On Unexpectedness in Recommender Systems: Or How to Expect the Unexpected

Panagiotis Adamopoulos and Alexander Tuzhilin
Department of Information, Operations and Management Sciences
Leonard N. Stern School of Business, New York University
{padamopo,atuzhili}@stern.nyu.edu

ABSTRACT

Although the broad social and business success of recommender systems has been achieved across several domains, there is still a long way to go in terms of user satisfaction. One of the key dimensions for improvement is the concept of *unexpectedness*. In this paper, we propose a model to improve user satisfaction by generating unexpected recommendations based on the utility theory of economics. In particular, we propose a new concept of unexpectedness as recommending to users those items that depart from what they expect from the system. We define and formalize the concept of unexpectedness and discuss how it differs from the related notions of novelty, serendipity and diversity. We also measure the quality of recommendations using specific metrics under certain utility functions. Finally, we provide unexpected recommendations of high quality and conduct several experiments on a "real-world" dataset to compare our recommendation results with some other standard baseline methods. Our proposed approach outperforms these baseline methods in terms of unexpectedness while avoiding accuracy loss.

Keywords

Recommender Systems, Unexpectedness, Utility Theory

"*If you do not expect it, you will not find the unexpected, for it is hard to find and difficult*".

- Heraclitus of Ephesus, 544 - 484 B.C.

1. INTRODUCTION

Over the last decade, a wide variety of different types of recommender systems (RSs) has been developed and used across several domains [4]. Although the broad-based social and business acceptance of RSs has been achieved and the recommendations of the latest class of systems are significantly more accurate than they used to be a decade ago [6], there is still a long way to go in terms of satisfaction of the users' actual needs. This is due, primarily, to the fact that many existing RSs focus on providing more accurate rather than more novel, serendipitous, diverse and useful recommendations. Some of the main problems pertaining to the narrow accuracy-based focus of many existing RSs and the ways to broaden the current approaches have been discussed in [19].

One of the key dimensions for improvement in RSs that can significantly contribute to the overall performance and usefulness of recommendations and that is still under-explored is the notion of "unexpectedness". RSs often recommend items that the users are already familiar with and, thus, they are of little interest to them. For example, a shopping RS may recommend to customers products such as milk and bread. Although being an accurate recommendation in the sense that the customer will indeed buy these two products, this recommendation is of little interest to the shopper because it is an obvious one: the shopper will, most likely, buy these products even without this recommendation. Therefore, motivated by the potential of higher user satisfaction, the difficulty of the problem and its implications, we try to resolve this problem of recommending items with which the users are already familiar, by recommending *unexpected* items of significant usefulness to them.

Following the Greek philosopher Heraclitus, we approach this hard and difficult problem of finding and recommending the unexpected items by first capturing expectations of the user. The challenge is not only to identify the set of items expected by the user and then derive the unexpected ones but also to enhance the concept of unexpectedness while still delivering recommendations of high quality and achieving a fair match of user's interests.

In this paper, we formalize this concept by providing a new formal definition of unexpected recommendations and differentiating it from various related concepts. We also suggest specific metrics to measure both unexpectedness and quality of recommendations. Finally, we propose a method for generating unexpected recommendations and evaluate the results of the proposed approach.

2. RELATED WORK AND CONCEPTS

In the past, several researchers tried to provide alternative definitions of unexpectedness and various related but still different concepts, such as recommendations of novel, diverse and serendipitous items. In particular, *novel* recommendations are recommendations of those items that the user did not know about [17]. Hijikata et al. in their work [14] use collaborative filtering to derive novel recommendations by explicitly asking users what items they already know and Weng et al. [28] suggest a taxonomy-based RS that utilizes hot topic detection using association rules to improve novelty and quality of recommendations. However, comparing novelty to unexpectedness, a novel recommendation might be unexpected but novelty is strictly defined in terms of previously unknown non-redundant items without allowing for known but unexpected ones. Also, novelty does not include positive reactions of the user to recommendations. Illustrating these differences in the movie context, assume that user John Doe is mainly interested in Action & Adventure films. Recommending the latest popular Children & Family film to this user is definitely a novel recommendation but probably of low utility for him since Children & Family films are not included in his preferences and will be likely considered "irrelevant" because they depart too much from his expectations.

Moreover, *serendipity*, the most closely related concept to unexpectedness, involves a positive emotional response of the user about a previously unknown (novel) item and measures how

surprising these recommendations are [24]; serendipitous recommendations are by definition also novel. Iaquinta et al. propose in [15] to enhance serendipity by recommending novel items whose description is semantically far from users' profiles and Kawamae et al. [16] suggest an algorithm for recommending novel items based on the assumption that users follow the earlier adopters who have demonstrated similar preferences but purchased items earlier. Nevertheless, even though both serendipity and unexpectedness involve positive surprise of the user, serendipity is restricted just to novel items without taking consideration of users' expectations and relevance of the items. To further illustrate the differences of these two concepts, let's assume that we recommend to John Doe the newly released production of his favorite Action & Adventure film director. Although John will probably like the recommended item, such a serendipitous recommendation does not maximize his utility because John was probably expecting the release of this film or he could easily find out about it.

Furthermore, *diversification* is defined as the process of maximizing the variety of items in our recommendation lists. Most of the literature in RSs and Information Retrieval including [2], [3], [26] and [27] studies the principle of diversity to improve user satisfaction. Typical approaches replace items in the derived recommendation lists to minimize similarity between all items or remove "obvious" items from them as in [8]. Adomavicius and Kwon [2], [3] address the concept of aggregated diversity as the ability of a system to recommend across all users as many different items as possible over the whole population while keeping accuracy loss to a minimum, by a controlled promotion of less popular items towards the top of the recommendation lists. Even though avoiding a too narrow set of choices is generally a good approach to increase the usefulness of the final list, since it enhances the chances that the user is pleased by at least some recommended items, diversity is a very different concept from unexpectedness and constitutes an ex-post process that can actually be combined with our model of unexpectedness.

