INFORMATYKA EKONOMICZNA BUSINESS INFORMATICS 1(31) • 2014

ISSN 1507-3858

Mirosława Lasek, Dominik Kosieradzki
University of Warsaw

PRODUCTS AND SERVICES RECOMMENDATION SYSTEMS IN E-COMMERCE. RECOMMENDATION METHODS, ALGORITHMS, AND MEASURES OF THEIR EFFECTIVENESS

Summary: The article concerns products and services recommendation systems in e-commerce which have become increasingly important for both consumers and retailers. The methods used for the recommendation of products and services, as well as the algorithms used to implement them, are presented in the article. Particular attention was paid to the problems of testing the suitability of algorithms, along with the effectiveness measures of the applications of the methods and algorithms.

Keywords: recommendation systems, recommender systems, recommendation methods, recommendation algorithms, collaborative filtering, content-based filtering, memory-based algorithms, model-based algorithms, predictive accuracy metrics, classification accuracy metrics, rank accuracy metrics.

DOI: 10.15611/ie.2014.1.23

1. Introduction

Electronic commerce constitutes an immense branch of the global economy and it is still growing in a very dynamic manner. More and more consumers eagerly use online retail stores due to their convenient use and easy access to products and services that are not available in local stores.

Thanks to contemporary logistics and distribution possibilities, any product can be easily delivered to almost any address in the world. This creates a great opportunity for retailers whose target group may be dispersed around the world, as well as for producers of niche products operating on limited local markets. Access to the electronic market allows to diversify the offered products and services. According to the data from 2004, as much as 25% of products available through the *Amazon* online retail store were not available at their local counterparts [Anderson 2004].

Assumedly this ratio has grown even higher over the years. Therefore the electronic market allows consumers to buy products and services matching their preferences better than ever before. Striving to reach consumers willing to buy their products, Internet retail stores implement tools offering personalized information about the products and services on their website.

Systems meant to recommend products or services precisely matching customer preferences are referred to as recommendation systems or recommender systems [Kosieradzki 2013]. When a user visits his or her favorite online store, the recommendation system offers products or services which may interest the user. For example, if a user is browsing a website concerning his or her favorite movie, the system recommends different movies which could be suitable for this user. Hence, recommender systems simplify the choice of merchandise or services, at the same time increasing customer satisfaction.

Focusing on buyers' convenience and customer support is an extremely important success factor of a prospering company, which significantly affects financial results. Available estimates state that as much as 30% of the sales of large online stores come from recommendations generated by recommendation systems [Schonfeld 2007]. Moreover, consumer behavior analysis indicates that as little as 22% of users visiting online stores are determined to make a specific purchase [*Retail/E-Commerce Industry Report* 2012]. This points to a huge discrepancy in the amount of shared information between the demand side and the supply side of the market. The use of recommendation systems helps to reduce this gap by assisting consumers in clearly identifying their needs or discovering new products and taking advantage of attractive offers. For retailers, this helps in presenting their actual product range to customers.

The business environment recognized the enormous commercial potential of recommendation systems relatively late, at the end of the 1990's, when P. Resnick and H.R. Varian popularized their use [Resnick, Varian 1997]. Since then, many methods have been developed to enable the implementation of recommendation algorithms for web-based recommendation systems.

The purpose of this paper is to present the currently used recommendation methods and algorithms used for their implementation, along with the metrics used for testing the final solution. Although e-commerce recommendation systems are already widely used in practice, and the available literature provides a general description of their use, the details of the methods and algorithms have been discussed much less frequently and only superficially. This article contributes to fill this gap by focusing on the methods and algorithms of recommendation, and by facilitating the choice of solutions that best meet the requirements of specific applications.

