An Experiment in Recommending Content from an Information Portal

Marián Mach, Jozef Stráňovský
Dept. of Cybernetics and Artificial Intelligence
University of Košice
Letná 9, 042 00 Košice, Slovakia
Marian.Mach@tuke.sk

Abstract. *There are manifold information sources producing new content offered through the Web. Information portals (especially those of information agencies) belong to most prolific sources - they produce new content seemingly continuously. They are typically not focused on a limited domain but cover a wide spectrum from actual events through sport and health to hobby and leisure time.*
User is naturally interested only in a portion of the produced content. He/she must filter out content in which he/she is not interested - and the filtering effort may decide whether he/she will return again or not. One possibility how to minimise the effort of this filtering is to use a recommender system which is able to recommend user those published information pieces which match his/her interests.

Keywords. content recommendation, document classification, user model

1 Introduction

Due to enormous amount of the information available in the Web, content recommendation is steadily a hot topic in content publishing and delivering. Traditional approaches tries to categorise the published content into different categories to make it for users more accessible. In order to rise the recommendation to a higher level, appearance of personalised private recommender systems is inevitable [1] [2]. They are expected to utilise a dynamic model of a particular user to recommend content to this user.

We have built a simple recommender on top of an information portal in order to experiment with it.

2 A recommender system

A simple recommender system was designed and implemented in order to test the feasibility of the idea of using text processing and classification of documents to recommend published news in accordance with interests of users. Therefore only the simplest methods for implementing the system were employed - the focus was on experimenting with the system.

Interests of a user are represented by a set of documents - examples of those documents which are interesting for the user as well as examples of the documents in which the user is not interested. The documents are stored in the training set of documents. This set is continuously updated based on feedback from the user - the user must define which documents are really interesting for him/her and which are not.

Based on the training set a model of a user can be developed. It has the form of a classifier which is able to classify documents to distinguish interesting documents from the others. This classifier is used to provide recommendations to the user on published documents. The classifier corresponds to the user's interests as they were known when the classifier was developed. If the interests change or become known in more detail, the classifier should be updated to reflect new knowledge on these interests.

2.1 Architecture

The overall architecture of a simple recommender system is depicted in Figure 1.

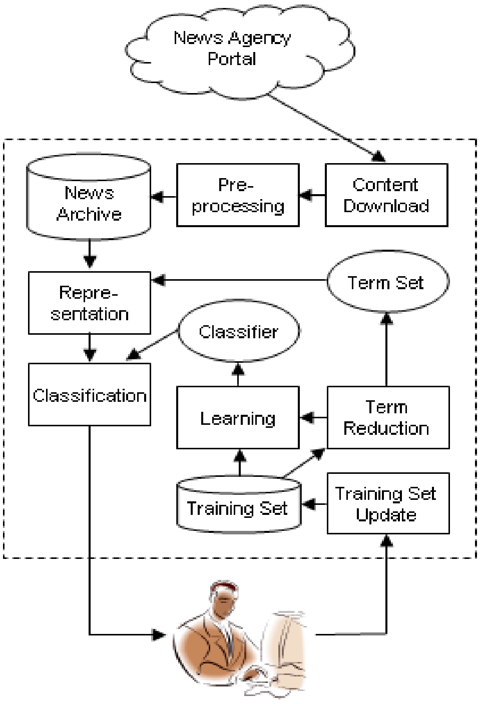


Figure 1: Architecture of a recommender system

The architecture consists of three main parts:

- Obtaining content - news published on a news agency portal are grabbed and after an initial preprocessing they are stored in an archive. Since news are published irregularly, the system periodically checks the portal whether there are some new news which could be obtained.
- Communicating with user - after a user accesses the system, news published after the last user's access are retrieved from the archive. They are subsequently classified as interesting or not interesting for the user and together with this recommendation they are displayed. If the user provides some feedback, the feedback can result in updating the training set.
- Update of user model - the changed training set may not correspond to a classifier or a set of classifier terms any more (the bigger change, the less correspondence). In order to update the model, a new set of terms is selected and subsequently a new classifier is developed. This update can be triggered in different ways - periodically, when a predefined change of the training set was reached, or when performance of the classifier on the training set drops below some threshold.

The following functional modules are included in the architecture:

Content download. The system uses an RSS channel (our previous attempt was based on a wrapper over published HTML code) which provides a list of several lastly published news. The module periodically reads the channel and compares it with the archive in order to identify whether some new news were published. If there are such news, they are downloaded.