Pertaining to *unexpectedness*, in the field of knowledge discovery, [22] and [23] proposed a characterization of unexpectedness relative to the system of prior domain beliefs and developed efficient algorithms for the discovery of unexpected patterns, which combined the independent concepts of unexpectedness and minimality of patterns. In the field of recommender systems, Murakami et al. [20] and Ge et al. [11] suggested both a definition of unexpectedness as the deviation from the results obtained from a primitive prediction model and metrics for evaluating unexpectedness and serendipity. Also, Akiyama et al. [5] proposed unexpectedness as a general metric that does not depend on a user's record and involves an unlikely combination of features. However, all these approaches do not fully capture the multi-faceted concept of unexpectedness since they do not truly take into account the actual *expectations of the users*, which is crucial according to philosophers, such as Heraclitus, and some modern researchers [22], [23]. Hence an alternative definition of unexpectedness, taking into account prior expectations of the user, and methods for providing unexpected recommendations are still needed. In this paper, we deviate from the previous definitions of unexpectedness and propose a new formal definition as recommending to users those items that depart from what they expect from the RS.

Based on the previous definitions and the discussed similarities and differences, the concepts of novelty, serendipity and unexpectedness are overlapping. Obviously, all these entities are linked to a notion of discovery, as a recommendation makes more sense when it exposes the user to a relevant experience that he/she has not thought of or found yet. However, the part of novelty and serendipity that adds to the usefulness of recommending a specific product can be captured by unexpectedness. This is because unexpectedness includes the positive reaction of a user to recommendations about previously unknown items but without being strictly restricted only to novel items and also because unexpectedness avoids recommendations of items that are obvious, irrelevant and expected to the user.

3. DEFINITION OF UNEXPECTEDNESS

In this section, we formally model and define the concept of unexpected recommendations as those recommendations that significantly depart from the user's expectations. However, unexpectedness alone is not enough for providing truly useful recommendations since it is possible to deliver unexpected recommendations but of low quality. Therefore, after defining *unexpectedness*, we introduce *utility* of a recommendation as a function of recommendation *quality* (specified by item's rating) *and* its *unexpectedness*. We maintain that this utility of a recommended item is the concept on which we should focus (vis-à-vis "pure" unexpectedness) by recommending items with the highest levels of utility to the user. Finally, we propose measures for evaluating the generated recommendations. We define unexpectedness in Section 3.1, the utility of recommendations in Section 3.2 and metrics for their evaluation in Section 3.3.

3.1 Unexpectedness

To define unexpectedness, we start with user expectations. The *expected items* for each user u can be defined as a collection of items that the user is thinking of as serving his/her own current needs or fulfilling his/her intentions indicated by visiting the recommender system. This set of expected items E_u for a user can be specified in various ways, such as the set of past transactions performed by the user, or as a set of "typical" recommendations that he/she expects to receive. For example, in case of a movie RS, this set of expected items may include all the movies seen by the user *and* all their related and similar movies, where "relatedness" and "similarity" are formally defined in Section 4.

Intuitively, an item included in the set of expected movies derives "zero unexpectedness" for the user, whereas the more an item departs from the set of expectations, the more unexpected it is until it starts being perceived as irrelevant by the user. Unexpectedness should thus be a positive, unbounded function of the distance of this item from the set of expected items. More formally, we define *unexpectedness* in recommender systems as follows. First, we define:

$$\delta_{u,i} = d(i\,; E_u) \tag{1}$$

where $d(i; E_u)$ is the distance of item i from the set of expected items E_u for user u. Then, *unexpectedness* of item i with respect to user expectations E_u is defined as some unimodal function Δ of this distance:

$$\Delta\left(\delta_{u,i}\,;\,\delta_u^*\right) \tag{2}$$

where δ_u^* is the best (most preferred) unexpected distance from the set of expected items E_u for user u (the mode of distribution Δ). Intuitively, unimodality of this function Δ indicates that (a) there is only one *most preferred unexpected* distance, (b) an item that greatly departs from user's expectations, even though results in a big departure from expectations, will be probably perceived as irrelevant by the user and, hence, it is not truly unexpected, and (c) items that are close to the expected set are not truly unexpected but rather obvious to the user.

However, recommending the items that result in the highest possible level of unexpectedness would be unreasonable and problematic since recommendations should be of high quality and fairly match users' preferences; otherwise the users might be dissatisfied with the RS. In order to generate recommendations of high quality that would maximize the users' satisfaction, we use certain concepts from the utility theory in economics [18].

3.2 Utility of Recommendations

In the context of recommender systems, we specify the utility of a recommendation of an item to a user in terms of two components: the utility of quality that the user will gain from using the product (as defined by its rating) and the utility of unexpectedness of the recommended item, as defined in Section 3.1. Our proposed model assumes that the users are engaging into optimal utility maximizing behavior [18]. Additionally to the assumptions made in Section 3.1, we further assume that, given the unexpectedness of an item, the greater the rating of this item, the greater the utility of the recommendation to the user.

Consequently, without loss of generality, we propose that we can estimate this overall utility of a recommendation using the previously mentioned utility of quality and the loss in utility by the departure from the preferred level of unexpectedness δ_u^*. This will allow the utility function to have the required characteristics described so far. Note that the distribution of utility as a function of unexpectedness and rating is non-linear, bounded and experiences a global maximum.

Formalizing these concepts, we assume that each user *u* values the quality of an item by a constant q_u and that the quality of the item *i* is represented by the corresponding rating $r_{u,i}$. Then, we define utility derived from the quality of the recommended item *i* to the user *u* as:

$$U_{q_{u,i}} = q_u * r_{u,i} + \varepsilon_{u,i}^q \quad (3)$$

where $\varepsilon_{u,i}^q$ is the error term defined as a random variable capturing the stochastic aspect of recommending item *i* to user *u*.