With the ongoing development of recommendation systems, the approach to describing their functions and designing their operativity has been changing constantly. Consequently, several definitions of a recommendation system have been defined over

the years. In 1997, P. Resnick and H.R. Varian [Resnick, Varian 1997] defined a recommendation system as a tool, with the use of which users can mutually recommend products or services to each other. Since then the role played by recommendation systems has changed. Recommendations of products and services for users have become automatically generated by systems. This has resulted in a substantial change in the definition of a recommendation system. In 2002, R. Burke [Burke 2002] described a recommendation system as any system that, in a personalized way, presents the user with interesting or useful items chosen from a large range of options. Although the definition proposed by R. Burke is quite wide, it clearly defines the function of a recommendation system. The purpose of a recommendation system is to support users in the decision-making process regarding which product to buy, what song to listen to or which article to read [Ricci et al. 2011]. This means, that the recommendation system is recommending the products or services that suit the consumer's preferences. Another definition cited in the literature emphasizes that a recommendation system is an application or a tool which suggests items of interest to the user, based on prior knowledge about the user's preferences [Martin et al. 2011]. This description refers to the application of machine learning algorithms and data mining solutions which are used in today's recommendation systems. The above definition seems to describe the existing systems very well, and therefore it is in this sense that the recommendation systems will be understood in this paper.

In this article the term *item* (according to the understanding of this concept in the literature on recommendation systems) will be used to refer to any item that can be recommended [Ricci et al. 2011]. Another term of generally accepted meaning used in the literature which will appear in this article is the term *active user*, denoting the user for which the recommendations are prepared.

The online retail store *Amazon.com* is considered a pioneer of the commercial use of recommendation systems [Schrage 2008]. On a website that contains a description of a given product, a list of similar items bought by other users appeared; this solution is also used nowadays. The economic success of *Amazon* has encouraged other companies and websites to implement similar recommendation systems, since they may constitute one of the reasons for this company's successful development.

2. Recommendation methods taxonomy

Progress in the development of recommendation methods has been achieved gradually. This coincided with subsequent waves of innovation changing the functioning of the Internet, taking the form of the so-called Web 1.0 and Web 2.0 [Martin et al. 2011]. The currently applied solutions vary primarily in the type of information which is used in generating the final result. The range of the input data that can be used is very broad. Information about items subject to recommendation, the characteristics of the users of the system, and their mutual relations can be used.

The currently used methods can be classified as [Melville, Sindhwani 2010]: (1) *Collaborative Filtering* methods (CF), (2) *Content-Based Filtering* methods (CBF) and (3) hybrid methods.

Collaborative filtering means generating recommendations based on information collected from other users of the website (i.e. items' ratings submitted by other users). This is a popular solution used by most sites allowing to rate items. It applies to different types of web portals, regardless of the type of items (products, services, or content) that are subject to the recommendation. The desired information can be provided as an item rating, as well as the user's activity on the site. On this basis, the aim is to identify possible patterns according to which users select items. The idea underlying collaborative filtering is the concept of so-called collective intelligence, based on the assumption of the superiority of intelligence and knowledge that comes from the cooperation of individuals. Bearing in mind that the number of items for which a single user can have knowledge of is relatively small, he/she can benefit from the knowledge of others. They may be aware of other products or services that will be of interest to the active user. The CF method can thus be compared to asking your friend to recommend a good movie or a book which you are not familiar with.

There are many algorithms for the implementation of collaborative filtering. J. Breese, D. Heckerman, C. Kadie defined two categories of these algorithms [Breese, Heckerman, Kadie 1998]: *neighborhood-based methods* and *model-based methods*. In modern literature, neighborhood-based algorithms have been consolidated with other, newer algorithms, in a category called *memory-based* methods.

CF is popular mainly due to the fact that it works well in the case of unstructured problems. It can be used to generate recommendations even in the case of data which is difficult to analyze computationally, such as news and reviews, containing unstructured text. Thanks to the knowledge of the subjective preferences of multiple users, the method allows to recommend an item to a user who did not even know about its existence. CF allows to generate recommendations of items, whose characteristics differ significantly from previous consumer's choices, while quite unexpectedly they meet the consumer's preferences.

