A Product Recommendation System Based on the Nearest Neighbor Algorithm and Cloud Computing

Chung C. Chang, Professor, Department of Information Management, Chinese Culture University
Gao W. Shiu, Graduate Student, Department of Information Management, Chinese Culture University

ABSTRACT

Product recommendation systems are also called recommender systems in the information industry. This study discusses the application of nearest neighbor algorithm and cloud computing in a mobile phone game recommender system. Mobile phone games are characterized by low development cost and time, and currently have the largest number and market among all electronic games. Searches for mobile phone games are carried out using only keywords and categories, which makes it very likely for users to miss games they might be interested in when searching through such a large number of games, especially when they do not have a specific target and are just browsing for new games. A good production or marketing strategy targets specific customers based on analyses of market or consumer demands. This study develops a recommender system based on the nearest neighbor algorithm and cloud computing using the most notable features of mobile phone games.

Keywords: nearest neighbor algorithm, recommender systems, mobile phone games, cloud computing

INTRODUCTION

When the product being searched for is too non-specific, e.g. a science fiction novel or movie that you have not read or seen before, it is often hard to use keywords or categories to find what you are looking for. Items for the masses (bestsellers) will take up the majority of the results page, making it hard to notice items that target niche markets (high levels of customization). When the number of results is too great, it is very unlikely that users will look through hundreds or even thousands of search results page by page.

Mobile phone games are easy to develop and have a low barrier to entry. They have become the mainstream in the market due to the proliferation of smart phones and increasing internet penetration, and the increasing number of developers has driven a rapid increase in the number of mobile phone games available in the market. Short product life-cycles indicate higher demand on new games, and low barrier to entry coupled with internet penetration has created a mobile phone game market with the long tail, a term popularized by Chris Anderson. The long tail means that the internet provides a vaster scope of scales, i.e. in the past, developers only focused on popular products in the top 20%, but now 80% of non-hit items all have value. Developers no longer need large capital and the support of a considerable consumer base, and can maintain their revenue by rapidly launching small products. When conducting searches on a website or online shop, the results page is often occupied by the most popular items or new items, moving other items down the page. Most products have a market but are not adequate to be listed at the top.
This study employs the nearest neighbor algorithm to recommend phone games, and matches data input by users with target customers of mobile phone games to recommend the most suitable to consumers. This study aims to increase the accuracy of search results in recommending the most relevant games, and utilizes the accessibility of cloud computing via the internet.

A good production or marketing strategy targets specific customers by analyzing market or consumer demands. The nearest neighbor algorithm is a classification method that uses a vector space model. The positioning of phone games and customer preferences are features, for which the similarity of their values is calculated, and the closest item in the space to the sample is found on this basis; when items have multiple features, all of the features are considered. Cloud computing is a computing method based on the internet, and is both flexible and expandable. The system is placed in the cloud and users can access resources or services via the internet. Users do not need to know how the system operates or the specifics of the cloud. All they need is access to the internet to gain the services they need.

This study emphasizes the direct use of data provided by users to recommend mobile phone games. The term mobile phone games refer to games played using mobile devices, e.g. smart phone or tablet PC. Features of mobile phone games are distinct items identified in previous studies. Items that appear on GOOGLE PLAY and APP STORE represent the two major mobile phone games: Android and iOS.

Recommender Systems

When facing a decision in the immense amount of data we have today, referring to the opinion of an expert or someone experienced is a good approach. After considering the opinions of people who are familiar with the product or widely acknowledged experts, consulting with peers, or reading reviews on newspapers and magazines, it is a good idea to choose products with good reviews and are popular. People may not be able to easily find substitutes based on their personal preference and experience (Terveen & Hill, 2001), e.g. people who like to travel also like cuisine.

A recommender system can help users search for and filter data. It uses multiple information and computation methods to help people combine their experiences and jointly make a good decision. This is an exchange of personal experience that can be carried out anonymously with strangers (Terveen & Hill, 2001). The recommender system displays results similar to a search engine, but emphasizes the fit of results with the user and not the keywords.

This study develops a mobile phone game recommender system that focuses on user preferences, and recommends items that users are interested in. Users will choose mobile phone games using this recommender system instead of directly searching for games in the database.

