at International Symposium on Music Information Retrieval, Oct, 2000, 2000.

[20] D. H. Widyantoro, T. R. Ioerger, and J. Yen, "An adaptive algorithm for learning changes in user interests," in *Proceedings of the eighth international conference on Information and knowledge management*. Kansas City, Missouri, United States: ACM

Press, 1999, pp. 405-412.

[21] H. Hirsh, C. Basu, and B. D. Davison, "Learning to personalize - Recognizing patterns of behavior helps systems predict your next move.," *Communications of the Acm*, vol. 43, pp. 102-106, 2000.

[22] K. Sayood, *Introduction to Data Compression. 2nd edn.*: Morgan Kaufmann, 2000.



Appendix A

音樂特徵抽取

A.1 Midi 檔案分析

在本研究中採用的音樂物件為 Midi(Musical Instrument Digital Interface) Format。為了實驗中後續的處理，因此我們需要先將系統中所有的 Midi Files 轉成文字檔的形式，以便從中分析出我們想要的資訊。下圖舉例顯示了 Midi 檔里的資訊。



The screenshot shows a Notepad window titled '0_24 - 記事本'. The window contains a table of MIDI data with the following columns: Unit, Length, At, Time, Track, Channel, Note, and Velocity. The data is as follows:

Unit	Length	At	Time	Track	Channel	Note	Velocity
1980	100	2171ms	109ms	T1	C1	d4	56
2116	104	2320ms	114ms	T1	C1	d4	64
2256	112	2473ms	123ms	T1	C1	d4	56
2428	88	2662ms	96ms	T1	C1	d4	62
2580	88	2828ms	97ms	T1	C1	d4	40
2720	112	2982ms	123ms	T1	C1	d4	59
2888	88	3166ms	97ms	T1	C1	d4	88
2924	132	3206ms	144ms	T7	C8	f1	69
3020	112	3311ms	123ms	T1	C1	d4	53
3180	108	3486ms	119ms	T1	C1	d4	55

The status bar at the bottom right of the window indicates '第 9 列, 第 34 行'.

Figure A-1 歌曲庫中 MIDI 格式範例

此圖顯示了本系統歌曲庫中某一首歌的片段資訊，每一行代表的是一個音符，行的排列就是歌曲發音的順序。我們從列出的資料中來大概介紹一下，每一

行的 Length 就代表了這個音符要持續多久的時間，因此可以由此屬性來推算出整首曲子的速度，要是每個音符都很持續很短的時間，那麼整體感覺便會是一手快節奏的歌。Track 則是代表此音符是屬於哪條音軌序列，Midi 檔案可以是多音軌的格式，通常有一主音軌代表著主旋律，其他的則多是和弦，由於主音軌包含了大多數的意義，因此如何抽出主音軌也是一需要考慮的問題。Channel 則是音色的選擇，即這個音符需要用哪種樂器來發音，此屬性可以查詢 GM 音色表來找出音源。Note 代表的則是音高，為整行內容中最重要資訊，而最後的 Velocity 則是代表著此音符的發音音量，數值越高則越大聲。

A.2 音樂特徵抽取

如同本文中提及的，每首曲子轉成可讀性的文字檔之後，系統以下定義的八種音樂特徵來抽取數值：

a. Tempo degree

分析此曲子的速率值

b. Loudness

計算所有音軌的音量強度平均

c. Pitch Entropy

The pitch entropy, derived from Sayood (2000), is defined as follows:

$$PitchEntropy = -\sum_{j=1}^{NP} P_j \log P_j$$

where P_j is defined as follows:

$$P_j = \frac{N_j}{T}$$

where N_j is the total number of notes with the corresponding pitch in the main track, T is the total number of notes in the main track.

d. Pitch Density

e. Mean of the pitch values

計算主音軌中所有音符的平均

f. Standard deviation of the pitch values

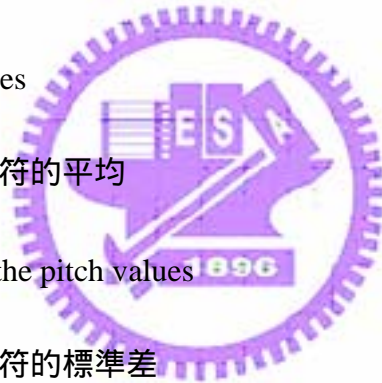
計算主音軌中所有音符的標準差

g. Number of Channel

取出曲子中出現的樂器種類個數

h. Catalogs of pitch interval

不同類型的樂曲的音程分布情形會有所差異，基於這樣的概念我們先計算十種分類的樂曲的音程差異分布情形，作為標準分類。因此有新的樂曲進入系統時，拿此樂曲的音程去和標準的音程種類比較，算出此樂曲近似於哪個種類。