Preprocessing. Downloaded documents are modified in a standard way used in text categorisation domain. The preprocessing is represented by a set of steps: transforming content to a lowercase representation, removal of punctuation and numbers, removal of stop words, and stemming. The preprocessed content is added to the original content and the given document is archived.

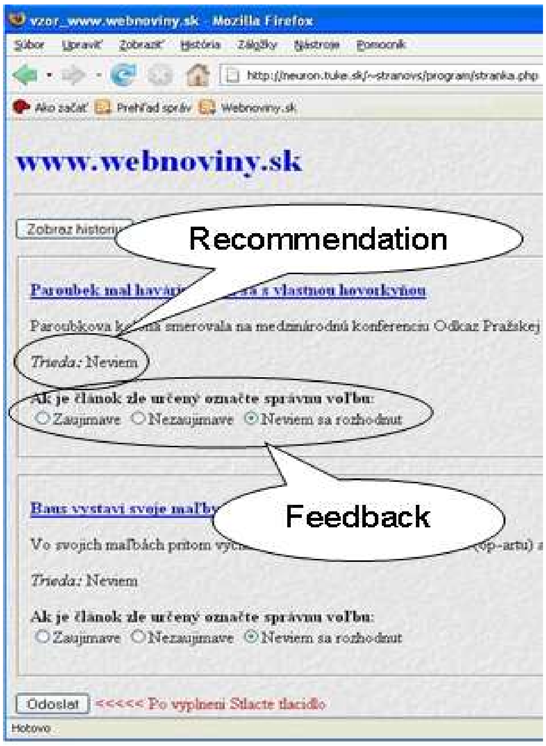
Representation. Each selected document is represented using a binary representation. The preprocessed content is matched with a set of selected terms . For each term from the set its presence or absence in the document should be determined - it results in a vector of binary values with the same number of elements as the term set (each element corresponding to one term).

Classification. Based on their binary representation, the selected documents are classified by a classifier. Since Naive Bayes is used in the role of classifier, the classification provides two probabilities - the probability that a document is interesting for a user and the probability that the document is not interesting for the user. If those two probabilities are close each other then the document is classified into class “Unknown” - no recommendation is given. Otherwise the document is classified into class “Interesting” or “Uninteresting” and this recommendation is added to the document.

Training set update. User's feedback can result in adding documents into the training set - if user provides correct classification of documents for which no recommendation was given or if user corrects the system's recommendations (documents were wrongly classified). The capacity of the training set is finite - if it was reached then adding new documents results in removing the same number of the oldest documents from the set.

Term reduction. The system can use only a reasonably small number of terms to represent documents. The terms should be selected from those which are present in the documents located in the training set. The reduction of the term set to a reasonable size is based on an information gain criterion.

Learning. Since Naive Bayes is used, learning has the form of calculating probabilities of classification classes and conditional probabilities of presence/absence of terms given classes. The calculation is based on the documents from the training set.



vzor_www.webnoviny.sk - Mozilla Firefox

Súbor Upraviť Zobraziť História Záložky Nástroje Pomocník

http://neuron.tuke.sk/~stranovs/program/stranka.php

Ako začať Prehľad správ Webnoviny.sk

www.webnoviny.sk

Zobraz historiu

Paroubek mal havári s vlastnou hovorkyňou

Paroubkova k smerovala na medzinárodnú konferenciu Odkaz Pražskej

Trieda: Neviem

Ak je článok zle určený označte správnu voľbu:
○ Zaujimave ○ Nezaujimave ● Neviem sa rozhodnut

Baus vystavi svoje maľby

Vo svojich maľbách pritom vyc (op-artu) a

Trieda: Neviem

Ak je článok zle určený označte správnu voľbu:
○ Zaujimave ○ Nezaujimave ● Neviem sa rozhodnut

Odoslat <<<<< Po vyplneni Stlacte tlacidlo

Hotovo

Figure 2: User interface of a recommender system

3 User interface

After user accesses the system, he/she is provided with a list of newly published news (original content is used) together with the system's recommendations. The interface is intentionally kept simple in order to be usable as much as possible. It is depicted in Figure 2.

