GiftGiver: The Gift Recommender

AAI – Instructor: Prof. Jane Hsu

Weiti Kuo, Penn Su, George Chang
{r99922003, r99922157, r99944002}@csie.ntu.edu.tw

Introduction

Although picking gifts for your best friend or your beloved family member are so common in daily life, often people had been troubled by it. People had been overwhelmed by the number of choices given by the commercial retailer stores and e-commerce websites, however, most of the gift recommendation websites aren't good enough for the general public because most of the gifts were cliche or repetitive there are also no common sense in those recommendation websites.

There are many factors to affect final decision. As a sender, budget is a big issue. Even the sender spends a lot of money preparing a gift, the receiver may think it too expensive and embarrassed to receive it. Under tight budget, sender might consider the receiver's interest, religious, gender, age, and the relationship between the receiver and you. In addition, the date to gift is also a very important factor. Under normal circumstances, you do not send a coat in a hot summer.

How to find a suitable gift? In the past, you go to a department store, and try to find the ideal gift, browsing through one store after the other, but nothing seems perfect. If you are not a decisive one, the process is time-consuming. At present, there are many online website to recommend gifts, e.g. Gift ideas, Hallmark Cards, Amazon gifts, Find gift, Yahoo! Gift Finder, JCpenny, Macy's, Sears[Reference]. However, according to our observations, most of them cannot supply satisfactory service. Usually, gifts websites prefer using best seller to be their recommendations. The gift website usually comes out the similar recommendation list event we sent the different preferences, occasion and receivers age, and the list has too many details.

Some people ask others' opinions when they are picking a nice gift, but gift giving should kept secret to the receiver, the gift might not be a surprise when everyone knows about it.

Based on the above reasons, we explored a system we called "GiftGiver" to recommend a list of gifts to help senders. In order to figure out what should be a nice gift, we followed the idea of Scenario-Oriented Recommendation [8]. We analyzed each gift into 8 features. GiftGiver come out a concept of the gift idea, such like flower, tea mug etc. In the way that sender can have their own gift choice. If sender has no idea about a particular item, our system can also gives sender the detail of the gift given from other e-commerce websites.

To give more variety of the recommendations for different occasions and different personalities, we built a website. The UI content includes occasion, relationship, sender's favor, receiver's profile. Sender can easily fill in their consideration step by step. After fill in the answer, system will comes out 20 recommended gifts. If receiver are not satisfied, he/she can also change the answer and search again.

We tried to make a system to be a good helper for the people who might just want a general nudge to the right directions or some inspirations on what to sent.

Related Work

Traditionally, recommender systems are classified into the following three categories: 1. User-based Collaborative Filtering Systems [4]: The user will be recommended items similar to the ones the user preferred in the past. 2. Item-based Collaborative Filtering Systems [7, 6]: The user will be recommended items that people with similar tastes and preferences liked in the past. 3. Hybrid Recommender Systems [1]: These systems combined the above two systems. They usually suggest books, music albums, from a set of input parameters, possibly including user profiles, purchase history, etc.

In recent years, many researchers use novel technology to recommend, e.g. Edward use Commonsense to recommend clothing [8]. Furthermore, some researchers use recommender system in interesting domains, such as Cosleys Movie Recommendation [2], and Koutrikas Course Recommendation [5]. Moreover, the display style of Top 10 is popular again recently [3]. We follow the latest trend, and use the novel technology in interesting domain. In short, we are the first one that combine traditional recommendations and Commonsense, and apply them in Gift Recommendation.

System Architecture

This section describes the basic structure of our system, the flow of the system and we go through some of the UI designs

and finally touch on the algorithms we used to rank and filter the gift list.

Flow

Users of our recommender system start with filling in their considerations on the UI interface.