The second group of the above mentioned methods is Content-Based Filtering (CBF). This is a method of recommending items where the information about the characteristics of items or users is used. In this method, attention is focused on the specifics of individual items. CBF makes comparisons of the information about an item with information about the preferences of an active user. Given the information that a user likes an item of specific characteristics, we can search a database to find items that are most similar to it. For example, knowing that you liked the movie "Saving Private Ryan" – a war drama, directed by Steven Spielberg, the system could recommend the movie "War Horse" – another war drama directed by Spielberg, which would most likely also suit your taste. Having extra knowledge about the

user (e.g. demographic information) and the item (e.g. genre), it becomes possible to significantly improve the quality of recommendations [Melville, Sindhwani 2010].

Content-based filtering methods use multiple algorithms. They are divided into two basic categories: *classification algorithms* and *information retrieval algorithms* (IR). Classification algorithms are based on the use of data mining algorithms, such as *k-nearest neighbors, decision trees, naive Bayesian classifier* and *neural networks*. IR algorithms are used primarily to recommend items with descriptive text information, such as websites, books etc. Characteristics linked to the user's preferences are treated as queries based on which items are evaluated in terms of relevance and similarity [Melville, Sindhwani 2010].

Content-based filtering methods are often used on smaller sites that do not have comprehensive information about their users. Their disadvantage is the lack of flexibility with regard to the identification of the similarities between items of different categories. Relying only on the characteristics of items poses a serious limitation, if we intend to recommend items from a different domain than the one from which the user chose items before. Most shops offer items from many different fields, such as books, electronic devices, household appliances and stationery. The information that the user enjoys reading Swedish mystery stories is not able to help in recommending such items as mp3 players, video cameras, electric kettles or luxury pens.

Hybrid methods – the last group mentioned in this article – is a group of methods that allows to overcome the limitations of each of the previously discussed methods in order to take advantage of combining the use of information about the relationships between users (using CF) with information on the characteristics of items (CBF). There are various ways of combining methods [Gediminas 2005]: carrying out recommendations separately using CF and CBF, and then combining the lists of received results; incorporating CF into a CBF model; incorporating CBF in a CF model and defining a model which equally balances the characteristics of both approaches.

3. Recommendation methods and algorithms

Presenting recommendation methods requires the inclusion of the general information about algorithms that are needed for their implementation. At this stage it is possible to examine the algorithms used more carefully. Most collaborative filtering algorithms can be classified into two categories: *memory-based* and *model-based* [Su, Khoshgoftaar 2009].

Memory-based algorithms focus on the information contained in the database of items and/or users, that allows to capture the similarity between items and the similarity between users. Model-based algorithms are defined as algorithms based on statistical models created in the process of training/learning. The recommendation methods implemented in recommendation systems can combine both categories of algorithms in order to obtain the best quality of prepared recommendations.

Algorithms belonging to the category of memory-based algorithms prepare an estimate of an item rating, based on the information contained in the database or the so-called "memory". An example of this category of algorithms can be *the k-nearest neighbors algorithm,* which allows to use one of the correlation measures of similarity between users. Initially, a certain group of users is selected, based on their similarity to the active user (users with the highest score on similarity form a group of the active user's neighbors). Then, ratings of items issued by the users are weighted and used to generate recommendations.

Besides the already mentioned *k*-nearest neighbors algorithm, the category of memory-based algorithms also contains: *the Pearson correlation coefficient* (used for example to calculate the similarity between users), *Spearman's rank correlation coefficient* (used for calculating similarity using ranks, meaning places on the list of favorite items, instead of ratings posted by users), *cosine similarity measure* (used for comparing text documents and product descriptions), *the weighted average,* the less known *Tanimoto coefficient* (used to compare the similarities between the two sets, such as shopping lists of two users) [Kwon et al. 2009], *the weighted average entropy of information* (valued in estimating similarity between users).