Correspondingly, we assume that each user values the unexpectedness of an item by a factor λ_u; $\lambda_u$being interpreted as user's tolerance to redundancy and irrelevance. The user losses in utility by departing from the preferred level of unexpectedness δ_u^*. Then, the utility of the unexpectedness of a recommendation can be represented as follows:

$$U_{\delta_{u,i}} = -\lambda_u * \varphi(\delta_{u,i}\,;\, \delta_u^*) + \varepsilon_{u,i}^{\delta} \quad (4)$$

where function φ captures the departure of unexpectedness of item *i* form the preferred level of unexpectedness δ_u^* for the user *u* and $\varepsilon_{u,i}^{\delta}$ is the error term of the specific user and item.

Thus, the utility of recommending an item to a user can be computed as the sum of functions (3) and (4):

$$U_{u,i} = U_{q_{u,i}} + U_{\delta_{u,i}} \quad (5)$$

$$U_{u,i} = q_u * r_{u,i} - \lambda_u * \varphi(\delta_{u,i}\,;\, \delta_u^*) + \varepsilon \quad (6)$$

where ε is the stochastic error term.

Function φ can be defined in various ways. For example, using popular location models for horizontal and vertical differentiation of products in economics [10], [21] and [25], the departure of the preferred level of unexpectedness can be defined as the linear distance:

$$U_{u,i} = q_u * r_{u,i} - \lambda_u * |\delta_{u,i} - \delta_u^*| \quad (7)$$

or the quadratic one:

$$U_{u,i} = q_u * r_{u,i} - \lambda_u * (\delta_{u,i} - \delta_u^*)^2 . \quad (8)$$

Note that the usefulness of a recommendation is linearly increasing with the ratings for these distances. Whereas, given the rating of the product, the usefulness of a recommendation increases with unexpectedness up to the threshold of the preferred level of unexpectedness δ_u^*. This threshold δ_u^* is specific for each user and context. It should also be obvious by now, that two recommended items with different ratings and distances from the set of expected items may derive the same levels of usefulness.

Once the utility function $U_{u,i}$ is defined, we can then make recommendations to user *u* by selecting items *i* having the highest values of utility $U_{u,i}$.

3.3 Evaluation of Recommendations

[4], [13] and [19] suggest that recommender systems should be evaluated not only by their accuracy, but also by other important metrics such as coverage, novelty, serendipity, unexpectedness and usefulness. Hence, we suggest specific measures to evaluate the candidate items and the generated recommendation lists.

3.3.1 Measures of Unexpectedness

Our approach regards unexpectedness of the recommended item as a component of the overall user satisfaction. Therefore, we should evaluate the proposed method for the resulting unexpectedness of the derived recommendation lists.

In order to measure unexpectedness, we follow the approach proposed by Murakami et al. [20] and Ge et al. [11], and adapt their measures to our method. In particular, [11] defines an unexpected set of recommendations (UNEXP) as:

$$UNEXP = RS \setminus PM \quad (9)$$

where PM is a set of recommendations generated by a primitive prediction model, such as predicting items based on users' favorite categories or items' number of ratings, and RS denotes the recommendations generated by a recommender system. When an element of RS does not belong to PM, they consider this element to be unexpected.

As the authors maintain, based on their definition of unexpectedness, unexpected recommendations may not be always useful and, thus, they also introduce serendipity measure as:

$$SRDP = \frac{|UNEXP \cap USEFUL|}{|N|} \quad (10)$$

where USEFUL denotes the set of "useful" items and N the length of the recommendation list. For instance, the usefulness of an item can be judged by the users or approximated by the items' ratings as described in Section 4.2.5.

However, these measures do not fully capture our definition of unexpectedness since PM contains the most popular items and does not actually take into account the expectations of the user. Consequently, we revise their definition and introduce our own metrics to measure unexpectedness as follows.

First of all, we define expectedness (EXPECTED) as the mean ratio of the movies which are included in both the set of expected movies for a user and the generated recommendation list:

$$EXPECTED = \sum_u \frac{|RS_u \cap E_u|}{|N|} . \quad (11)$$

Furthermore, we propose a metric of unexpectedness (UNEXECTED) as the mean ratio of the movies that are not included in the set of expected movies for the user and are included in the generated recommendation list:

$$UNEXPECTED = \sum_u \frac{|RS_u \setminus E_u|}{|N|} . \quad (12)$$

Correspondingly, we can also derive a new metric for serendipity as in (10) based on the proposed metric of unexpectedness (12).

Finally, recommendation lists should also be evaluated for the catalog coverage. The catalog coverage of a recommender describes the area of choices for the users and measures the domain of items over which the system can make recommendations [13].

3.3.2 Measures of Accuracy

The recommendation lists should also be evaluated for the accuracy of rating and item prediction.

(i) *Rating prediction*: The Root Mean Square Error (RMSE) is perhaps the most popular measure of evaluating the accuracy of predicted ratings:

$$\text{RMSE} = \sqrt{\frac{1}{|R|}\sum_{(u,i)\in R} (\hat{r}_{u,i} - r_{u,i})^2} \quad (13)$$

where $\hat{r}_{u,i}$ is the estimated rating and R is the set of user-item pairs (*u, i*) for which the true ratings $r_{u,i}$ are known.

Another popular alternative is the Mean Absolute Error (MAE):

$$\text{MAE} = \sqrt{\frac{1}{|R|}\sum_{(u,i)\in R} |\hat{r}_{u,i} - r_{u,i}|} \,. \quad (14)$$

(ii) *Item prediction*: We can classify all the possible results of a recommendation of an item to a user as in Table 1:

Table 1. Classification of the possible result of a recommendation.

	Recommended	**Not Recommended**
Used	True-Positive (tp)	False-Negative (fn)
Not Used	False-Positive (fp)	True-Negative (tn)

and compute the following popular quantities for item prediction:

$$\text{Precision} = \frac{\#\, tp}{\#\, tp + \#\, fp} \quad (15)$$

$$\text{Recall (True Positive Rate)} = \frac{\#\, tp}{\#\, tp + \#\, fn} \quad (16)$$

$$\text{False Positive Rate (1 - Specificity)} = \frac{\#\, fp}{\#\, fp + \#\, tn} \quad (17)$$

4. EXPERIMENTS

To empirically validate the method presented in Section 3 and evaluate unexpectedness of recommendations generated by this method, we conduct experiments on a "real-world" dataset and compare our results to popular Collaborative Filtering methods.