A.3 統計音程表

音程的意思為相鄰音符的音高差距，統計一首曲子中不同音程的次數多寡，可以用以當作樂曲分類的依據。上述的特徵定義其中一項便是分析曲子屬於哪一種音程。

我們分別對 10 種不同種類，每個種類包含了約 50 首樂曲，取出主旋律來計算音程的分布情形，再將同種類的取平均作為此風格的標準音程表，因此我們在系統中便統計了 10 種類型的標準音程。當在特徵抽取時便可以拿新進的樂曲和標準音程表作比對，比對出最相似的類型，以此來當作此首樂曲在[音程種類]的屬性。

以下便是本系統中統計出來的 10 種音程表。

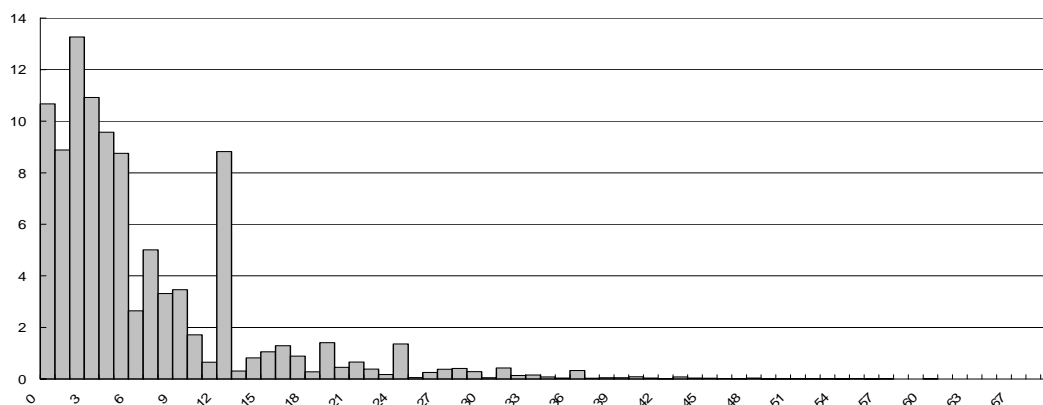


Figure A- 2 音程表:藍調音樂