Algorithms of the model-based category are based on the estimation of the parameters of statistical models, regarding the ratings issued by users. Most model-based algorithms use *latent variables* or *matrix decomposition.* Models containing *latent variables* follow the assumption that variables not observed directly may affect the observed variables which are used in the model to explain certain dependence. In the case of recommendation methods, usually it is assumed that both the similarity between users and items is induced by the same *low-level factors,* the basic characteristics of any user or item. For example, the rating of the film is the result of personality traits and some psychological determinants of the user, expressed in a positive inclination towards a specific movie genre or actors [Melville, Sindhwani 2010].

Algorithms related to the decomposition of a matrix (also called *matrix factorization*) constitute a group of algorithms which allow the presentation of the initial matrix in the form of a product of several matrices with specific properties. The use of a matrix decomposition has become popular in recent years, primarily because of its versatility, due to the possibility of use in various fields of recommendation, and scalability for large data sets [Koren, Bell, Volinsky 2009]. In the case of recommendation systems, matrix factorization corresponds to a decomposition of the ratings matrix into matrices containing information about the correlation of users and items with previously mentioned low-level factors. As a result of a matrix decomposition, we obtain two matrices with vectors containing the characteristics of users and items with respect to the low-level factors. The content of the obtained matrices has no direct interpretation and is simply a quantitative representation of user preferences and characteristics of items. The above-described algorithm is known as singular value decomposition (SVD). Matrices obtained by singular

value decomposition allow to calculate a prediction of the rating of individual items for individual users as a result of a multiplication of vectors representing the characteristics of items, and vectors representing the characteristics of users showing interest in the items. A detailed description of the algorithm, along with the problems associated with its implementation, such as the fact that the ratings matrix is usually incomplete, can be found in [Kosieradzki 2013, pp. 28-31]. Other algorithms used in model-based recommendations include Bayesian networks, algorithms based on naive Bayesian classification, clustering algorithms, regressions (mainly logistic regressions) and Markov Decision Processes (MDP).

Due to the limited size of this article, memory-based algorithms as well as model-based algorithms are only briefly mentioned. Their detailed description, possible applications and problems can be found in [Kosieradzki 2013, pp.18-37].

4. Problems of testing the suitability of recommendation algorithms

Algorithms have different suitability of use, mainly depending on the tasks to be fulfilled and the type of data which will be used. Before deciding on an algorithm to use for the recommendation, it is advised to check its efficiency as well as any other suitable combination of algorithms on the test data set. Algorithm testing is based on a comparison of the results obtained in the process of rating prediction and the classification of items by the use of the selected recommendation method, with real ratings or results obtained using a different method.

The selection of the test data set is one of the fundamental decisions that can influence the choice of an appropriate algorithm to develop a recommendation system providing advisable recommendations. If one does not possess a database of ratings, for example when creating a recommendation system for a new website, it is possible to use a publicly accessible data set. The selection of an appropriate data set ensures that the test results will properly represent the suitability of the algorithms which will be applied in the recommendation system. Herlocker, Konstan, Terveen, and Riedl distinguish three groups of features of the data sets that need to be taken into account when selecting a data set to test recommendation algorithms [Herlocker et al. 2004]: *domain, inherent* and *sample*.

The features corresponding to domain properties of the data relate to the information contained in the data set. Test data should reflect as closely as possible the information that the recommendation system will eventually use, especially the data subject and the context in which ratings are issued.

The inherent features of the data set arise from the specific methods of gathering information and denote the characteristics of the accumulated information, such as how interest is registered, the registration date and time of a rating, user demographics, and the selectiveness of ratings (range of subjects, which users rate).

The last group of features of a data set that we should take into account are the sample features. These features include: the density of a rating matrix – for example expressed as the average percentage of items rated by a single user, the distribution of ratings between users, the overall size of a data set and the distribution of variables.