User can click on the title of the selected news to display the whole content (currently he/she is redirected to the original portal) or can provide his/her feedback using a radio button. Default feedback setting is "Unknown" but user can change this to "Interesting" or "Uninteresting". In order to submit the feedback back to the system, user must click on a button located at the bottom of the page.

3.1 Parameter setting

Functionality of the system depends on two parameters - how many terms are used to represent documents and how big the capacity of the training set is. Very small values of these parameters implicate not very satisfactory operation of the system (based on low accuracy of used classifier). On the other hand, high values of these parameters result in unreasonable requirements on space and time resources. Therefore a balanced values have to be found.

Experiments with different values have been performed (similar experiments like those described in the next section). It was discovered that the capacity of the training set at least 500 documents and the number of terms at least 200 terms provide quite usable setting. Although higher values can make recommendations of the system better, the increase is only very small and it is not worth increasing requirements on technical resources.

4 Experiments

The main aim of the series of experiments we performed was the simulation of user behaviour with subsequent evaluation of the influence of this behaviour on characteristics of our recommender system. Different ways of user behaviour were simulated in order to test the recommender in different situations which can occur during real operation.

4.1 Experiment setting

The implemented recommender system was tested using a collection of documents obtained from a portal of the SITA news agency (`http://www.webnoviny.sk`). Each document represented an article published on the portal and all documents represented all the news which had been published during a given time period.

The portal classifies all published news into eight categories (each document is classified exactly into one category/class). Those categories are: Sport, World, Slovakia, Finance and business, Hi-Tech, Health, Show business, and Auto-moto. Since all the documents were pre-classified, it was possible to perform tedious lengthy experiments without the involvement of real users - we were able to simulate different types of users.

Every user has his/her domain of interest - a domain information from which can be regarded interesting for him/her. Therefore, some portal categories were selected as interesting for user. Union of these categories represented a class "Interesting" and each document belonging to one of the categories constituting the class was considered to be interesting (for the simulated user).

In addition, the subset formed from the other categories (disjunctive with the subset representing interesting information) represented a class "Uninteresting" - a domain in which user is clearly not interested. For example, such automatic document classification into two classes enabled to simulate a user with interests in sport - but all sport-related information without any exception. User's interests can be defined only by boundaries between the defined categories but not within the categories. Thus, it was not possible to simulate a user interested in a few sports only and not interested in the other sports.

The implemented recommender classifies documents into three categories: "Interesting", "Uninteresting", and "Unknown". When testing the system, only documents classified to the first two classes were used to calculate required numerical characteristics. Documents classified into the class "Unknown" were not taken into account since such documents cannot be matched with user's interests.

4.2 Experiment pattern

All performed experiments were designed according to one common pattern. Thus, all experiments shared the same structure and differed only by their goals and the ways how those goals were achieved within the used pattern.

Available documents were divided into document subsets. Each subset represented a set of documents which had been published after the last user log-on - that means those documents were expected to be provided to user after his/her new visit to the system. In order to make experimenting simple (unlike testing under real operation conditions), all subsets consisted of the same number of documents - thirty was selected as a reasonable number of documents in one subset. It roughly corresponded to everyday regular visit to the used news portal - a regular reader of the portal was in average exposed to this number of news every day during the considered time period.

Each experiment consisted of several cycles - one cycle corresponded to all activities related to processing one document subset (i.e. a cycle corresponded to one user's visit to the system). Those activities could be:

- classification of the documents from the subset into "Interesting", "Uninteresting", and "Unknown" classes
- calculating classification characteristics by comparing classification results with available document pre-classification
- providing user feedback for wrongly classified documents and/or for documents classified into the "Unknown" category
- updating the training set of documents using (a part of) obtained user feedback
- re-learning of the classifier if modification of the training set occurred

Not all activities were performed in each cycle. During the first cycle no classification was possible (there was no training set to develop a classifier) and during the last cycle there was no classifier update (since it would be useless). Typically, the beginning of the experiment tried to build a starting training set and to develop a classifier, the end of the experiment was focused on classification using an existing classifier, and the middle part of the experiment simultaneously paid attention to using existing classifier as well as to obtaining feedback to improve the classifier. But which activities (and in which form) were performed in which cycle depended on a particular experiment.

In order to evaluate quality of classification, a contingency table of the form depicted in Table 1 was produced (columns represent user interests and rows stand for results of classification).