Unfortunately, we could not compare our results with other methods for deriving unexpected recommendations for the following reasons. Most of the existing methods are based on related but different principles such as diversity and novelty. Since these concepts are different from our definition, they cannot be directly compared with our approach. Further, among the previously proposed methods of unexpectedness that are consistent with our approach, as explained in Section 2, authors of these methods do not provide any clear computational algorithm for unexpected recommendations but metrics, thus making the comparison impossible. Consequently, we selected a number of standard collaborative filtering (CF) algorithms as baseline methods to compare with the proposed approach. In particular, we selected the k-nearest neighborhood approach (kNN), the Slope One (SO) algorithm and a matrix factorization (MF) approach.[1] We would like to point out that, although the selected CF methods do not explicitly support the notion of unexpectedness, they constitute fairly reasonable baselines because, as was pointed out in [9], CF methods perform reasonably well in terms of some other performance measures besides classical accuracy measures, and indeed our empirical results reported in Section 5 confirm this general observation of [9] for unexpected recommendations.

4.1 Dataset

The basic dataset we used is the RecSys HetRec 2011 [1] MovieLens dataset. This is an extension of a dataset published by GroupLens research group [12], which contains personal ratings and tags about movies. This dataset consists of 855,598 ratings (0.5 - 5) from 2,113 users on 10,197 movies (on average about 405 ratings per user and 85 ratings per movie). In the dataset, the movies are linked to the Internet Movie Database (IMDb) and RottenTomatoes (RT) movie review systems. Each movie has its IMDb and RT identifiers, English and Spanish titles, picture URLs, genres, directors, actors (ordered by "popularity" per movie), RT audience' and experts' ratings and scores, countries, and filming locations. It also contains the tag assignments of the movies provided by each user. However, this dataset does not contain any demographic information about the users.

The selected dataset is relatively dense (3.97%) compared to other frequently used datasets (e.g. the original Netflix Prize dataset [7]) but we believe that this specific characteristic is a virtue that will let us better evaluate our methods since it allows us to better approximate the set of expected movies for each user.

In addition, we used information and further details from Wikipedia and the database of IMDb. Joining these datasets we were able to enhance the information included in our basic dataset by finding any missing values of the movie attributes that were mentioned above and, also, identifying whether a movie is an episode or sequel of another movie included in our dataset. We succeeded to identify *related* movies (i.e. episodes, sequels, movies with exact the same title) for 2,443 of our movies (23.95% of the movies with 2.18 related movies on average and a maximum of 22 "related" movies). We used this information about related movies to identify sets of expected movies, as described in Section 4.2.3.

4.2 Experimental Setup

We conducted in total 2160 experiments. In the one half of the experiments we explore the simpler case where the users are homogeneous (Hom) and have exactly the same preferences. In the other half, we investigate the more realistic case (Het) where users have different preferences that depend on their previous interactions with the system. Furthermore, we use two different sets of expected movies for each user, and different utilities functions. Also, we conducted experiments using different rating prediction algorithms, various measures of distance between movies and between a movie and the set of expected movies for each user. Finally, we derived recommendation lists of different sizes (k = {10, 20, …, 100}). In conclusion, we used 2 sets of expected movies × 3 algorithms for rating prediction × 3 correlation metrics × 3 distance metrics × 2 utility functions × 2 assumptions about users preferences × 10 different lengths of recommendation lists, resulting in 2160 experiments in total.

4.2.1 Utility of Recommendation

In our experiments, we considered the following utility functions:

(a1) *Homogeneous users with linear distance* (Hom-Lin): This is the simpler case where users are homogeneous and have similar preferences (i.e. q, λ, δ^*) and the departure of the preferred level of unexpectedness is linear as in function (7).

(a2) *Homogeneous users with quadratic distance* (Hom-Quad): The users are assumed to be homogeneous but the departure of the preferred level of unexpectedness is quadratic as in function (8).

[1] Various algorithms including baseline methods for rating prediction and matrix factorization with explicit user and item bias were tested with similar results.

(b1) *Heterogeneous users with linear distance* (Het-Lin): Here, the users are heterogeneous and have different preferences (i.e. $q_u, \lambda_u, \delta_u^*$) and the departure of the preferred level of unexpectedness is linear. This case corresponds to function (7)

(b2) *Heterogeneous users with quadratic distance* (Het-Quad): This is the more realistic case. Users have different preferences and the departure of the preferred level of unexpectedness is quadratic. This case corresponds to function (8).

4.2.2 Item Similarity

To build the set of expected movies, the system calculates the distance *d* between two movies by measuring the relevance of these movies. In our experiments, we use both collaborative-based and content-based similarity for the item distance. [2]

(i) The collaborative filtering similarity can be defined using (a) the Pearson correlation coefficient:

$$\rho_{i,j} = \frac{\sum_{u \in U(i,j)} \left((r_{u,i} - \overline{r_{u,i}})(r_{u,j} - \overline{r_{u,j}}) \right)}{\sqrt{\sum_{u \in U(i,j)} (r_{u,i} - \overline{r_{u,i}})^2 \sum_{u \in U(i,j)} (r_{u,j} - \overline{r_{u,j}})^2}}, \quad (18)$$

(b) the Cosine similarity:

$$sim(i,j) = \cos(\theta) = \frac{i \cdot j}{\|i\| \|j\|} = \frac{\sum_u r_{u,i}\, r_{u,j}}{\sqrt{\sum_u r_{u,i}^2} \sqrt{\sum_u r_{u,j}^2}} \quad (19)$$

and (c) the Jaccard coefficient:

$$J(i,j) = \frac{|A \cap B|}{|A \cup B|} \quad (20)$$

where A is the set of users who rated movie *i* and B the set of users who rated movie *j*.