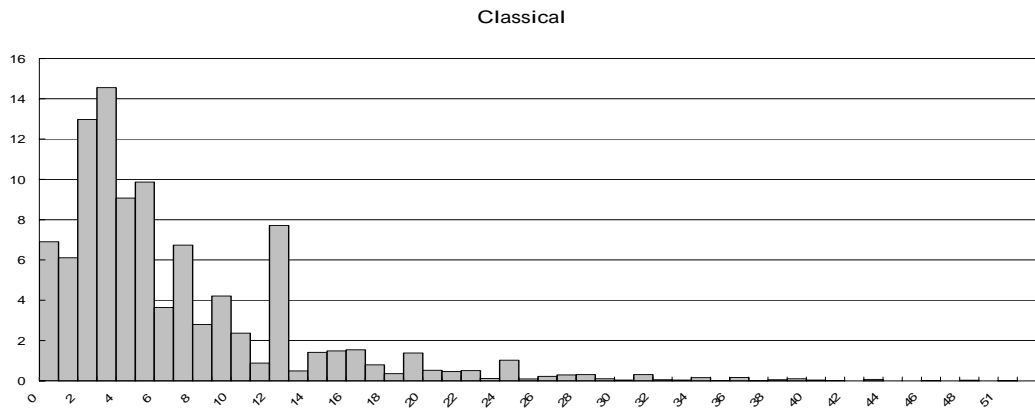


Figure A- 3 音程表:古典音樂

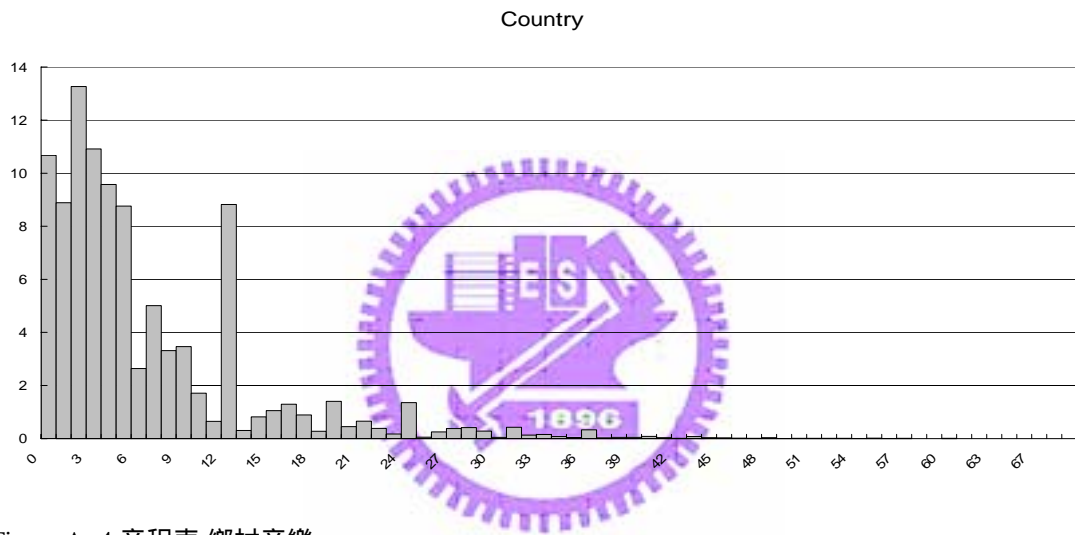


Figure A- 4 音程表:鄉村音樂

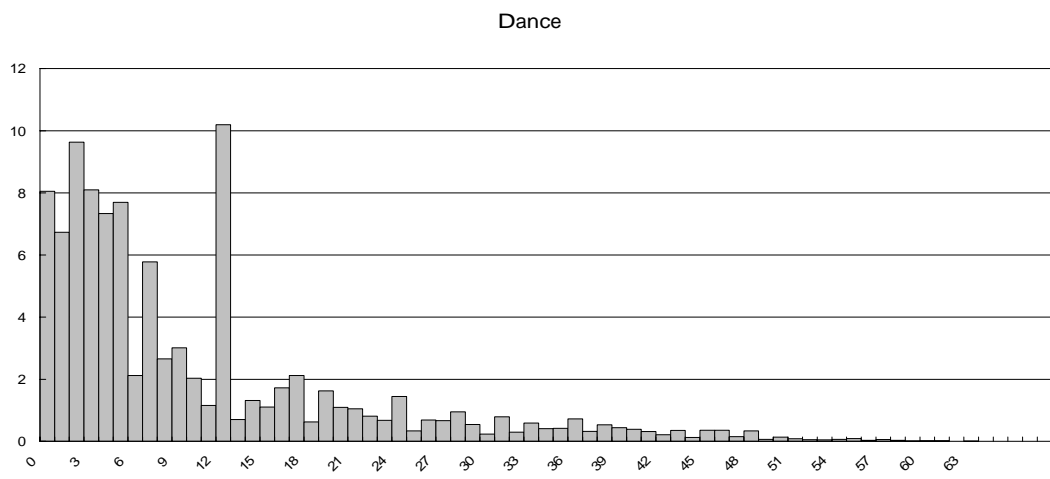


Figure A- 5 音程表:舞曲

Folk

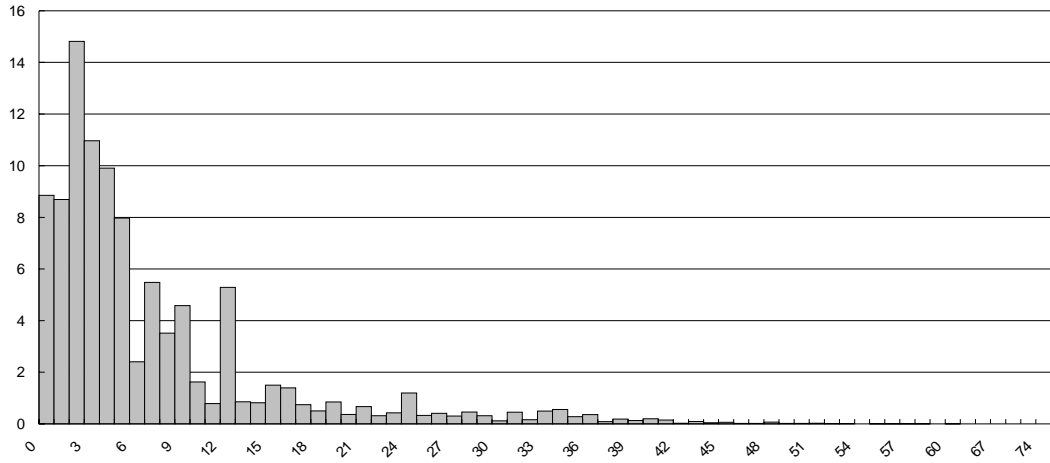


Figure A- 6 音程表:民俗音樂

Jazz

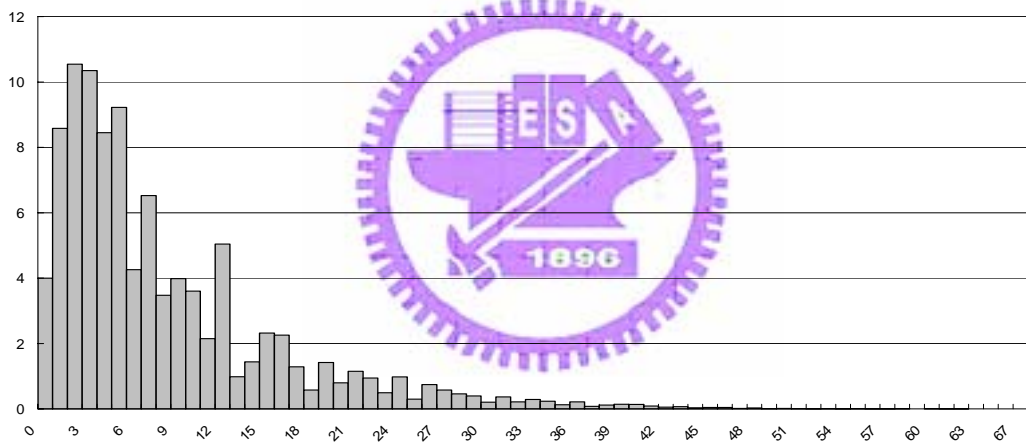


Figure A- 7 音程表:爵士樂

Latin