Table 1: Experiment contingency table

	Interes-ting	**Uninteres-ting**
Interesting	TP	FP
Unknown	UP	UN
Uninteresting	FN	TN

The table enables to compute precision Π, recall R, and accuracy A. Since documents classified into the "Unknown" class were not considered for calculation, the following definitions of the characteristics were used:

$$\Pi = \frac{TP}{TP + FP} \tag{1}$$

$$R = \frac{TP}{TP + FN} \tag{2}$$

$$A = \frac{TP + FP}{TP + FP + FN + TN} \tag{3}$$

In order to grasp dynamism of an experiment, accuracy was calculated not only for the whole experiment but for each experiment cycle as well.

4.3 Experiment: ideal user

The experiment was focused on simulating behaviour of a user during almost eight months (233 days represented by 233 cycles). The experiment consisted of two phases - building and testing. The aim of the first phase was to build a proper training set of documents from those documents which were published during this phase in order to enable the recommender system to develop a classifier of high quality. This phase comprised 133 cycles (3990 documents were published during the period). The second phase was focused on classification of new documents published during this period. It was composed of 90 cycles (2700 documents were published).

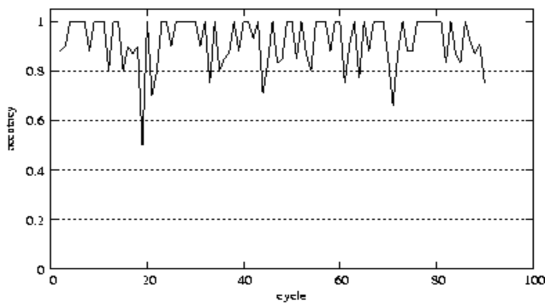


Figure 3: Classification accuracy for an ideal user

User was simulated in such way that he/she was interested in information about sport and was not interested in news about Slovakia. Since he/she follows only those two categories of news, other categories were not included into the experiment. The user was simulated as an ideal user - he/she provided feedback on each published documents during the whole experiment.

The result of the first cycle was an initial classifier which was improved during subsequent cycles. During the other cycles of the first phase, published documents were classified by a current version of the classifier, wrongly classified documents (based on user's feedback) were included into the training set and a new classifier version was built. This was repeatedly performed for each cycle of the first phase. The same activities were performed during the second phase - including modification of the training set and classifier update. The only difference was that classification characteristics were calculated in each cycle of the second phase.

During the second experiment phase 747 news were published within the two categories of interest. The classification achieved the following characteristics for the whole testing period:

$$\Pi = 0.95 \qquad R = 0.93 \qquad A = 0.93 \tag{4}$$

Classification accuracy for each cycle of the testing period is depicted in Figure 3.

A similar experiment was performed where the training set was updated not only with wrongly classified documents - all classified documents were included in the set. Since similar results were achieved, adding all documents into the set can be considered useless and this way of building the training set was rejected.

The experiment has proven feasibility of the idea on recommending published information to users - but only for ideal users. Unfortunately, there is only a very little probability that users are willing to provide feedback on each published information. Real users are too busy (or lazy) to provide such kind of feedback. In order for the recommender system to be usable, it must be prepared to deal with less ideal feedback.

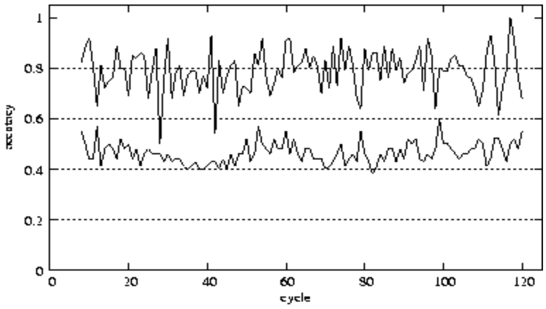


Figure 4: Classification accuracy for no feedback

4.4 Experiment: no feedback

The aim of this experiment was to evaluate operation of the system in situations when no feedback from user is available. After a short building phase when user was willing to provide feedback to the system for it to build a classifier, the user stopped his/her feedback. During the testing phase of the experiment the system was not able to update the classifier since due to a lack of feedback from user it was not able to update its training set. The same classifier was used during all cycles of the testing phase.

User was interested in categories Sport and World and not interested in categories Slovakia, Finance and business, and Show business. The other categories were not read by the user. The testing phase consisted of 133 cycles (3990 published documents). The building phase comprised 1 to 7 cycles.