(ii) The content based similarity of movies *i* and *j* is defined as:

$$sim(i,j) = \frac{\sum_{k=1}^{n} w_k * \sigma_{\kappa_{i,j}}}{\sum_{k=1}^{n} w_k} \quad (21)$$

where movie *i* is represented by a vector of its attributes:

$$i = (a_1, a_2, \ldots, a_n)$$

and $\sigma_{k_{i,j}}$ is the similarity of the value of attribute *k* of the movie *i* with the corresponding value of this attribute for movie *j* and w_k the weight of this attribute.

4.2.3 Expected Movies

We use the following two examples of definitions of *expected* movies in our study. The first set of expected movies ($E_{u,Short}$) for user *u* follows a very strict user-specific definition of expectedness, as defined in Section 3. The profile of user *u* consists of the set of movies that he/she has already rated. In particular movie *i* is *expected* for user *u* if the user has already rated some movie *j* such that *i* has the same title or is an episode or sequel of movie *j*, where episode or sequel is identified as explained in Section 4.1. In our dataset, on average a user rated 405 movies and the number of expected movies per user is 586; augmenting the number of rated movies by 44.75%.

The second set of expected movies ($E_{u,Long}$) follows a broader definition. It includes the first set plus a number of closely "related" movies ($E_{u,Long} \supseteq E_{u,Short}$). In order to form the second set of expected movies we, also, use content-based similarity between movies. We first compute the attribute-specific distance between the values of each attribute (e.g. distance between the Comedy and Adventure genres) based on the similarity metrics and, then, use the weighted distance described in Section 4.2.2 for the attributes of each movie (i.e. language, genre, director, actor, country of filming and year of release) in order to compute the final distance between two movies.

More specifically for this second case, two movies are *related* if at least one of the following conditions holds: (i) they were produced by the same director, belong to the same genre and are released within an interval of 5 years, (ii) the same set of protagonists appear in both of them (where protagonist defined as actor with ranking in our dataset = {1, 2, 3}) and they belong to the same genre, (iii) the two movies share more than twenty common tags, are in the same language and their correlation metric is above a certain threshold θ (Jaccard Coefficient (J) > 0.50), (iv) there is a link from the Wikipedia article for movie *i* to the article for movie *j* and the two movies are sufficiently correlated (J > 0.50) and (v) the content-based distance metric defined in this subsection is below a threshold θ (d < 0.50). The average size of the extended set of expected movies per user is 1127, thus increasing the size of rated movies by 178% (7% of the total number of movies).

4.2.4 Distance from the Set of Expected Movies

We can then define the distance of movie *i* from the set of expected movies E_u for user *u* in various ways. For example, it can be determined by averaging the distances between the candidate item *i* and all the items included in set E_u:

$$d(i, E_u) = \frac{\sum_{j=1}^{|E_u|} d(i,j)}{|E_u|} \quad (22)$$

where *d* is defined as in Section 4.2.2. Another approach is based on the Hausdorff distance:

$$d(i, E_u) = \inf\{d(i,j) : j \in E_u\}. \quad (23)$$

Additionally, we also use the Centroid distance that is defined as the distance of an item *i* from the centroid point of the set of expected movies E_u for the user *u*.

4.2.5 Measures of Unexpectedness and Accuracy

To evaluate our approach in terms of unexpectedness, we use the measures described in Section 3.3.1. For the primitive prediction model of (9) we used the top-N items with the highest average rating and the top-N items with the largest number of ratings in order to form the list of top-K items (where K=100) which form our PM recommendation list.

Additionally, we introduce expectedness´ (EXPECTED´) as the mean ratio of the movies that are either included in the set of expected movies for a user or in the primitive prediction model and are also included in the generated recommendation list:

$$EXPECTED' = \sum_u \frac{|RS_u \cap (E_u \cup PM)|}{|N|}. \quad (24)$$

Correspondingly, we define unexpectedness' (UNEXPECTED´) as the mean ratio of the movies that are neither included in the set of expected movies for users nor in the primitive prediction model and are included in the generated recommendation list:

$$UNEXPECTED' = \sum_u \frac{|RS_u \setminus (E_u \cup PM)|}{|N|}. \quad (25)$$

Based on the ratio of Ge et al. (10), we also use the metrics SERENDIPITY and SERENDIPITY´ to evaluate serendipitous recommendations in conjunction with the proposed measures of unexpectedness in (12) and (25), respectively. In our experiments, we consider an item to be useful if its average rating is greater than 3.0 (USEFUL = $\{i : \hat{r}_i > 3.0\}$).

Finally, we evaluate the generated recommendations lists based on the coverage of our product base and accuracy of rating and item prediction using the metrics discussed in Section 3.3.

[2] Other measures such as the set correlation and conditional probabilities were tested with no significant differences.

5. RESULTS

In order to estimate the parameters of preferences (i.e. q_u, λ_u) we used models of multiple linear regression. In our experiments, the average q_u was 1.005. For the experiments with the first set of expected movies the average λ_u was 0.158 for the linear distance and 0.578 for the quadratic one. For the extended set of expected movies the average λ_u was 0.218 and 0.591, respectively. Furthermore, to estimate the preferred level of unexpectedness δ_u^* for each user and distance metric, we used the average distance of rated movies from the set of expected movies; for the case of homogeneous users, we used the average value over all users.

The experiments conducted using the Hausdorff distance indicate inconsistent performance and sometimes, except for the metric of coverage, under-performed the standard CF methods. Henceforth we present the results only for the rest of the experiments.[3] We have to note that the experiments using heterogeneous users uniformly outperform those conducted under the assumption of homogeneous users. The most realistic case of heterogeneous users for the extended set of expectations outperformed all the other approaches including the standard CF methods in 99.08% of the conducted experiments. Also, it was observed that smaller sizes of recommendation lists resulted in constantly greater improvements.

5.1 Comparison of Coverage

For the first set of expected movies, in the case of homogeneous users (Hom-Short), the average coverage was increased by 36.569% and, in the case of heterogeneous users (Het-Short), by 108.406%. For the second set of expected movies, the average coverage was increased by 61.898% and 80.294% in the cases of homogeneous users (Hom-Long) and heterogeneous users (Het-Long), respectively (figure 1).