Collaborative Filtering

Collaborative filtering is commonly used in e-commerce, and is especially important to large internet companies, since rapid and accurate recommendations provided by the recommender system will attract customers’ attention (Su & Khoshgoftaar, 2009). This is the main recommendation method of this study, using user preferences and mobile phone game features as corresponding features for recommendation.

The greatest advantage of collaborative filtering is personalization (Terveen & Hill, 2001). Users can learn about themselves through their nearest neighbors. It is possible to give recommendations based on personal preferences without learning about all items. The recommender system uses collaborative filtering to match data input by suppliers and users to make recommendations, and emphasizes the calculation of features and matching results. Preferences of many users can be weighted when making predic-
tions of a user’s preferences, which is often necessary to find items that are indeed recommendable (Ter-
veen & Hill, 2001).

Using collaborative filtering, however, will usually encounter three major issues, namely cold start, scalabil-
ity and sparsity (Suzuki & Arikawa, 2007). Cold start refers to the difficulty of effective analysis when there
is insufficient data; scalability emphasizes the large amount of computing resources required for analysis;
sparsity refers to the fact that one customer usually does not buy too many types of products in contrast
with the variety of products offered at a sales location, i.e. even though a large amount of data is collected,
the amount of data collected for individual products is still insufficient.

Cold start and sparsity are issues that will always occur with collaborative filtering. The system is
usually used in a massive set with many types of items, and calculations are carried out an extremely
sparse situation. When a new user joins, it is often hard to find similar items due to the lack of data, mak-
ing it hard to make recommendations. When users without features or very few features join the system,
it will have an adverse effect on recommendations (Su & Khoshgoftaar, 2009). Recommendations require
large amounts of data, but requirements analysis for any company of considerable scale often exceeds the
storage and processing ability of a single machine, causing the issue of scalability, which can be resolved
via the internet and cloud computing.

Content Filtering

Content filtering is a technique that matches pre-defined features with customers’ preferences. Recom-
mender systems using this approach require customer preferences and historical data. The key to this
approach is extending choices made by a customer in one item to other items (Brusilovsky, 2007).

Since relationships are pre-defined, once the relationships are in place, systems that use content fil-
tering only need to receive input from users (whether it may be clicks on an ad or preferences manually
input) to directly calculate results. There are less restrictions in the use of data and making recommenda-
tions compared with collaborative filtering. Therefore, content filtering can serve as an alternative when
collaborative filtering is too difficult (Basilico & Hofmann, 2004).

Structured or semi-structured data is used for analysis in content filtering; user data is not directly
used for comparison. This approach is more like performing searches based on content and not filtering.
In terms of machine learning, content filtering is similar to executing a classification task, classifying us-
er based on specific rules. Due to the sparsity of data, complex algorithms cannot show their value
without sufficient data, and a simpler method must be used (Burke & Felfernig, 2008).

Hybrid Recommendation

After recommender systems using individual approaches were established, the development of hy-
brid recommender systems made great contribution by providing more possibilities and resolving dilem-
as of systems that use a single approach (Burke & Felfernig, 2008). Hybrid recommender systems use
multiple approaches to make recommendations, e.g. adding general perception or statistical results when
determining features, and achieve the effects of both collaborative filtering and content filtering. Fur-
thermore, hybrid recommendation systems can easily combine multiple data sources, and more and more
studies are showing that hybrid recommendation systems can improve the efficiency of recommendations
and resolve issues with systems that use only one approach (Adomavicius & Tuzhilin, 2005). Hence,
making suitable choices and combining technologies and knowledge is an important topic when it comes
to hybrid recommender systems (Burke & Felfernig, 2008).
Cloud computing is an internet-based computing model that has been widely applied due to its convenience. Cloud computing enables computing resources to be rapidly allocated and a wide variety of services to be provided based on user requirements (Mell, 2011). The concept of cloud computing has existed since the development of the internet (Qusay, 2011). Cloud computing is needed for its services, regardless of the level of cloud facilities, users choose them for the services they provide and not the arrangements in between (Hamdaqa & Tahvildari, 2011).