Classification accuracy for the testing phase is depicted in Figure 4. There are two cases depicted - building phase of 3 cycles (below) and building phase of 7 cycles (above).

As it can be seen, a short building period enabled to collect only an inadequately small training set which resulted in a very serious drop of classification accuracy. Surprisingly, a few more cycles for building the training set enabled the system to increase the accuracy of its recommendation to a quite high level. According to common sense it is not reasonable to expect high quality recommendations without providing feedback sufficient to collect an adequate training set of documents. On the other hand, the building phase may not be necessary very long - the system is able to provide reasonable recommendations after a few cycles of collecting information on user interests.

Although the recommender system is not a suitable tool for a user who is not willing to provide any feedback on the system recommendations, the system is able to operate satisfactory during longer periods without feedback as well.

4.5 Experiment: sparse feedback

This experiment tried to reflect the fact that real users do not use the system in a uniform way during a number of system sessions (cycles). They are in different mood and/or there are different external conditions influencing users' behaviour. As a result, sometimes they are more willing to provide a reasonable feedback than at the other time. Such user attitude is reflected in the frequency of providing user feedback - within some cycles feedback is provided in a usual way while within the other cycles the recommender system receives no feedback at all. There are two extremes (an ideal feedback and no feedback), but behaviour of a real user is expected somewhere between these extreme poles. The proportion of providing to rejecting feedback can characterise a particular user - some user is closer to an ideal user while another user provides feedback only very scarcely.

The question is, how system operation is influenced by the frequency of feedback submission. To test it, user provided his/her feedback on system recommendations randomly with some probability. The probability 0.2 meant that, in average, the system received information from the user in 20 cycles from each 100 cycles - but the distribution of the cycles with feedback among all cycles was random.

Similarly to the previous experiment, user was interested in categories Sport and World and not interested in categories Slovakia, Finance and business, and Show business. The other categories were not read by the user. The testing period consisted of 101 cycles (3030 published documents). The building phase comprised 1 cycle only. That means that the classifier resulting from this period was of a rather low quality and in order for operation of the system to be satisfactory, the system was expected to increase the quality of its recommendations based on receiving an additional information on user's interests.

During the testing phase periods when the same classifier was used within more cycles were mixed with periods when the system's classifier was updated. The frequency of these changes and duration of the periods were randomly distributed - the less probability of user response, the longer periods without classifier update. The testing phase consisted of 100 cycles.

During the testing experiment phase 2067 news were published within the two categories of interest. When using probability 0.2 as the probability of user response, the classification achieved the following characteristics for the whole testing period:

$$\Pi = 0.72 \qquad R = 0.81 \qquad A = 0.71 \qquad (5)$$

The same characteristics for the case when the probability 0.8 was used were:

$$\Pi = 0.79 \qquad R = 0.89 \qquad A = 0.79 \qquad (6)$$

Classification accuracy for each cycle of the testing phase for the probability of providing feedback equal to 0.2 is depicted in Figure 5.

The achieved results corresponded with our intuitive expectations - the higher probability, the more successful recommendations of the system. In addition to it, the recommender system has proven a quite robust behaviour - differences in system behaviour using such different values for the probability of feedback as 0.2 and 0.8 were smaller than we had expected.

As it can be seen from the graph, accuracy at the beginning of the testing phase was quite low - the training set of documents after the very short building phase was unsatisfactory. Based on subsequent user feedback during next 30 cycles, the quality of the training set were increasing which resulted in increasing the quality of the system's recommendations. The higher the probability, the shorter this period - the system was able to collect a proper training set sooner than in case of more sparse feedback from user.

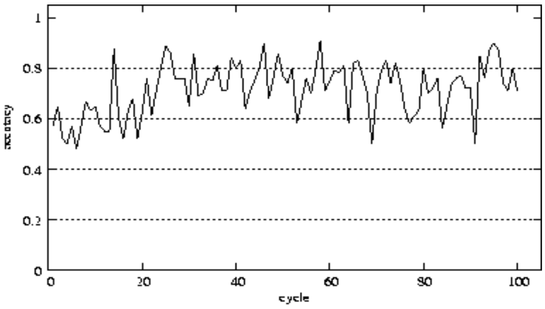


Figure 5: Classification accuracy for sparce feedback

4.6 Experiment: biased feedback

In the previous experiments, when user provided his/her feedback, he/she expressed his/her opinion on each document published within a given cycle. The aim of this experiment was to test a limited form of feedback offered to the system - the system received the user's opinion on only some of the documents published within each cycle.