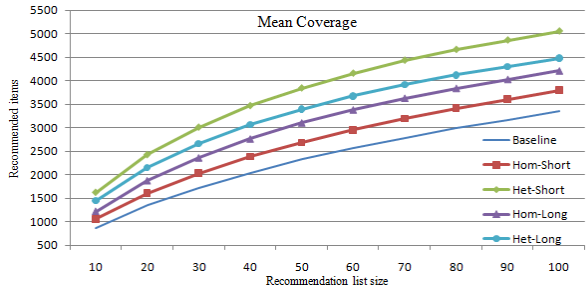

Figure 1. Comparison of mean coverage.

Coverage was increased in 100% of the experiments with a maximum of 7982 recommending items (78.278%). No differences were observed between the linear and quadratic distances whereas the average distance performed better than the centroid one. The biggest average increase occurred for the Slope One algorithm and the smallest for the Matrix Factorization.

5.2 Comparison of Unexpectedness

For the first set of expected movies, the EXPECTED metric was decreased by 6.138% in the case of homogeneous users and by 75.186% for the heterogeneous users. For the second set of expected items, the metric was decreased by 61.220% on average for the homogeneous users and by 78.751% for the heterogeneous. Similar results were also observed for the EXPECTED´ metric. For the short set of expected movies, the metric was decreased by 3.848% for the homogeneous users and by 26.988% for the heterogeneous. For the long set of expected movies, the ratio was decreased by 39.197% and 47.078%, respectively. Our approach outperformed the standard methods in 94.93% of the experiments (100% for heterogeneous users).

Furthermore, the UNEXPECTED metric increased by 0.091% and 1.171% in the first set of experiments for the homogeneous and heterogeneous users, respectively. For the second set of expected movies, the metric was improved by 4.417% for the homogeneous users and by 5.516% for the heterogeneous (figure 3).

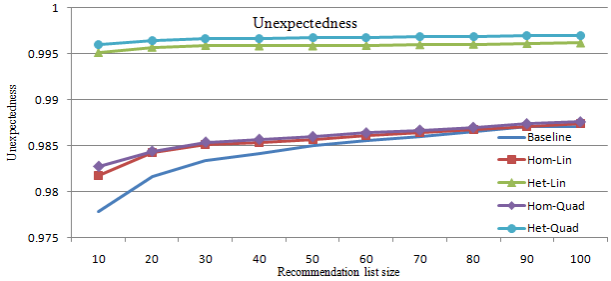

Figure 2. Comparison of Unexpectedness for the 1st set of expectations.

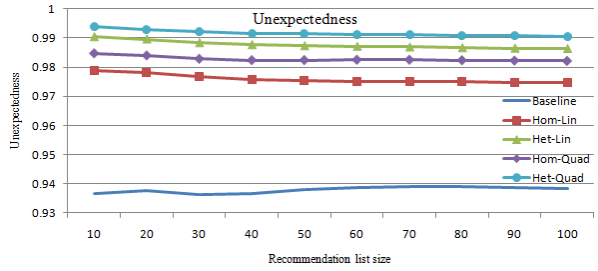

Figure 3.Comparison of Unexpectedness for the 2nd set of expectations.

The worst performance of our algorithm was observed in the experiments using the Matrix Factorization algorithm, the first set of expected movies and the linear function of distance under the assumption of homogeneous users (figure 4).

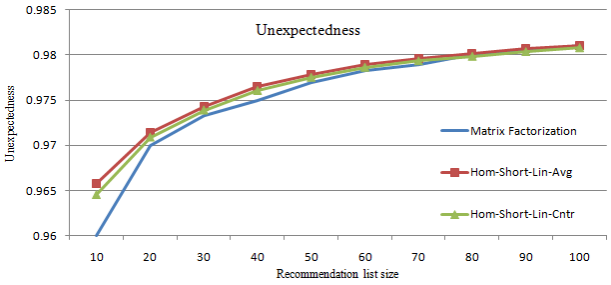

Figure 4. Worst case scenario of Unexpectedness.

As it was expected based on the previous metrics, for the first set of expected movies, the UNEXPECTED' metric was increased by 3.366% and 8.672% in the cases of homogeneous and heterogeneous users, respectively. For the second set of expected movies, in the case of homogeneous users the ratio increased by 8.245% and for the heterogeneous users by 11.980%. It was also observed that using the quadratic distance resulted in more unexpected recommendations. The greatest improvements were observed for the case of Slope One algorithm. Correspondingly, for the metric of unexpectedness given by (9), for the first set of expected movies, the ratios increased by 3.491% and 7.867%. For the second set of expected movies, in the case of homogeneous users, the metric was improved by 4.649% and in the case of

[3] Due to space limitations and the large number of experiments, only aggregated results are presented. For non-significant differences we plot the necessary dimensions or mean values.

heterogeneous users by 7.660%. Our approach outperformed the standard CF methods in 92.83% of the experiments (97.55% for the case of heterogeneous users).

Moreover, considering an item to be useful if its average rating is greater than 3.0, the SERENDIPITY metric increased, in the first set of experiments, by 2.513% and 3.418% for the homogeneous and heterogeneous users, respectively. For the second set of expected movies (figure 6), the metric was improved by 5.888% for the homogeneous users and by 9.392% for the heterogeneous.

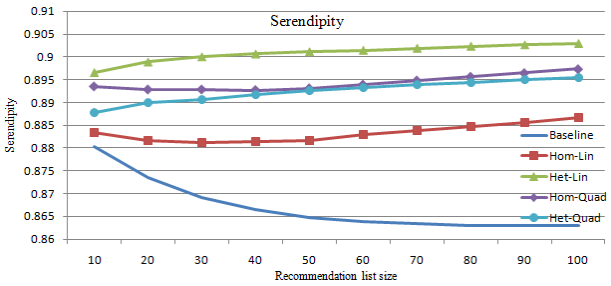

Figure 5. Comparison of Serendipity for the 1st set of expectations.