After filling in the options and considerations, the users will be greeted a list of 20 recommended gifts generated by the system from the information provided by the users. If the users want more information on the recommended gift item, they can click the link and the system will help the users to find the relative products from several popular e-commerce websites.
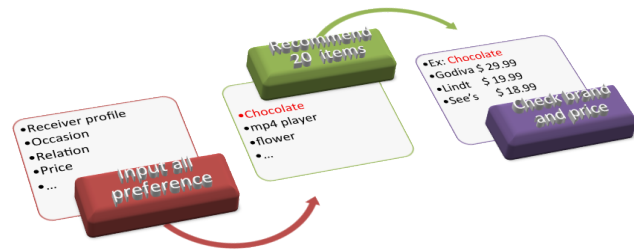

Figure 1: Flow Chart

UI Design

The UI of the GiftGiver is given in Figure 3, it is a website with a rich set of options for users to choose his/her preference for the gifts. We categorize the questions in to 3 main parts, the main considerations, the Receiver profile and the Result based on the result of the questionnaire. Originally we considered both the features and preference of the sender and the receiver, however we want to learn more about what might be the influential factors for picking gifts. We designed a questionnaire to ask 3 adult females and 10 adult males about what they will consider when they sent a gift. Each question has a scale from 1 to 5, from unimportant to most important.

Furthermore, we interviewed 5 differnet people with the primary concerns when picking a nice gift. We later found out the result of the interviews that they agreed the same considerations with the questionnaire results. The results had shown that people concern about the relationship more than occasion. Other considerations did not seem to influence too much to the decision making process compared to occasion and relationship.

Based on those two results from two different groups of people, we cut off some of the considerations from original design, such as Sender's religion, Sender's Gender, etc.

Compare to major e-commerce websites that sells real products, our website doesn't have the real database for all different kinds of gifts, but we had a better flow and more variety and flexibility of options for users to pick. There is also another extinction between our UI design and those of
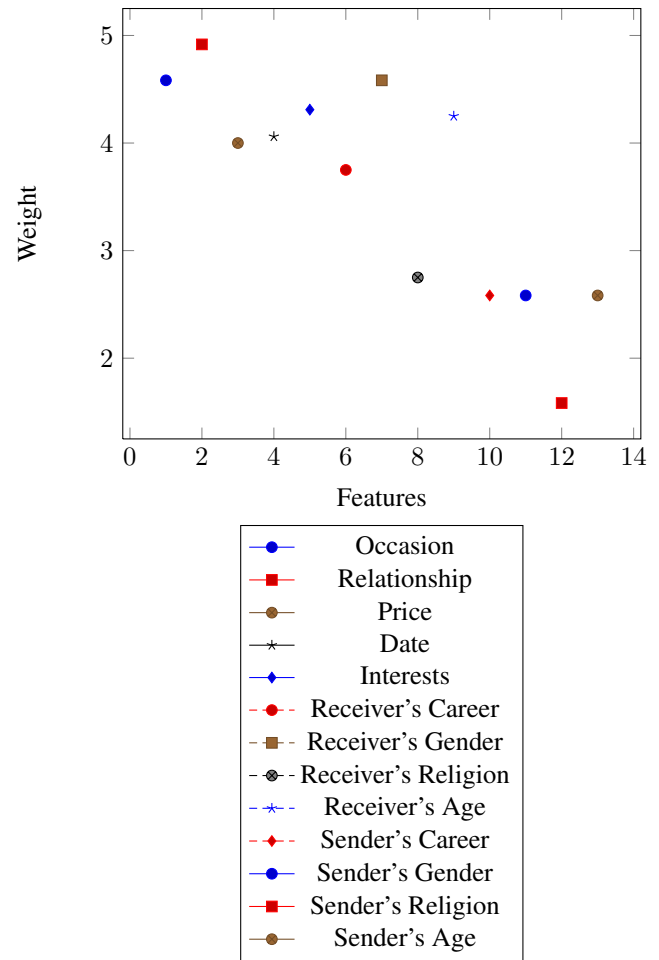

Figure 2: Feature weights

major corporations. For example, Gifts.com has more than 10 ages degree, such as 0 6, 7 12, 13 19, 20 30, 31 40 etc, we use fuzzy set to define our age instead of clear separation by years. Because nouns like child, adult etc. will help the system in the commonsense part. On the other hand, some people still looks like a child even that person has already more than 30s. The fuzzy part will come out positive result.

Methodology

This section briefly discussed our appoaches to rank the gifts and algorithms with commonsense knowledge base

Feature Vectors We have decided to take a feature matching approach of the recommendation system. We analyze 60 gifts items and 3 common occasions and extract only 8 features. The features are listed in Table 1.