The cloud is an environment where users only need an application or browser to access services through the internet. Users are neither required to understand the arrangements of the cloud, nor download or update the system, making services easily accessible and scalable. From the perspective of enterprises, these advantages of cloud computing enable them to flexibly utilize resources and make production activities more cost effective. The rapid allocation of computing resources allows deployment and upgrade to be completed within a short amount of time, and planning for changes in requirements in advance is no longer that important (Riley, 2012). Therefore, we placed the recommender system in the cloud in the form of a webpage, effectively resolving the scalability issue and making the system accessible at all times.

**Nearest Neighbor Algorithm**

The nearest neighbor algorithm uses features of items as dimensions to form vectors. It is a simple and fundamental algorithm that is also applied in machine learning, and can be used for developing unknown items or items that are hard to verify (Peterson, 2009). The recommender system of this study adopts this method for implementing collaborative filtering.

Classification calculations using the nearest neighbor are carried out under the assumption that all groups are in a stable state (Kozma, 2008). A distance formula (e.g. Euclidean distance) is used for calculating the distance between individual objects and groups of objects, which is further used to calculate the similarity of each item and object, using this as the basis for assessment or classification. There is usually an unknown object being measured and several known objects (e.g. data stored in the database) for comparison (Lammertsma, 2007). There are many variations for calculating this distance, e.g. agglomerative nearest neighbor and k-nearest neighbor algorithm, which have the following features (Peterson, 2009):

- Calculates the geometrical distance between samples
- Adjustable feature values for calculation
- Form decision rules and confusion matrix
- Cross validation

Similarity is the standard for assessment and classification in the nearest neighbor algorithm. Similarity is related to the distance between objects in the vector space. Either linear distance or linear discriminant analysis can be used for calculation. This study uses linear distance, which can be indicated in absolute distance or Euclidean distance (Lammertsma, 2007).

This study calculates the Euclidean distance between objects, which is calculated as shown in formula (1). Since the feature axes have different scales, it is inappropriate to use the distance between their original values, so values must first be normalized for calculations. Normalization is achieved by dividing the original distance by the maximum difference of each feature.

$$d_{euclid}(x,y) = \sum_{i=1}^{n} \sqrt{|d_i(x,y)|^2}$$  \hspace{1cm} (1)
MOBILE PHONE GAMES

Mobile phone games are software that can be executed on a mobile operating system, and are usually playable at any time the same as mobile devices. From users’ perspective, mobile phone games are simple, easy to play, and do not take too much time. From the perspective of developers, there is a low barrier to entry because mobile phone games are easy to develop. Even though traditional mobile phones may come with simple games in them, these games are not comparable to the games available on smart devices today (Haggerty, 2012).

The current commercial environment for mobile phone games was formed because of two reasons: 1. Better performance of handheld devices and 2. Popularization of cloud services and the internet (Haggerty, 2012). Better performance of handheld devices enables more complex processing, and the convenience provided by network platforms makes it easier for users to choose and acquire games, also providing a distribution channel for developers.

Business models of mobile phone games mainly include pay-per-download, subscription, free-to-play, advertising, and a fifth model that appeared later on – freemium. At first, mobile phone games were mainly built-in games that were free-to-play (Behrmann, et al, 2012). Studies show that current mobile phone games mainly adopt the advertising and freemium models, in which freemium has become the main source of revenue for developers in recent years (McDuling, 2014).

An investigation report of the Market Intelligence & Consulting Institute, Institute for Information Industry (Institute for Information Industry, 2014) pointed out that consumers are affected by the following factors when choosing a mobile phone game, from highest to lowest the factors are: recommendations by friends, selected games in the app store, gaming communities, and general communities; consumers are further affected by the price (or free), ratings by the app store, and ratings by other users when downloading or buying a mobile phone game.

