Socially Collaborative Filtering: Give Users Relevant Content

What You Will Learn

This paper introduces the concept of collaborative filtering, which offers improved recommendations that are targeted to the individual.

In contrast to traditional collaborative filtering, socially collaborative filtering is based on socially relevant gestures that provide greater insight into how users perceive content. These gestures inform a list of personally interesting content and personalized recommendations that are more relevant to the user.

Applying what has been learned about one item to a new item builds a similarity relationship, which can be used, along with socially relevant gestures, to reduce the time it takes for new content to be recommended.

Introduction

Broadband adoption and the digitization of content are empowering consumers and fundamentally changing the entertainment experience. The number of entertainment choices and delivery methods has grown dramatically due to the digitization of content. As a result, users are now faced with the challenge of discovering content that interests them, while at the same time finding ways to connect to that content. Users want to have a more personalized experience where they can interact with content and other users.

Users want an engaging web experience that is both relevant and interesting for them. Given the wide variety of content available on any one website, it is not reasonable to expect the user to have to pinpoint the content which may interest them, nor can all users be expected to be interested in the most popular content. The situation demands a recommendation system that takes into account both the needs of the individual user and the combined effect of other people who have similar interests.

Traditional Collaborative Filtering

*Collaborative filtering* is the process of filtering for information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources, and the like. Applications of collaborative filtering typically involve very large data sets. Collaborative filtering methods have been applied to many different kinds of data including web 2.0 applications where the focus is on user data.
The standard approach to making recommendations to a user in order to encourage them to buy a product is through a form of collaborative filtering in which the system tracks all the items a user touches. The resulting database of 1-to-1 relationships between a user and any piece of content is easy to update and quick to access. The system may also keep track of the relationship for items a user has viewed as well as bought.

When another user views the item, the system can then find a list of all users who have also viewed the item, and then for all of those users it can generate a list of all the content that those people have also viewed. By summing up the number of times an item appears in the second list, it is easy for the system to generate a list of the most popular items related to it. In this way, the system quickly generates a list of items which have been most popular in the past.

However, this simple approach is not without problems. The first has been called “the Harry Potter problem”, meaning that any extremely popular book will show up on the list for any book in any genre, reducing the effectiveness of the recommendations.

The second and more difficult problem is the time it takes for a meaningful recommendation to be made. In order to make any reasonable recommendation for an item, a minimum number of relationships must be established between that item and users. This is typically solved by introducing another mechanism by which a user can discover an item—either through search or some other directory.

Unfortunately, the user must navigate through two different mechanisms to find their content just to satisfy the algorithm.
The third problem is that every new item, no matter how close to an existing item, must go through this learning curve before it too can be recommended. As a result there is a delay from the time the content is introduced until the content can be recommended.

Lastly, and most importantly, collaborative filtering makes the same recommendations to everyone who views an item. While this is useful for the majority of users, it ignores the differences in interest that many people have.

**Socially Collaborative Filtering**

In order to produce a set of recommendations more targeted to the individual, it is necessary to have a richer understanding of how the user interacts with the content. A user can take a range of actions on any piece of content, from strongly positive actions such as creating the content or giving it a very positive rating, to negative actions where a user provides a negative comment about the content. These actions are called *socially relevant gestures* (SRGs) because they provide insight into how a user perceives the content.
As in traditional collaborative filtering, socially relevant gestures are stored in a database that tracks the relationship between an individual and the content. However, the type and strength (positive, negative, or neutral) of the gesture is also stored, and this information can be used in conjunction with other SRGs to develop a prioritized list of content and people as they relate to other content.
To understand how socially collaborative filtering works, consider the content related to $C_1$ and $C_2$ in Figure 5. Imagine that this content consists of a couple of video clips, and the task is to find other related clips to show. Starting with these two content items, all of the people must be discovered who have expressed an interest in that content. Because these gestures have relative weight, the values for the gestures can be summed up and an ordered list of people can be produced from $P_1$ to $P_n$ where people at the top of the list expressed the most positive gestures toward the content and people on the bottom expressed the most negative gestures.

Given this ordered list of people who are related to the content, all of the gestures that the people expressed toward other content can be examined, again using the relative weight of the gestures to generate a prioritized list of Related Content $C_{r1}$ through $C_{rn}$. Note that in this case, the people who disliked the original content and also disliked another piece of content affect the priority of how results are weighted in the list. This differs greatly from collaborative filtering (Figure 2) in that it is now clear which content is inversely related to the original content because of the gestures.