Each object such as gift is defined by the feature vector. Each value is normalized from [0,1]. Occasion is also defined by the feature vector, but its existance is essensial to our algorithms. Just as the same as the feature vectors defined in Shen's work [8].
For instance:

Table 1: Feature Vector

Practical	Is it functional or just for decoration?
Meaningful	Is it has other meaning besides its function?
Beautiful	Is it good looking?
Trendy	Is it fashion?
Impersonal	Is it just for him/her or other people can use it?
Romantic	Is it present for love and sweet?
Geeky	Is it electrical?
Fun	Is it let you happy in a pure way?

Figure 3: UI design

Book (Gift)	(1.0, 0.6, 0.2, 0.4, 1.0, 0.0, 0.0, 0.6)
Birthday (Occasion)	(0.2, 1.0, 0.6, 0.2, 0.0, 1.0, 0.2, 0.2)

Occasion is treated differently than normal object. We sort our gift candidates from our sample pool of gifts of 20 by the mean square distance with target occasion as the desired mean that is specified in the UI by the users. Below the pseudocode described our first algorithm to sort gifts.

Algorithm 1 MeanSquaredDistanceSort(Occasion, Gifts)

$dist = 0$
for g in Gifts **do**
   $dist = \frac{1}{8} \sum_{i=1}^{8} \sqrt{(Occasion_i - gift_i)^2}$
end for

However to accommodate selective considerations such as Food, Travel, Sport, etc for users who have a preference to influence the concept relatedness of the gifts candidates, we took a threshold defined for each consideration to further sort and place candidates into two buckets, a superior and inferoir for gifts that satisfied all the constraints and those not.

Commonsense knowledge base   In our system, we use Open Mind Common Sense, a knowledge corpus that contains 800,000 sentences about everyday life, gathered from Web volunteers. Using this resource, we successfully built a fashion recommender system.

Algorithm 2 FilterWithPreferenceThreshold(Sender, Gifts, Features)

for g in Gifts **do**
   for feature in Features that Sender chooses **do**
      if feature(g) < feature(threshold) **then**
         g append to InferiorGifts
      else if feature(g) ≥ feature(threshold) **then**
         g append to BetterGifts
      end if
   end for
end for

ConceptNet ConceptNet aims to give computers access to common-sense knowledge, the kind of information that ordinary people know but usually leave unstated. The data in ConceptNet is being collected from ordinary people who contributed it on sites likeOpen Mind Common Sense. ConceptNet represents this data in the form of a semantic network, and makes it available to be used in natural language processing and intelligent user interfaces. ConceptNet is an open source project, with a Python implementation and a REST API that anyone can use to add computational common sense to their own project. A great tool to help you use ConceptNet in your software is Divisi.
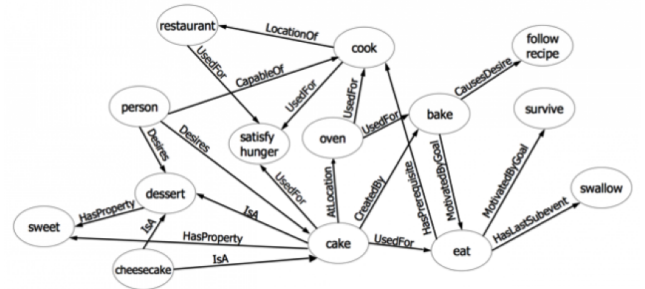

Figure 4: ConceptNet concept graph

Divisi is a library for reasoning by analogy and association over semantic networks, including common sense knowledge. Divisi uses a sparse higher-order SVD can help find related concepts, features, and relation types in any knowledge base that can be represented as a semantic network. By including common sense knowledge from Con-

ceptNet, the results can include relationships not expressed in the original data but related by common sense.

Algorithms A few knowledge about the relationship/relatedness between words are helpful for our gift recommendations to discover similar items and reject disappropriate items. Such as "rose" has a strong relation to Valentines Day, "Cake" has a strong relation to "birthday". To find these kinds of relation, people usually resort to WordNet or statistical models to search for similar appearance between words. If two key word are commonly showed up together, then they possibly have strong relation. But we thought that words that appear together frequently are not always related, using Commonsense Knowledge base such as Divisi we can find out the real 'relatedness' between words. Compare to the text-based statistical models, commonsense has the detail information about two words relation by features.

