Movie Recommender System for Profit Maximization (Short LBP)

Amos Azaria[1], **Avinatan Hassidim**[1],
Sarit Kraus[1,2], **Adi Eshkol**[3], **Ofer Weintraub**[3] and **Irit Netanely**[3]
[1] Department of Computer Science, Bar-Ilan University, Ramat Gan 52900, Israel
[2] Institute for Advanced Computer Studies University of Maryland, MD 20742
[3] Viaccess-Orca, 22 Zarhin Street, Ra'anana 43662, Israel
{azariaa1,avinatan,sarit}@cs.biu.ac.il, {adi.eshkol, ofer.weintraub,irit.netanely}@viaccess-orca.com

Abstract

In this paper we provide an algorithm for utility maximization of a movie supplier service, in two different settings, one with prices and the other without. This algorithm is provided along with an extensive experiment demonstrating its performance. We also uncover a phenomenon where movie consumers prefer watching and even paying for movies that they have already seen in the past than movies that are new to them.

Introduction & Related Work

The main goal in designing recommender systems is usually to predict the user's wish list and to supply her with the best list of recommendations. However, in most cases, the engineers that design the recommender system are hired by the business which provides the suggestions. The business' end goal is usually to increase sales, revenues, user engagement, or some other metric. In that sense, the user is not the end customer of the recommendation system, although she sees the recommendations (Lewis 2010). Still, one could argue that it is better for the business to give the user the best possible recommendations, as it will also maximize the business' profit, either in the short run (no point in giving recommendations which are not followed by the users) or at least in the long run (good recommendations make users happy).

In this paper, we provide evidence that a business may gain significantly (with little or no long-term loss) by providing users with recommendations that may not be best from the users point of view but serve the business' needs. We provide an algorithm which uses a general recommender system as a black-box and increases the utility of the business. We perform extensive experiments with it in various cases. In particular, we consider two settings:

1. The *Hidden Agenda* setting: In this setting, the business has items that it wants to promote, in a way which is opaque to the user. For example, a movie supplier which provides movies on a monthly fee basis but has different costs for different movies.
2. The *Revenue Maximizing* setting: In this case the goal of the recommender system is to maximize the expected revenue, e.g. by recommending expensive items. In this setting, there is an inherent conflict between the user and the business.

An interesting phenomenon that we uncover is that subjects are more willing to pay for movies that they've already seen. While a similar phenomena is known for other types of consumer goods, coming across it with regards to movies is new and somewhat counter-intuitive. We supply some explanation for this phenomenon, and explain how it can improve the design of recommender systems for movies. However, further research is required.

Chen et al. (2008) develop a recommender system which tries to maximize product profitability. Chen et al. assume the usage of a collaborative filtering recommender system which, as part of its construction, provides a theoretically-based probability that a user will purchase each item. They multiply this probability by the revenue from each item and recommend the items which yield the highest expected revenue. However, in practice, many recommender systems do not rely only upon collaborative filtering, but also rely on different engines (such as popularity, semantic similarity, etc.). Therefore, we assume a generic recommender system which is treated as a black-box component, and dedicate most of our work to building a human model in order to predict the acceptance rate of a given item using a generic recommender system.

In (Azaria et al. 2012) we model the long-term affect of advice given by a self-interested system on the users in path selection problems. However, in (Azaria et al. 2012) we assume that the user must select his action among a limited number of options and the system merely recommends a certain action. Therefore the system does not act as a classic recommender system, which recommends a limited number of items from a very large corpus. Still, this work may be found useful if combined with the approach given in this paper, when considering repeated interactions scenarios.

Profit and Utility Maximizing Algorithm

In this section we present the Profit and Utility Maximizer Algorithm (PUMA) for the revenue maximizing setting. PUMA mounts a black-boxed recommender system which supplies a ranked list of movies.

In order to learn the impact of the price on the likelihood of the users buying a movie, we use the recommender system as is, providing recommendations from 1 to n. We cluster the data into pricing sets where each price (fee f) is associated with the fraction of users who want to buy a movie (m) for that price. Using least squares regression we find a function that best explains the data as a function of the price. The log function resulted with a nearly perfect fit to the data. Therefore, the probability that a user will be willing to pay in order to watch a movie as a function of its fee takes the form of (where α and β are constants): $p(m|f(m)) = \alpha_1 - \beta_1 \cdot ln(f(m))$.