Thanks to the cooperation of individuals involved in the development of algorithms used in recommendation systems, a number of publicly available data sets can be found on the Internet. In many cases, data sets are collected by academics seeking to create the standardization of research on recommendation systems. Due to publicly available data sets, many scientists use the same data which makes it possible to compare objectively the performance of the proposed solutions and facilitate research on recommendation systems and the use of appropriate algorithms and methods for the needs of a specific application.

5. Measures of effectiveness of applications of methods and recommendation algorithms

Measures of effectiveness of recommendation methods and algorithms presented in the literature are used to assess quality in terms of three criteria: **prediction accuracy**, understood as the ability to accurately predict the ratings of items (the correspondence between actual and forecasted ratings of items), **classification accuracy**, representing the ability to correctly classify items (correctly allocate items to the sets of items preferred and not preferred by a user) and **rank accuracy**, relating to the ability to create a proper ranking of items which takes into account the active user's preferences (the correct ordering of items in order to suit one's taste).

It is emphasized that the primary, and in many cases the most important criterion, is the assessment of the prediction accuracy. This can be justified by the fact that the rating prediction may convey information concerning the value of a recommendation of a specific item to the user. A high rating may suggest suitability to the user's preferences, as well as the high quality of an item and its popularity.

The next criterion is the accuracy of the classification of items into groups of relevant items and irrelevant items, that is, those that do and do not meet the user's preferences. This criterion is very important, especially if the recommendations are prepared on a regular basis for the same user. For example, when the user constantly selects items belonging to the same category (e.g. movies), recommending items from the user's favorite category can be much more important than emphasizing an accurate rating prediction.

The last criterion, rank accuracy, denotes the preparation of a ranking of items in terms of their suitability to the user's preferences as accurately as possible. This criterion is fulfilled when an item classified in first position is better suited to the user's preferences than all the other items, and the same condition concerns each following item in the ranking. It can be seen that this criterion is becoming very

important in the selection of an item best matching the user's preferences out of a set of similar items.

Predictive accuracy metrics allow to assess the accuracy of ratings prediction on the basis of forecasted ratings generated using the training dataset. Ratings estimated by using recommendation methods and algorithms, i.e. the forecast of the rating of a selected item, is compared with the real rating of an item in the test dataset.

As a measure of prediction accuracy, a *mean absolute error (MAE)* can be used. Mean absolute error is an average of the absolute value of the difference between the forecasted and real ratings issued by the users. It is not the best measure when the most important objective of a recommendation is to match only the top items to the user's preferences. MAE is the average error of all the ratings in the test set. Larger prediction errors for the most accurately matched items may be offset by a smaller error of ratings for items corresponding far less to the user's preferences. Many variants of this measure are used, such as the *mean square error (MSE), root mean square error (RSME), normalized mean absolute error (NMAE).*

Among the above-mentioned measures, one of the most frequently described and applied to assess the accuracy of the prediction of issued recommendations is the *root mean square error (RMSE)*. RMSE is the root of the mean of the squared difference between the predicted and the real assessment issued by the user. This measure focuses on individual predictions, strongly deviating from the real values of ratings.

Measures of classification accuracy (called *classification accuracy metrics*) examine the effectiveness of the recommendation methods and algorithms in terms of their ability to differentiate between relevant items which correspond to the user's preferences and irrelevant items not suited to the user's preferences or interests. Classification accuracy metrics are used to verify that the methods and algorithms correctly classify items into relevant and irrelevant, and also provide a valuable clue if proper items are chosen to be recommended.

Classification accuracy metrics are based on the determination of the number of items that are classified correctly. After determining the number of items that have been classified properly, we check if the recommended item matches the preferences of the consumer, and whether relevant items were recommended.