For the reason below, we use commonsense Divisi to do word analysis. We use fast spreading activation to calculate a categorys relation to a word by spitting out common features. And this function can also help you to find out what are the same concepts between the categorys items.

In step 3, we want to filter the nonrelated items by commonsense. Since occasion and relationship are the most 2 important Information, we wrap them into a category we call a scenario, to find out what kinds of gift has a strong relation with both of them. The strong relation has a harder condition to satisfy. This gift must be top 20 relative items by the category.

Because step 3s condition is too hard to complete, we extend step 4 to find out more relative items by commonsense. This time, all the consideration form UI will be used to calculate spreading activation score. We put each consideration in to one category, and then give each score a weight which from the equalize score of the pre-questionnaire.

Algorithm 3 FilterWithCommonSenseRelation(divisi, Occasion, Relationship, Gifts)

> category = Occasion,Relationship
> **for** g in Gifts **do**
>   **if** category.spreadActivation(g) is in top 20 **then**
>   g.strongscore = category.spreadActivation(g)
>   **else if** feature(g) $\geq$ feature(threshold) **then**
>   g.strongscore = 0
>   **end if**
> **end for**
> sort(Gifts)

Algorithm 4 FilterWithCommonSpreadingActivation(divisi, weights, ReceiverPreference, Gifts)

> **for** g in Gifts **do**
>   **for** pref in ReceiverPreference **do**
>   g.weakscore += pref.spreadActivation(g) * weights(pref)
>   **end for**
>   g.weakscore /= sum(weights)
>   **if** g.weakscore < threshold **then**
>   g.weakscore = 0
>   **end if**
> **end for**
> **for** g in Gifts **do**
>   g.finalscore = 10 * g.strongscore + g.weakscore
> **end for**
> sort(Gifts)

Evaluation

It is rather hard to evaluate recommender system, and even so with common sense knowledge because there is no common criterions to evaluate common sense. However, since we haven't found the state of the art recommender engine using common sense for gift recommendation, we have conducted a couple surveys and user studies to evaluate the accuracy of our recommender system.

Based on time constraints, we conducted a usability test and two surveys with five National Taiwan University graduate students and alumnae. The results of the surveys and the usability test are listed below.
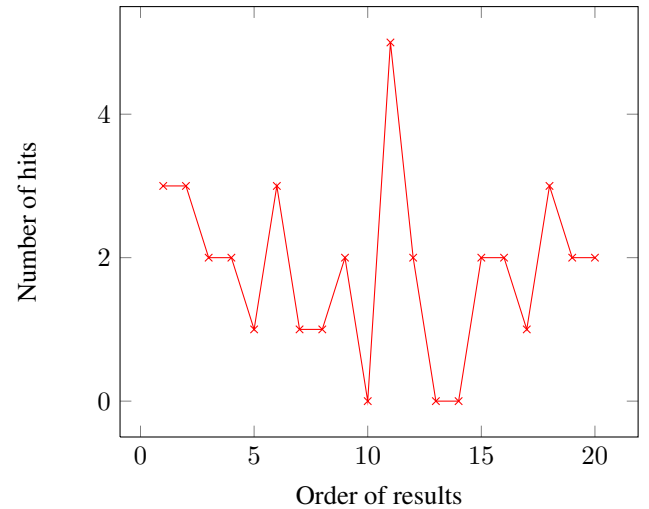

Figure 5: Distribution of hits by order of results

According to the plot above, the distribution is pretty even through out the search results but some bumpy hills along. We define hit rate as the number of times the tester's desired gift is "hit" on the result list. The higher the number the better our system performs. To illustrate the result of our survey in terms of hit rate, the scatter plot below provides the representation for that.