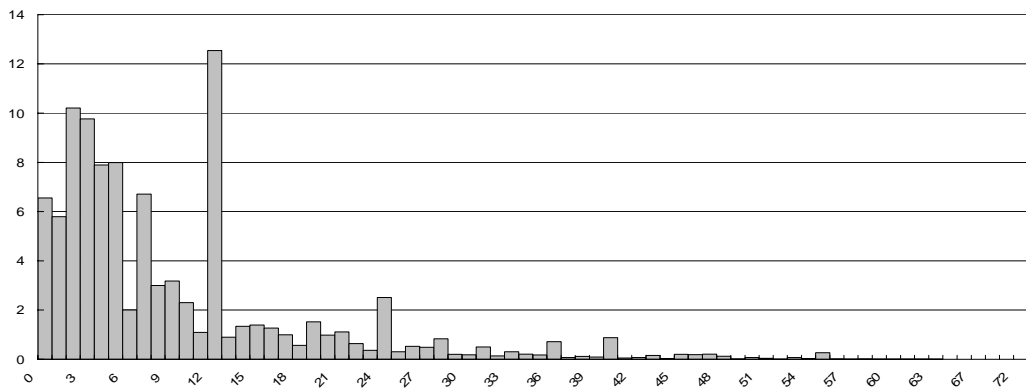


Figure A- 8 音程表:拉丁舞曲

New Age

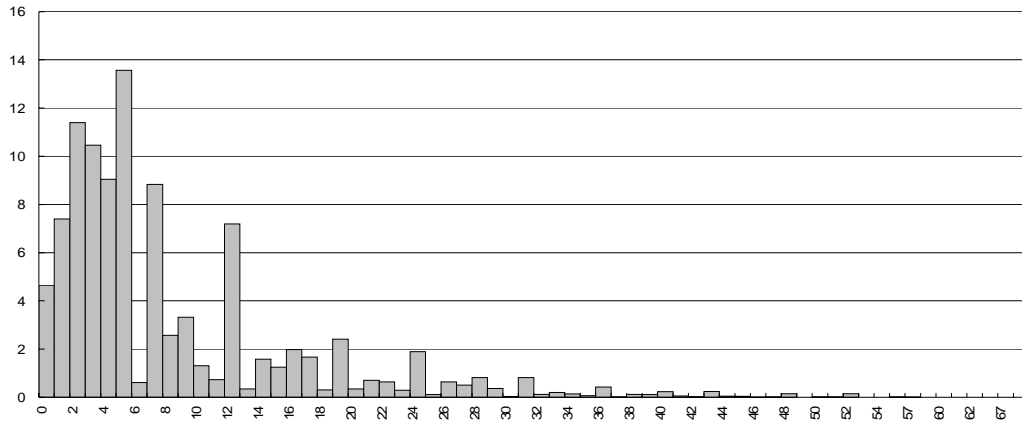


Figure A- 9 音程表:新世紀

Rap

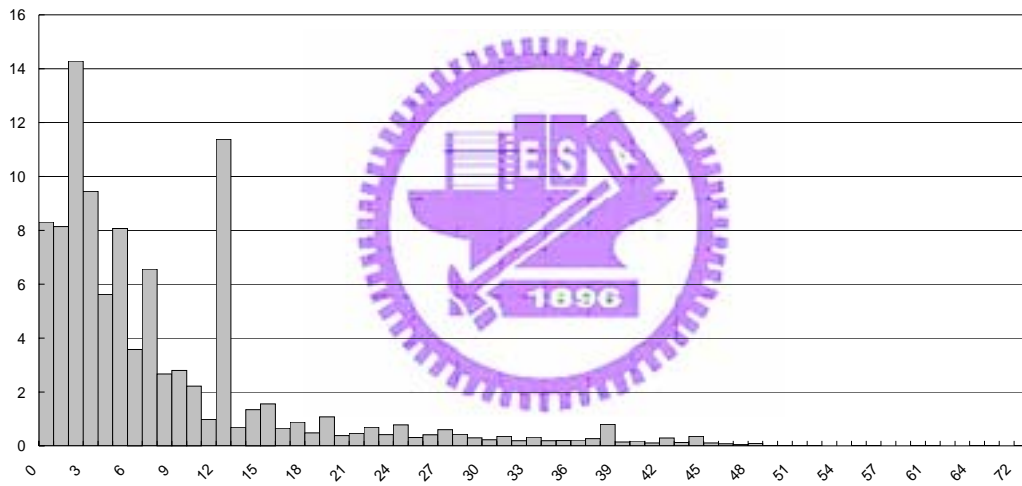


Figure A- 10 音程表:饒舌樂

Rock

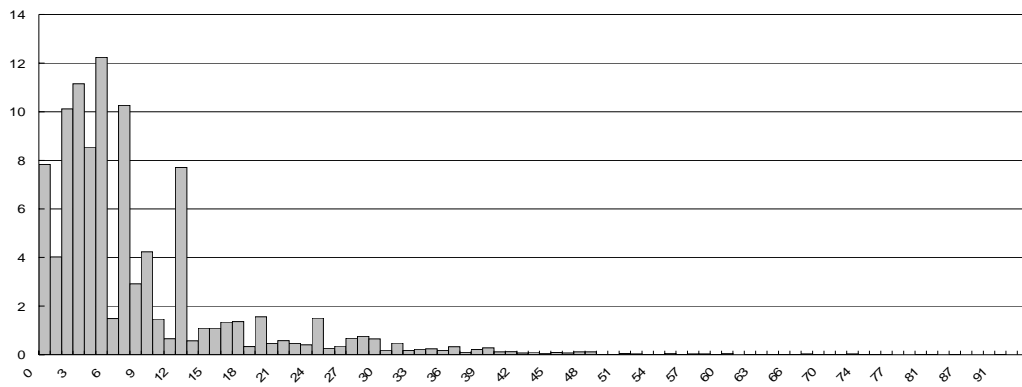


Figure A- 11 搖滾樂

Appendix B

基因演算法相關

B.1 初始分布策略