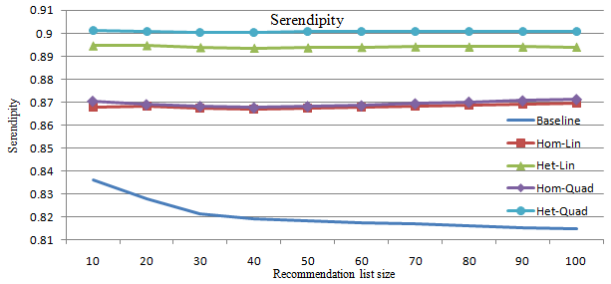

Figure 6. Comparison of Serendipity for the 2nd set of expectations.

The worst performance of our algorithm was observed again using the assumption of homogeneous users with the first set of expected movies and the linear function of distance (figure 7).

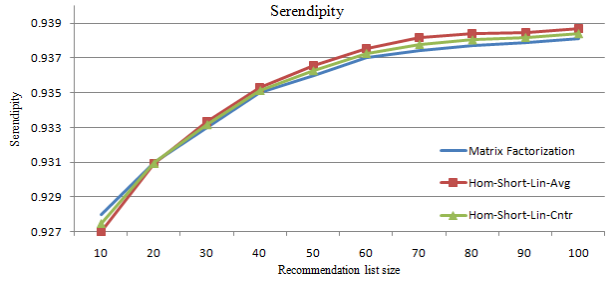

Figure 7. Worst case scenario of Serendipity.

In the first set of experiments, the metric SERENDIPITY' increased by 6.284% and 11.451% for the homogeneous and heterogeneous users, respectively. For the second set of expected movies, the metric was improved by 10.267% for the homogeneous users and by 16.669% for the heterogeneous. As expected, the metric of serendipity given by (10) increased by 6.488% in the case of homogeneous users and by 10.625% in the case of heterogeneous, for the short set of expected items. For the case of homogeneous users and the second set of expected movies the ratio was improved by 6.399% and by 12.043% for the heterogeneous users. Our approach outperformed the standard methods in 85.03% of the experiments.

Additionally, qualitatively evaluating the experimental results, our approach, unlike to many popular websites, avoids anecdotal recommendations such as recommending to a user the movies "The Lord of the Rings: The Return of the King", "The Bourne Identity" and "The Dark Knight" because the user had already highly rated all the sequels / prequels of these movies (MF, k = 10, user id = 11244).

5.3 Comparison of Rating Prediction

The accuracy of rating prediction for the first set of expected movies and the case of the homogeneous users resulted in 0.058% higher RMSE and 0.015% lower MAE on average. Respectively, in the case of heterogeneous customers the RMSE was improved by 1.906% and the MAE by 0.988% on average. For the second set of expected movies, in the case of homogeneous users, the RMSE was reduced by 1.403% and the MAE by 0.735%. For heterogeneous users, the RMSE was improved by 1.548% and the MAE by 0.821% on average with an overall minimum of 0.680 RMSE and 0.719 MAE. The differences between linear and quadratic utility functions are not statistically significant.

Table 2. Mean % improvement of accuracy.

%		Hom-Short		Het-Short		Hom-Long		Het-Long	
		RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
KNN	Avg	0.11	0.01	0.67	4.17	8.30	4.00	8.23	4.03
	Cnt	-0.5	-0.2	8.59	0.33	0.10	0.00	0.18	0.07
MF	Avg	0.02	0.04	0.32	0.23	0.00	0.10	0.03	0.09
	Cnt	0.00	0.10	0.30	0.22	0.00	0.10	0.10	0.14
Slope One	Avg	0.01	0.08	0.80	0.50	0.01	0.12	0.32	0.23
	Cnt	0.01	0.06	0.76	0.48	0.01	0.09	0.43	0.36

5.4 Comparison of Item Prediction

For the case of the first set of expected movies, the precision was improved by 25.417% on average for homogeneous users and by 65.436% for heterogeneous users (figure 8). For the extended set (figure 9), the figures are -58.158% and 65.437%, respectively. Similar results were observed for other metrics such AUC and F1.

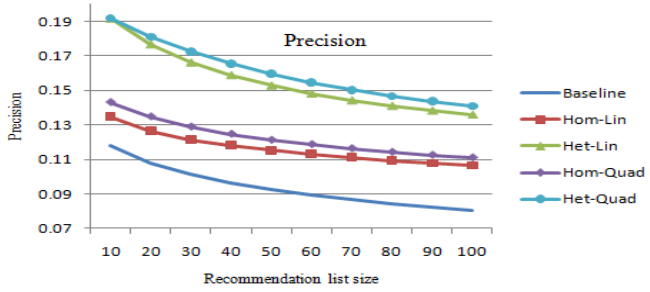

Figure 8. Comparison of Precision for the 1st set of expectations.

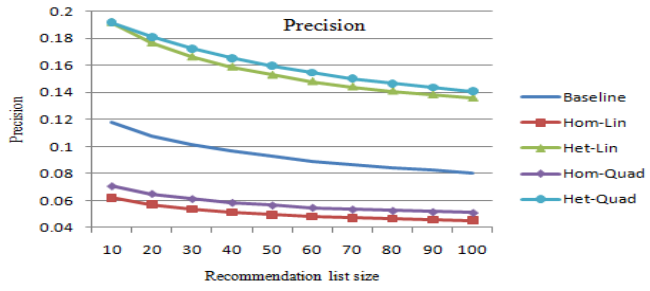

Figure 9. Comparison of Precision for the 2nd set of expectations.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed and studied a concept of unexpected recommendations as recommending to a user those items that depart from what the specific user expects from the recommender system. After formally defining and formulating theoretically this concept, we discussed how it differs from the related notions of novelty, serendipity and diversity. We presented a method for deriving recommendations based on their utility for the user and compared the quality of the generated unexpected recommendations with some baseline methods using the proposed performance metrics.