The two most important classification accuracy metrics are *precision* and *recall*. Precision (also called *TPA – true positive accuracy*) is defined as the ratio of relevant recommendations to the general pool of recommended items [Schroeder et al. 2011]. Recall (also called *TPR – true positive rate*) is a measure calculated as the ratio of relevant items classified correctly to the total number of relevant items. Recall is sometimes interpreted as the probability that an item corresponding to the user's preferences will be recommended. It is noted that precision and recall are inversely proportional. Increasing the number of items which fall within the group of recommended items typically increases the value of recall and lowers precision. Tests of the accuracy of a classification of items belonging to the group of irrele-

vant and not recommended items can be carried out using measures that are the inverse functions of precision and recall, called *inverse precision* and *inverse recall*.

To measure the effectiveness of a recommendation related to the classification accuracy, several metrics assessing the recommendation of irrelevant items can be used – such as the *fallout* and the *miss rate*. Fallout (also called *FPR – false positive rate*) is the ratio of incorrectly recommended items (recommended irrelevant items) to the total number of items not matching the user's preferences. Fallout can be interpreted as the probability that an irrelevant item will be accidentally recommended [Schroeder et al. 2011]. Miss rate (also known as *FNR – false negative rate*) is calculated as the ratio of items that have not been recommended, despite the fact that they suit the user's preferences (not recommended relevant items) to all the relevant items. This indicator can be interpreted as the probability that a relevant item will not be recommended.

Generally, a precise prediction of a rating is primarily important in cases of items that best fit the user's preferences. The lower an item is ranked among items chosen to be recommended, the less important the accuracy of the prediction of a rating is. Given the above, the use of the *average precision* (*AP*) and the *mean average precision* (*MAP*) has been proposed. These measures use the precision metric, but they take into account the rank that an item received among other recommended items. AP is a measure calculated simultaneously for all items recommended to one user. After calculating the average precision for each user, MAP, which is a measure of the average value of AP, can be calculated. By using MAP it is possible to compare the performance of a variety of methods and algorithms in independent recommendation systems [Schroeder et al. 2011].

In order to test the accuracy of a classification made by the methods and algorithms used in the recommendation system, the ROC curve, known for its many statistical and econometric applications, is often used. If you perform a recommendation, the ROC curve serves as a visual presentation of the relationship between fallout and recall. ROC shows the extent to which a good classifier selects relevant items and ignores irrelevant items – maximizing recall and minimizing fallout. Using a ROC curve instead of a descriptive analysis of quantitative results allows to perform a graphical analysis. The horizontal axis represents the measure of fallout, and the vertical axis depicts the value of recall. The better these methods and algorithms segregate items, the more concave the ROC curve is [Hernandez del Olmo, Gaudioso 2008].

Rank accuracy metrics is a group of measures that assess the arrangement of items in terms of user preferences - from the best suiting user's preferences to the least. As a measure of accuracy of ranks established by methods and algorithms of a recommendation system, statistical measures of rank correlation are usually used. Rank accuracy metrics, in contrast to prediction accuracy and classification accuracy measures, take into consideration only the ranking of items based on a chosen criterion. The accuracy of a predicted rating of an item, in comparison to the actual

user rating, does not affect the evaluation of the rank accuracy. Only the position of an item in the ranking in terms of a predicted rating is taken into account. Each of the listed items should have a higher real rating than all the items classified on a lower position, which means better suitability to the user's preferences. This means that if an algorithm constantly underestimates the rating of all the items (e.g. for each item it provides a prediction of a rating one degree lower than the real rating), but the order is maintained, the efficiency of the algorithm will be considered the same. Rank accuracy metrics are among the most commonly used measures for determining the similarity between users.

An interesting example of a rank accuracy metric is the measure of *half-life utility*. This is a measure designed to assess the ordering of items when users are not likely to be interested in items in distant positions in a ranking [Breese et al. 1998]. In the case of a recommendation of websites based on a query entered by a user (e.g. a search engine), a user usually checks only the initial elements of a generated list and if necessary clarifies a query to obtain more precise results without looking at more distant positions on the list of results.