在基因演算法中，基因的初始值一般都是隨機產生。但是在我們的資料庫中，音樂特徵的分布不平均，因此若以隨機產生的方法來初始基因，那麼產生的 Agent 可能會因為基因的挑選區域太偏離，而造成此 Agent 無法推薦歌曲來代表自己。舉例而言，*Figure B-1* 下圖為本系統資料庫中所有曲子在音樂特徵[Tempo]的數值分布情形，我們可以看到最多的曲子的特徵都聚集在中間約 80 附近，假設系統產生隨機基因對於 Tempo 的比對範圍是介於 35-40，那麼可以比對到的樂曲數便非常少。此類的 Agent 可以視為”品味獨特”，這些 Agents 因為挑不出歌曲來，所以得不到來自於使用者的分數，因此在演化中很快就會被替換或者淘汰掉了，這對於系統而言，是無用的且會延長收斂的時間。因此適當地調整初始區域是有需要的。

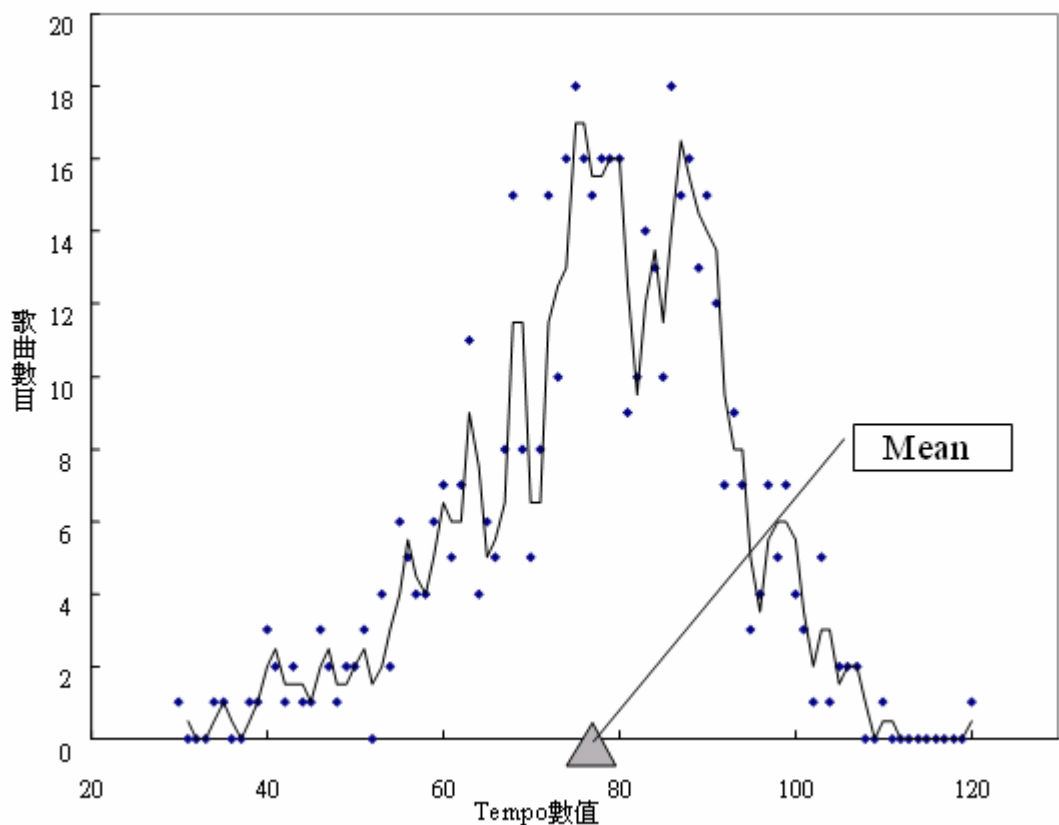


Figure B- 1 音樂特徵數量分布圖



在我們的系統中採用了適當限制初始值區域的方法來減少這種 Agents 的產生，也就是在演化前就已經調整系統，以減少人類評斷的時間。以 Figure B- 2 中 Tempo 特徵為例，此特徵的平均值為 78，標準差為 14，可以觀察到大部分的歌曲都落在平均正負 2 倍標準差之間，且佔了 90% 以上。基於這個概念，系統在初始基因時，會先參考相對應特徵的平均值以及標準差，劃分出幾個區域，基因初始時會有較高的機率落在中間的區域。系統預設值為有 60% 機率會落在平均加減一倍標準差的範圍內，90% 的機率落在平均加減二倍標準差的範圍中。以此方