In order to learn the impact of the movie rank (r) in the recommender system on the likelihood of the users buying a movie, we removed all prices from the movies and asked the subjects if they were willing to pay to watch the movie (without mentioning its price). As in the hidden agenda settings, we provided recommendations in leaps of k' (i.e. recommendations are in the group $\{1, k'+1, ..., (n-1) \cdot k'+1\}$). We clustered the data according to the movie rank and once again using least squared regression we found a function that best explains the data as a function of the movie rank. The log function turned out to provide the best fit to the data for the movie rank as well. The probability that a user will be willing to pay in order to watch a movie as a function of its rank takes the form of: $p(m|r(m)) = \alpha_2 - \beta_2 \cdot ln(r(m))$.

A human model for predicting the human willingness to pay to watch a movie, $p(m|r(m), f(m))$, requires combining the above two equations; We omit details on the process and provide only our final human model:

$$p(m|r(m), f(m)) = \alpha_1 - \beta_2 \cdot ln(\frac{r(m)}{\frac{n}{2}+1}) - \beta_1 \cdot ln(f(m))$$

Once a human model is obtained, PUMA calculates the expected revenue from each movie simply by multiplying the movie revenue with the probability that the user will be willing to pay to watch it (obtained from the model) and returns the movies with the highest expected revenues. The revenue is simply the movie price ($f(m)$) minus the movie cost to the vendor ($c(m)$).

Experiments

All of our experiments were performed using Amazon's Mechanical Turk service (AMT) (Amazon 2010). Participation in all experiments consisted of a total of 215 subjects from the USA, of which 49.3% were females and 50.7% were males, with an average age of 31.3. The movie corpus included $16,327$ movies. The original movie recommender system receives a list of preferred movies for each user and returns a ranked list of movies that have a semantically similar description to the input movies. Each subject was recommended 10 movies. In the revenue maximizing settings all movies were randomly assigned a price which was in $F = \{\$0.99, \$2.99, \$4.99, \$6.99, \$8.99\}$.

The subjects were first asked to choose 12 movies which they enjoyed most among a list of 120 popular movies. Then, depending on the experiment, the subjects were divided into different treatment groups and received different recommendations. The list of recommendations included a description of each of the movies. The subjects were shown the price of each movie, when relevant, and then according to their treatment group were asked if they would like to pay in order to watch it, or simply if they would like to watch the movie. In order to assure truthful responses, the subjects were also required to explain their choice. After receiving the list of recommendations and specifying for each movie if they would like to buy it (watch it), the subjects were shown another page including the exact same movies. This time they were asked whether they have seen each of the movies and rated the full list on a scale from 1 to 5.

Results

In the hidden agenda setting PUMA significantly ($p < 0.001$ using student t-test) outperformed the Original Recommender System (ORS) by increasing its promotion value by 57% with an average of 0.684 per movie for PUMA versus an average of only 0.436 per movie for the ORS. No statistically significant differences were observed between the two groups from the average satisfaction for each of the movies (71% rated as good recommendations in the PUMA group vs. 69% in the ORS) or in the user satisfaction from the full list.

PUMA significantly ($p < 0.05$ using student t-test) outperformed ORS in the revenue maximizing setting as well, yielding an average revenue of \$1.71, as opposed to only \$1.33 obtained by the ORS. No significance was obtained when testing the overall satisfaction level from the list: 4.13 vs. 4.04 in favor of the ORS. The average movie price was also similar in both groups, with an average movie price of \$5.18 for ORS and an average movie price of \$5.27 for PUMA.

Subject-Preference for Movies that Have Been Watched Before

We discovered that in all treatment groups subjects preferred paying for movies that they have already watched than movies which were new to them (although differences reached statistical significance only in some of the groups).

This finding may be very relevant to designers of recommender systems for movies. Today, most systems take great care not to recommend movies that the user already saw, while instead perhaps one should try to recommend movies that the user saw and liked.

Conclusions

In this paper we introduce PUMA, an algorithm which mounts a given black-boxed movie recommender system and selects movies which it expects that will maximize the system's revenue. PUMA builds a human model which tries to predict the probability that a user will pay for a movie, given its price and its rank in the original recommender system. We demonstrate PUMA's high performance empirically using an experimental study.

Another important contribution of the paper is uncovering a phenomenon in which people prefer watching and even paying for movies which they have already seen. This phenomenon, which we term WATSA, was tested and found statistically significant in an extensive experimental study as well.

References

Amazon. 2010. Mechanical Turk services. http://www.mturk.com/.

Azaria, A.; Rabinovich, Z.; Kraus, S.; Goldman, C. V.; and Gal, Y. 2012. Strategic advice provision in repeated human-agent interactions. In *AAAI*.

Chen, L.-S.; Hsu, F.-H.; Chen, M.-C.; and Hsu, Y.-C. 2008. Developing recommender systems with the consideration of product profitability for sellers. *Information Sciences* 178(4):1032–1048.

Lewis, A. 2010. If you are not paying for it, you're not the customer; you're the product being sold. *Metafilter*.