A measure of *half-life utility* examines the usefulness of the ordering of items for the user (called *utility*), defined as the difference between a default rating of an item and its rating by the user. The default rating is usually defined as a neutral rating – corresponding to the center of the rating scale or located just below the middle of the scale. In cases of half-life utility, it is assumed that the probability of seeing an item on the list is decreasing steadily. To describe this relationship, generally an exponential probability function is used according to the so-called law of natural decay with a half-life parameter. The concept of half-life parameters was taken from physics, where it means the period after which the activity of the radionuclide is reduced by half. In cases of the recommendation methods and algorithms, the parameter mentioned above usually takes a value equal to the position of an item on the list (rank), which is expected to have a 50% chance to be used by the user. After determining the expected utility for each user, the expected total utility for all users (a common value of the half-life utility for all users together) is calculated. The formulae used in applying the half-life utility measure and its explanation can be found in [Kosieradzki 2013, pp. 62-63]. Thanks to the exponential distribution, the measure of half-life utility gradually reduces the weight which is attributed to the position of an item on the list. According to the half-life utility, recommendation methods and algorithms are regarded as good if they allocate the most relevant items on the top of the ranking list. However, if the selected probability distribution of the use of n-th item on the list is significantly different from the actual probability associated with the natural behavior of users, a serious error may occur. For example, if you always read the first 10 results (i.e. the first page of recommended items), the probability of checking results 1-10 will be 1, but the probability that you reach number 11 will instantly drop to 0. This sharp decline is not consistent

with the adopted assumption concerning an exponential probability distribution. It has been also shown that using a maximization function to calculate expected utility insufficiently punishes the inadequate distribution of clearly irrelevant items on the leading positions of the list [Herlocker et al. 2004].

Several measures have been developed that allow to check how unexpected and innovative recommendations are, from the point of view of a potential customer who possibly did not even know about the existence of certain items. The measure of innovation of recommendations is called the measure of *unexpectedness*. Assessment of the degree of unexpectedness of a recommendation, as well as the unexpected recommendation itself, do not have to be a useful indication to the user. New and recommended items have to be of interest to the user. A measure of the effectiveness of methods and algorithms, in terms of their ability to recommend items that are both unexpected and attractive, is an indicator of *serendipity*, meaning luck or an accidental discovery of something useful, especially while looking for something completely different [Matusiak et al. 2011]. In general, the proposed measure is the ratio of the sum of the utility of positively surprising recommendations, to the total number of recommended items.

6. Conclusions

This paper presents a number of methods and algorithms used to recommend items (products, services, content) within the e-commerce recommendation systems. Much attention has been devoted to testing the effectiveness of the application of recommendation methods and algorithms.

It should be emphasized that most algorithms and methods are not universal, but perform well within certain areas in various applications. Methods and algorithms serve different purposes and they give diverse results in terms of e.g. prediction accuracy, classification accuracy and rank accuracy.

Chosen methods can be combined to improve the effectiveness of their use [Amatriain et. al. 2011, p. 48]. Analysis of available recommendation methods and algorithms indicates that their further development requires the skillful use of achievements from many fields of science. This should include advanced methods popular in mathematics, statistics, econometrics, data mining, marketing, information technology, management and decision making. It should also not be forgotten to take into account the achievements of sociology and psychology, domains extremely relevant in the analysis of consumer behavior.

References

Amatriain X., Jaimes A., Oliver N., Pujol J.M., 2011, *Chapter 2: Data mining Methods for Recommender Systems*, [in:] F. Ricci, L. Rokach, B. Shapira, P.B. Kantor (eds.), *Recommender systems Handbook*, Springer Science+Business Media LLC,.

Anderson C., 2004, *The Long Tail*, Wired 12.10, October.

Breese J., Heckerman D., Kadie C., 1998, *Empirical analysis of predictive algorithms for collaborative filtering*, Microsoft Research, Redmond, USA, September.

Burke R., 2002, *Hybrid recommender systems: Survey and Experiment*, "User Modeling and User-Adapted Interaction", no. 12(4).