**Personally Interesting Content**

Finding the list of related content using SRGs is a good start, but it does not personalize the content to an individual. The same list is produced, no matter who is looking at the content. To address this issue, a list of **personally interesting content** must be generated that covers all of the content that may be interesting to a user—either positive or negative. Given SRGs, a list can be constructed of both content that the user should like and content that they don’t like.

Figure 6. Generating Personally Interesting Content
Figure 6 shows how these relationships can be used to first generate an ordered list of all the content in which a user has expressed an interest ($C_1$ through $C_n$). Then all the other users who have expressed gestures toward that content are found and ordered to produce a list $P_{r1}$ through $P_{rn}$ of people who range from liking content that the user liked all the way to disliking content that the user liked. The gestures that these people have expressed toward all other content make it possible to generate the complete list of personally interesting content $C_{i1}$ through $C_{in}$ for the original user.

Several useful observations can be made about the resulting list. First, it is extremely large and thus could take some time to generate. However, because of the nature of the large number of gestures that went into forming it, the list does not change very much over time with only minor adjustments to the order of individual items. This slow rate of change means the list can be calculated less frequently and cached for multiple operations.

From the list, content can be recognized that should be avoided due to its negative relationship to the user. Even more useful is the content in the middle, such as $C_{i3}$, which can be used as an opportunity to discover more about a user’s tastes when they have gone through much of the content that it is known they like.

**Making Personalized Recommendations**

Given the personally interesting content, the mechanism for finding related content can now be refined in a way that is unique to each user. When two people come to the same place to look at some content, it is preferable for the recommendations related to that content to be different if the users have different tastes. In effect, the goal is to distill the *relevant content* from the related content. Figure 7 illustrates this process.
A list can be generated of related content for any content—for example, a video playlist. Then, from the user’s personally interesting content, an intersection of prioritized content can be generated. Next, business rules can be applied to that content, such as age and location restrictions. A user history can also be applied to that filtered content to drop out any content which the user may have recently seen. The combination of business rules and user history makes it possible to generate the list of relevant content that is unique to the user.

If a user has seen all the filtered content, an empty list could potentially occur. To accommodate this scenario, the list can be pre-seeded with content that is not expected to be filtered, and content can be added back when the list is found to be empty.

The result is that recommendations are now both related to the content and relevant to the end user.

Eliminating the Learning Curve

One problem that SRGs do not solve is reducing the time it takes for new content to be recommended. This process can be expedited by taking advantage of the fact that new content that is added is often similar to content that is previously known.

Figure 8. Improving Quality of Results

Applying what has been learned about an original item to a new item that is similar, what is already known about that item can be utilized to produce immediate recommendations for the new item. To do this, first a similarity must be determined between one item and another. This can be done in a number of ways. When there is a significant amount of metadata about an item, the metadata can often be compared to discover how similar it is. If the content is episodes of a television series, it can certainly be assumed that a new episode will have substantially similar content to any previous episode.

Other content may require closer inspection to determine its similarity. This inspection can include technology such as image, text, and even semantic analysis. Many companies are now working in this area with video, music, and even text snippets. Given this analysis, a similarity database can be constructed to show the relationship between any content item C and a corresponding item C_s.
This similarity relationship can now be used to build the related content for any new content by using the SRGs for the similar content as shown in Figure 10.

Attention must be paid to the minimum number of useful interactions before a recommendation can effectively be made. Assume the minimum number to be 10 interactions. Starting with no interactions to determine the related content, 100 percent of the gestures associated with the similar content is used. Once a single gesture is received, the dependency on the similar content can be reduced to 90 percent. As more gestures are received, dependency on the similar content is gradually reduced until the recommendation is based 100 percent on the actual gestures.

The same technique can be applied to populate the personally interesting content, as shown in Figure 11.
To do this, whenever new content has been added which lacks sufficient gestures and which matches against a user’s personally interesting content, the new content can be inserted into that list. However, the new content is inserted significantly lower in the list than the actual content. This will allow the new content to be available when there is no more closely matching content.

**Conclusion**

As the number of entertainment choices and delivery methods has grown dramatically, user behaviors and expectations have changed. In the face of overwhelming choice, users now have the challenge of pinpointing content of interest to them and learning how to connect with that content. Through the analysis of socially relevant gestures, it is possible to provide recommendations to users that are both related to the topic at hand and of particular interest to them.

**For More Information**

To find out more about the Cisco Media Solutions Group, go to [http://www.cisco.com/go/media](http://www.cisco.com/go/media).