The theory of market segmentation was proposed by Wendell R. Smith in 1956, and it has been an important theory in economics and marketing since then. Feature selection in this study is based on market segmentation. Mobile phone games are being rapidly developed and in great variety. The market has grown into a massive market as the number of customers grew, and demands of customers have become more diverse. By classifying customers based on the concept of market segmentation, products and services can target a specific segment with the same demands. Different methods have been used for market segmentation, but are basically based on geographic, behavioral and demographic variables. Table 1 lists features of phone game users that correspond to each type of variable (Wei, 2014). Demographic variables are the most important and direct in market segmentation for simple products, because data for such variables are usually the easiest and first to be collected (Riley, 2012). However, this may not be true in all markets, and data of demographic variables for mobile phone games in this study is hard to gather.

Table 1: Segmentation Variables and Related Features

<table>
<thead>
<tr>
<th>Demographic variables</th>
<th>Sex, Social circles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Behavioral variables</td>
<td>Gaming experience, Operating system, Playing time of users</td>
</tr>
<tr>
<td>Geographical variables</td>
<td>Area</td>
</tr>
</tbody>
</table>
MOBILE PHONE GAME RECOMMENDER SYSTEM

The system has four functions: users input data, the system reads the input, similarity is calculated, and results are output. Euclidean distance is used when calculating the similarity between items and user data.

The user input function allows users to provide data required for calculations, e.g. personal information and preferences. The system then reads data from the database and compares them with data input by the user; data is normalized before comparison to ensure that each feature has the same influence. Similarity calculations are carried out using the nearest neighbor algorithm and then results are saved. The output function receives data from the similarity calculations and displays corresponding data to users.

Features used in this study are items that reached the level of significance in previous studies, and are selected based on the theory of market segmentation. Features are divided into three levels, namely direct, observation, and deduction. The three levels represent the approach for determining feature values. The same feature may belong to a different level in different games due to the amount of data available, and there are also gray areas between levels.

The recommender system prototype is implemented using MySQL and PHP, and an additional file system is used to store data. The database stores feature data, and basic information of games (e.g. images and descriptions) are stored as file paths or indexes.

At the input interface, users provide various attributes and weights for calculations. Users are required to fill out all of the items on the interface, and each item has a default weight and feature value. The default weight is the basic weight for recommendation; user data is set at the middle of five levels by default, but when an item has the lowest weight (irrelevant), then the item is ignored. The weight in content selection is divided into five levels, the default level is ignored in calculations, so users must at least raise or lower an item by one level for the content to influence recommendation results. When users select dislike for a specific item, the distance between the item and the user will increase when calculating neighbors.

Figure 1: An Example of Input Screen of the System
The results are a series of recommended games displayed in one line on the webpage according to the order they are recommended (Figure 2). Only the top five are listed here and the most recommended result is placed at the top. The distance is indicated for each item. Users must click on the icon to link to a detailed description in the system.

![Figure 2: An Example of Output Screen of the System](image)

**CONCLUSION**

This study selected features and implemented a product recommendation system using the nearest neighbor algorithm. Factors that influenced users the most when selecting mobile phone games were social circles and gameplay, both are hard to evaluate. iOS and Android users had different preferences. In past studies, researchers have been the most interested in user behavior, which is the most valuable variable in market segmentation, but also the hardest to verify. Even though gameplay had a lesser effect than social circles, it is the most apparent and direct variable when users search for mobile phone games. Traditional game categories include role play, adventure, racing and music. Due to the different requirements and playing time of users, mobile phone games are positioned differently from traditional computer games or games on other devices, resulting in different gameplay. Besides role play, adventure, racing and music games, there are also puzzle games, which account for the largest percentage of mobile phone games.

Back-end feature calculations of the recommender system include parts that affect weights and are unsuitable for users to input data. Even though the price of mobile phone games will show differences in certain user preferences (e.g. supporters of specific operating system or type of game), with freemium as the current mainstream, a very low ratio of users are willing to and actually pay for games, so price is not an item suitable for user input. Features that are an especially good match with specific gameplay, and users of different operating systems with different preferences all influence the weight of items during recommendations. In the complex calculations of 22 features, cloud computing enables effective processing and resolves the issue of scalability, increasing the system’s usability and availability.

This study recommends that future studies focus on social circles and gameplay, both of which have considerable influence on mobile phone game selection but are hard to evaluate. Future studies can establish an evaluation method and utilize feature combination and feature augmentation, or collect large amounts of data for collaborative filtering.
REFERENCES