Table 2: Usability test results

| Question	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5
Was the UI easy to use?	3	4	3	3	4
Did the options cover all my considerations for gift recommendation?	3	4	3	4	3
Were the ideal gifts in the result list?	4	4	3	4	4
Were most of the recommendations realistic?	5	3	3	5	4
Was the price checking feature helpful?	4	5	5	5	4
Will you consider using our system for gift recommendation in the future?	5	4	4	5	4
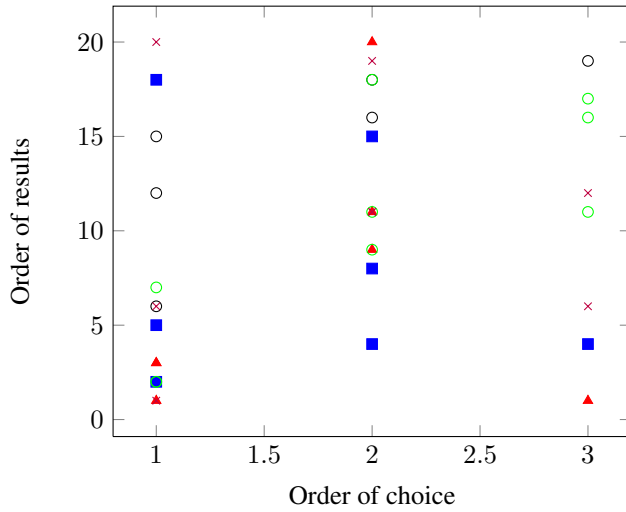
Figure 6: Usability test

Conclusion

Send gift is a cross culture social activity. There are too many combinations in different occasions, different time and different situations. Which gift should we pick? Sometimes even expert might go wrong. As we mention in the beginning, our system is a good idea advisor rather than decision maker. The assistant part let the system popular. And the recommendation result does not have too many details. This kind of gift list help sender not only figure out what he/she really wants to send, but also let him/her think the part of surprise and intention. The volunteer of our usability test gave a nice feedback of our system. Most of the users answered that they will use our system if they plan to send a gift in the future. This success not only attributed to good recommendation results, but also the great design of our UI.

Future Work

Gift recommendation is a new topic of this academic area. There are no such ancestors to reference. As a beginner of this domain, we can only evaluate our system by usability test.

Features   Speaking to our system, the feature vector is build by us. We merge a few peoples idea, then argue with each other, and finally come out the first version of feature table. Although this table comes out a reasonable result, we still want to modify these data from the method like crowd voting, web mining and machine learning. We believe that these methods will let feature table closer and closer to the truth value.

Gift list   On the other hand, our gift list has only 60 gifts. We spent a long time browsing the internet, trying to summarize a list that can overlay the entire domain of the gift candidate and also not too much detail information. We are going to build another page of GiftGiver which people can add new gift items on it. If a particular gift owned more than 80% of Is it suitable to be a gift item? voting. Then this gift might be added into our new gift list.

Date integration   As you can see, the date part has not been used since our system build. In the future, we strongly hope that we can add locations or other useful information, and the commonsense part can be stronger when these information add in.

References

[1] Robin Burke. *Hybrid Systems for Personalized Recommendations.* In Intelligent Techniques for Web Personalization (Lecture Notes in Computer Science) (Vol. 3169, pp. 133-152), 2005.

[2] Dan Cosley, Dan Frankowski, Loren Terveen, and Riedl John. Suggestbot: using intelligent task routing to help people find work in wikipedia. In *International Conference on Intelligent User Interfaces (IUI-07)*, 2007.

[3] Daniel Deutch, Tova Milo, and Tom Yam. Goal-oriented web-site navigation for on-line shoppers. *VLDB Endowment*, 2:1642 – 1645, 2009.

[4] Jonathan L Herlocker, Joseph A Konstan, AI Borchers, and Riedl John. An algorithmic framework for performing collaborative filtering. In *International ACM SIGIR conference on Research and development in information retrieval (SIGIR-99)*, 1999.

[5] Georgia Koutrika, Benjamin Bercovitz, and Hector Garcia-Molina. Flexrecs: expressing and combining flexible recommendations. In *international ACM SIGMOD conference on Management of data (SIGMOD-09)*, 2009.

[6] Greg Linden, Brent Smith, and Jeremy York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Computer Society*, 7:76 – 80, 2003.

[7] Badrul Sarwar, George Karypis, Joseph Konstan, and Reidl John. Item-based collaborative filtering recommendation algorithms. In *International conference on World Wide Web(WWW-01)*, 2001.

[8] Edward Shen, Henry Lieberman, and Francis Lam. What am i gonna wear: Scenario-oriented recommendation. In *International Conference on Intelligent User Interfaces (IUI-07)*, Honolulu, January 2007.