Our experimental results demonstrate that our proposed method improves performance in terms of both unexpectedness *and* accuracy. As discussed in Section 5, all the examined variations of the proposed method, including homogeneous and heterogeneous users with different departure functions, significantly outperformed the standard Collaborative Filtering algorithms, such as k-Nearest Neighbors, Matrix Factorization and Slope One, in terms of measures of unexpectedness. This demonstrates that the proposed method is indeed effectively capturing the concept of unexpectedness since in principle it should do better than unexpectedness-agnostic classical CF methods. Furthermore, the proposed unexpected recommendation method performed at least as well as, and in most of the cases even better than, the baseline CF algorithms in terms of the classical rating prediction accuracy-based measures, such as RMSE and MAE. In the case of heterogeneous users our method also outperforms the CF methods in terms of usage prediction measures such as precision and recall. Thus, the proposed method performed well in terms of both the classical accuracy and the unexpectedness performance measures.

The greatest improvements both in terms of unexpectedness and accuracy vis-à-vis all other approaches were observed in the most realistic case of the extended set of expected movies under the assumption of heterogeneous users. The assumption of heterogeneous users allowed for better approximation of users' preferences at the individual level, while the extended set of expected movies allowed us to better estimate the expectations of each user through a more realistic and natural definition of closely "related" movies.

As a part of the future work, we are going to conduct experiments with real users for evaluating unexpectedness and analyze both qualitative and quantitative aspects in order to enhance the proposed method and explore other ideas as well. Moreover, we plan to introduce and study additional metrics of unexpectedness and compare recommendation performance across these different metrics. We also aim to use different datasets from other domains with users' demographics so as to better estimate the required parameters and derive a customer theory. Overall, the field of unexpectedness in recommending systems constitutes a relatively new and underexplored area of research where much more work should be done to solve this important, interesting and practical problem.

7. REFERENCES

[1] 2nd Workshop on Information Heterogeneity and Fusion in Recommender Systems, 5th ACM Conf. on Recommender Systems (RecSys 2011) http://ir.ii.uam.es/hetrec2011

[2] Adomavicius, G., & Kwon, Y. Toward more diverse recommendations: Item re-ranking methods for recommender systems. In *WITS* (2009).

[3] Adomavicius, G., & Kwon, Y. Improving aggregate recommendation diversity using ranking-based techniques. *IEEE TKDE* (2011), pp. 1-15.

[4] Adomavicius, G., & Tuzhilin, A. Toward the next generation of recommender systems: A Survey of the state-of-the-art and possible extensions. *IEEE TKDE* (2005), pp. 734-749.

[5] Akiyama, T., Obara, T., & Tanizaki, M. Proposal and evaluation of serendipitous recommendation method using general unexpectedness. In *PRSAT RecSys* (2010).

[6] Bell, R., Bennett, J., Koren, J., & Volinsky, C. The million dollar programming prize. *IEEE Spectrum* 46, 5 (2009).

[7] Bennett, J., & Lanning, S. The Netflix Prize. In KDD (2007).

[8] Billsus, D., & Pazzani, M. User modeling for adaptive news access. *UMUAI* 10, 2-3 (2000), pp. 147-180.

[9] Burke, R. Hybrid recommender systems: Survey and experiments. *UMUAI* 12, 4 (2002), pp. 331-370.

[10] Cremer, H. & Thisse, J.F. Location models of horizontal differentiation: A special case of vertical differentiation models. *The Journal of Industrial Economics* 39, 4 (1991).

[11] Ge, M., Delgado-Battenfeld, C., & Jannach, D. Beyond accuracy: Evaluating recommender systems by coverage and serendipity. In *RecSys* (2010).

[12] GroupLens Research. http://www.grouplens.org

[13] Herlocker, J., Konstan, J., Terveen, L., & Riedl J. Evaluating collaborative filtering recommender systems. ACM *TOIS* 22, 1 (2004), pp. 5-53.

[14] Hijikata, Y., Shimizu, T., & Nishida, S. Discovery-oriented collaborative filtering for improving user satisfaction. *IUI* (2009), pp. 67-76.

[15] Iaquinta, L., Gemmis, M. D., Lops, P., Semeraro, G., Filannino, M., & Molino, P. Introducing serendipity in a content-based recommender system. In *HIS* (2008).

[16] Kawamae, N., Sakano, H., & Yamada, T. Personalized recommendation based on the personal innovator degree. In *RecSys* (2009).

[17] Konstan, J., McNee, S., Ziegler, C.N., Torres, R., Kapoor, N., & Riedl, J. Lessons on applying automated recommender systems to information-seeking tasks. In *AAAI* (2006).

[18] Marshall, A. *Principles of Economics*, 8th ed. Macmillan and Co., London, UK, (1926).

[19] McNee, S., Riedl, J., & Konstan, J. Being accurate is not enough: How accuracy metrics have hurt recommender systems. In *CHI* (2006).

[20] Murakami, T., Mori, K., & Orihara, R. Metrics for evaluating serendipity of recommendation lists. In *JSAI* (2007).

[21] Neven, D. Two stage equilibrium in Hotelling's model. *The Journal of Industrial Economics* 33, 3 (1985), pp. 317-325.

[22] Padmanabhan, B., & Tuzhilin, A. A belief-driven method for discovering unexpected patterns. *KDD* (1998), pp. 94-100.

[23] Padmanabhan, B., & Tuzhilin, A. Unexpectedness as a measure of interestingness in knowledge discovery. *Decision Support Systems* 27, 3 (1999), pp. 303-318.

[24] Shani, G., & Gunawardana, A. Evaluating recommendation systems, In *Recommender Systems Handbook*, Springer-Verlag, New York, NY, USA, (2011), pp. 257-297.

[25] Tirole, J. Product differentiation: Price competition and non-price competition. *The Theory of Industrial Organization*. The MIT Press, Cambridge, USA, (1988).

[26] Zhang, M., & Hurley, N. Avoiding monotony: Improving the diversity of recommendation lists. In *RecSys* (2008).

[27] Ziegler, C.N., McNee, S., Konstan, J., & Lausen, G. Improving recommendation lists through topic diversification. In *WWW* (2005).

[28] Weng, L.T., Xu, Y., Li, Y., & Nayak, R. Improving recommendation novelty based on topic taxonomy. *WIC* (2007), pp. 115–118.