Gediminas A., 2005, *Towards the Next Generation of Recommender Systems*: *A Survey of the State-of-the-Art and Possible Extension*, "IEEE Transactions on Knowledge and Data Engineering", vol. 17, no. 6, June.

Herlocker J., Konstan J., Terveen L., Riedl J., 2004, *Evaluating Collaborative Filtering Recommender Systems*, "ACM Transactions on Information Systems", vol. 22, no. 1, January.

Hernandez del Olmo F., Gaudioso E., 2008, *Evaluation of recommender systems: A new approach*, "Expert Systems with Applications", vol. 35, Issue 3, October.

Koren Y., Bell R., Volinsky C., 2009, *Matrix factorization techniques for recommender systems*, "IEEE Computer Society", 0018-9162/09.

Kosieradzki D., 2013, *Analysis and an overview of methods used for building, implementing and testing recommender systems on the electronic market*, Faculty of Economic Sciences, University of Warsaw, Warsaw.

Kwon H.J., Lee T.H., Hong K.S., 2009, *Improved Memory-based Collaborative Filtering Using Entropy-based Similarity Measures*, Proceedings of the 2009 International Symposium on Web Information Systems and Applications (WISA'09), Nanchang, China, 22-24 May.

Martin F.J., Donaldson J., Ashenfelter A., Torrens M., Hangartner R., 2011, *The Big Promise of Recommender Systems*, "AI Magazine", vol. 32, September.

Matusiak K. et al., 2011, *Innowacje i transfer technologii. Słownik pojęć*, PARP, wydanie III, Warszawa.

Melville P., Sindhwani V., 2010, *Recommender Systems*, Encyclopedia of Machine Learning, Chapter No: 00338, April.

Resnick P., Varian H.R., 1997, *Recommender Systems*, "Communications of the ACM", no. 40(3).

Retail/E-Commerce Industry Report Q4 2011, iPerceptions Inc., 2012, http://www.iperceptions.com/files/Retail-E-Commerce-Industry-Report-Q4-2011-FINAL2.pdf (2.12. 2012).

Ricci F., Rokach L., Shapira B., Kantor P.B., 2011, *Chapter 1: Introduction to Recommender Systems Handbook*, [in:] F. Ricci et al. (eds.), *Recommender systems Handbook*, Springer Science+Business Media LLC.

Schonfeld E., *Click here for the upsell*, CNN Money, 11 July 2007, http://money.cnn.com/magazines/business2/business2_archive/2007/07/01/100117056 (18.11.2012).

Schrage M., 2008, *Recommendation nation*, MIT Technology Review, May-June.

Schroeder G., Thiele M., Lehner W., 2011, *Setting Goals and Choosing Metrics for Recommender System Evaluations*, UCERSTI2 Workshop at the 5th ACM Conference on Recommender Systems, Chicago, USA, 23 October 2011.

Su X., Khoshgoftaar T.M., 2009, *A Survey of Collaborative Filtering Techniques*, Advances in Artificial Intelligence, vol. 2009.

SYSTEMY REKOMENDACJI PRODUKTÓW I USŁUG HANDLU ELEKTRONICZNEGO. METODY I ALGORYTMY REKOMENDACYJNE ORAZ MIARY SKUTECZNOŚCI ICH STOSOWANIA

Streszczenie: Artykuł dotyczy systemów rekomendacji produktów i usług handlu elektronicznego, które wraz z jego rozwojem odgrywają coraz większą rolę zarówno dla konsumentów, jak i sprzedawców. W artykule przedstawiono metody rekomendacyjne, a także algorytmy umożliwiające ich realizację. Szczególną uwagę poświęcono problemom testowania przydatności algorytmów i miarom skuteczności zastosowań metod i algorytmów.

Słowa kluczowe: systemy rekomendacji w handlu elektronicznym, metody i algorytmy rekomendacyjne, miary oceny skuteczności rekomendacji.