法適當地調整初始值，以利於後來的演化。

B.2 系統演化流程圖



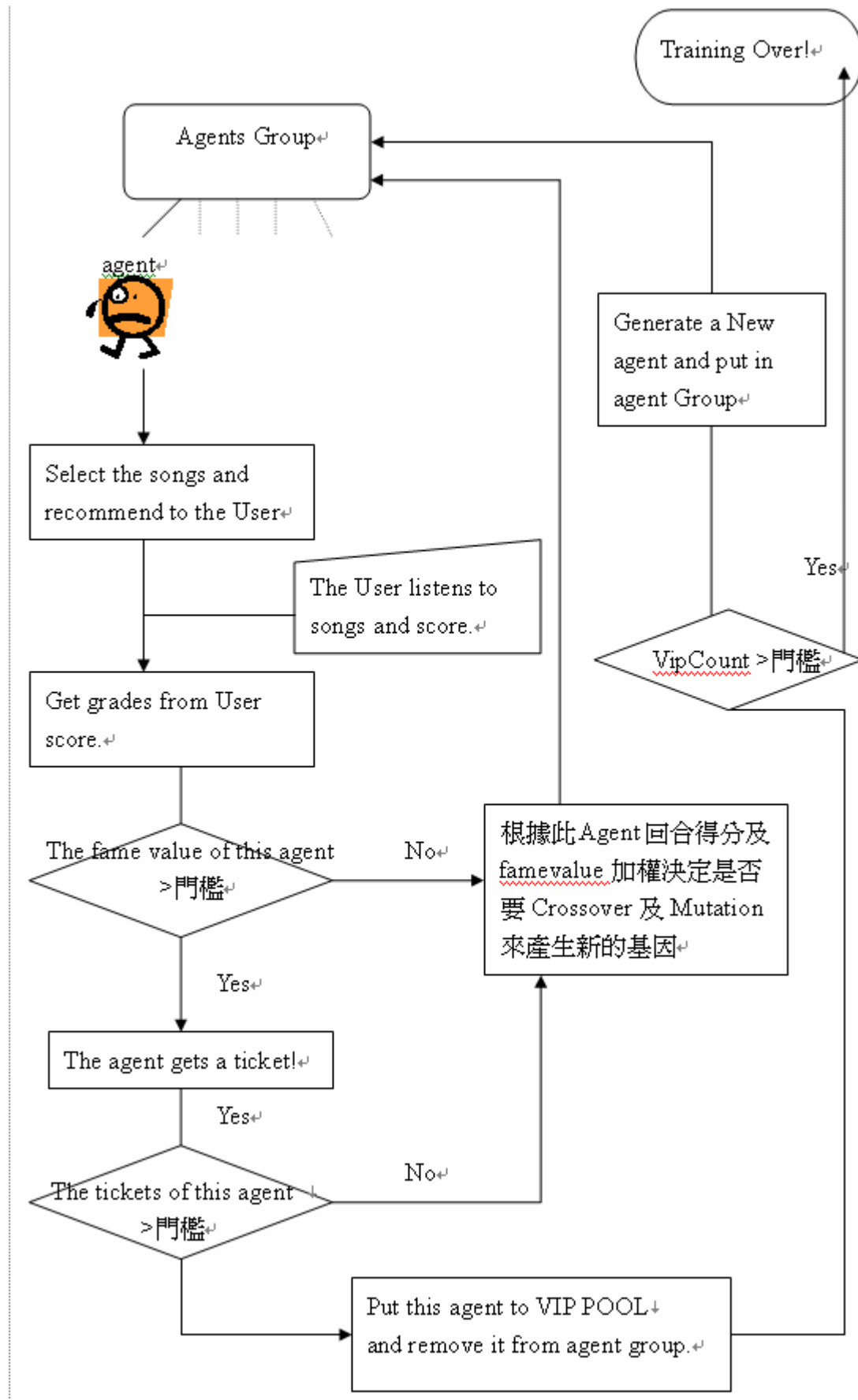


Figure B- 3.系統演化及收斂機制