Categories World, Sport, and Health represented user's domain of interest. The other categories represented the categories user was not interested in. User evaluated all news published during the building phase (full feedback) in order to initiate operation of the system properly. After a short building phase (1 cycle only), user provided his/her feedback in each cycle of the testing phase (100 cycles) - but the user provided the feedback in a limited form.

The feedback of user was limited on those published news which were classified into the class "Unknown" - the system was not able to determine whether user is interested in them or not. During the testing phase accuracy achieved 0.46 only. Classification accuracy for each cycle of the testing period is depicted in Figure 6.

The graph (and overall accuracy as well) signalises that feedback limited on only those news the system is not able neither to recommend nor reject is not sufficient for successful operation of

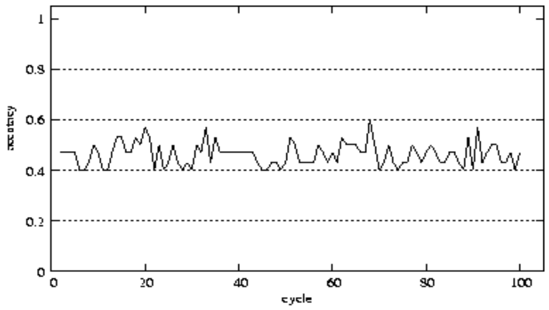


Figure 6: Classification accuracy for feedback biased to unclassified news

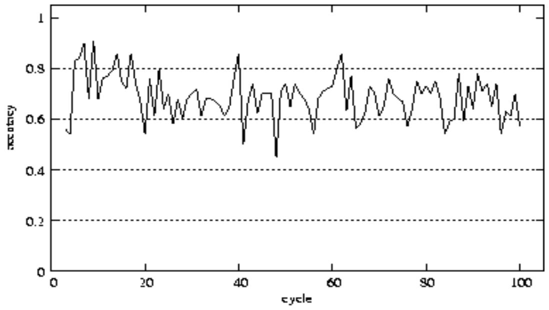


Figure 7: Classification accuracy for feedback biased to interesting news

the recommender. After the short building phase the training set was inadequate (as expected), but the system was not able to collect a more appropriate training set and to increase the accuracy of its recommendations.

A similar experiment was performed with another form of limited feedback. User marked only those published news which he/she regarded to be interesting for him/her (no matter to which class those news were classified by the system). Achieved classification accuracy was higher than in the previous form of limited feedback - almost 0.68. Classification accuracy for each cycle of the testing phase is depicted in Figure 7.

Superficially, this form of feedback seems to be acceptable. But a problem is hidden in the structure of the training set - majority of documents located in the training set belongs to class “Interesting” while class “Uninteresting” is represented insufficiently by a small number of documents only (obtained during the building phase of the experiment). As a result, recall was 0.95 for news from interesting categories while it was only 0.25 for documents from the other categories.

The experiment has proven that biased feedback can heavily influence the operation of the recommender system - it can cause the system infeasibility or decrease of the quality of produced recommendation at least. In order for the system to be a really useful tool, user should reflect on both false positive as well as false negative miss-recommendations.

5 Conclusions

Our experiments have shown that a recommender system can operate quite satisfactory in a real setting under conditions which are far from ideal - occasional lack of feedback, feedback provided only sparsely, or biased feedback. Although implemented in a very simple way, it can be successfully used as a personal tool to decrease information overload.

6 Acknowledgements

The work presented in the paper was supported by the Slovak Grant Agency of the Ministry of Education and Academy of Science of the Slovak Republic within the 1/4074/07 Project "Methods for annotation, search, creation, and accessing knowledge employing metadata for semantic description of knowledge".

References

[1] Chen T, et al.: Content recommendation system based on private dynamic user profile, Proc. of the 6th Int. Conference on Machine Learning and Cybernetics, Hong Kong, August 2007, pp. 2112-2118.

[2] Yamamoto N, et al.: Recommendation algorithm focused on individual viewpoints, Proc. of the Consumer Communications and Networking Conference, Las Vegas, NE, USA, January 2005, pp. 65-70.