Social Recommender Systems An Influence on Public Media

CONFERENCE PAPER · APRIL 2015

Gypsy Nandi
Assam Don Bosco University

Social Recommender Systems
An Influence on Public Media

Bornali Borah[1], Priyanka Konwar[2], Gypsy Nandi[3]
School of Technology, Assam Don Bosco University, Guwahati, Assam, India
{bornali.borah673[1] & pkonwar00000[2] & gypsy.nandi[3]}@gmail.com

Abstract **- Nowadays, recommendation plays an increasingly significant role in our daily lives. Recommender systems cover up a huge variety of applications related to large-scale social networks, such as online social network recommendations for a product, image or friend. This facility in social networks has become very popular due to the easy access of information on the Internet. For example, recommender systems may give suggestion to an online user all the items or products that might be of concern to him or her. Such systems are mainly based on collaborative, content-based and/or hybrid filtering techniques. In this paper, we mainly focus on the standard techniques used for recommendation of items or products in online social networks that help an online user to get influenced and choose those particular products or items.**

Index Terms - Online Social Network, Recommender system, Collaborative filtering, Content-based filtering, Hybrid filtering

I. INTRODUCTION

Recommender Systems (RS) provide recommendations to online users with reference to a set of articles, products or services they might be interested in. This facility in online social networks has become very popular on the Internet. There are various applications used in websites for recommendation of items such as movies, books, music, etc. The information can be acquired typically by collecting users' ratings or by observing the users' actions, such as songs heard, websites visited and books read.

RS make use of various sources of information for providing users with predictions and recommendations of items. With the advent of the Internet, RS implementation has facilitated its use in diverse areas. Over the last few years many algorithms and systems came into existence which was developed by the researchers working in this field. These systems predict user preferences for new items mainly based on the user's past ratings on other items.

There are some basic considerations to be made for building a recommender system, which is mentioned below:

- The filtering algorithm(s) to be used (e.g., content-based, collaborative, hybrid, etc.)
- The data available in database (e.g., user and/or item information, ratings for each item, etc.)
- The choice of model (e.g., memory-based, model-based and hybrid)
- The specific techniques to be used (e.g., Bayesian networks, probabilistic approaches, etc.)
- Performance of the system (e.g., running-time, memory consumption, etc.)

Figure 1 below illustrates an example of movie RS in which ratings from users on movies are taken and a matrix of user and movies are formed so that the recommender system can recommend the top –N recommendation which is based on those similar users rating.

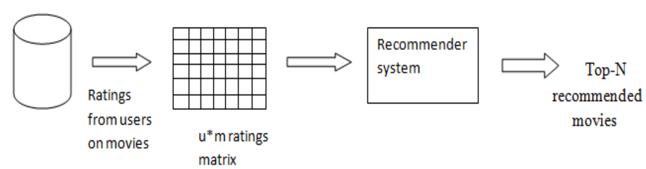

Fig. 1 A Traditional Movie Recommender System

II. STANDARD RECOMMENDER SYSTEM TECHNIQUES

There are mainly two types of methods used for recommender systems - content-based methods and collaborative filtering. Content-based methods determine the likeness of the recommended item to the ones that a target user likes or dislikes based on item attributes. On the other hand, collaborative filtering is a method of making automatic predictions (filtering) about the interests of a user by collecting information about similar preferences or taste from many users. It finds users who have similar tastes with the target users based on their past ratings. Hence, it will then give recommendations to the target user based on the suggestions of those similar users.

Another filtering technique, known as demographic filtering, mainly emphasizes on the fact that users with common personal attributes (such as, age, place, etc.) will have

H. K. Kalita, P. K. Mallick, I. Hussain, B. Bhuyan, S. Roy (Eds.): I3CS'15
978-1-4799-5875-01/15©2015 IGI

common preferences. For good results, recommender system combines content based filtering and collaborative filtering or demographic filtering and collaborative filtering to form hybrid filtering techniques where it can overcome all the problems faced by individual filtering techniques.

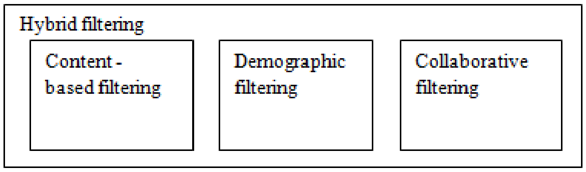

Fig. 2 Standard Recommender System Techniques

Figure 2 above shows the standard RS techniques used for various recommendations for items or products. All these techniques are further explained in detail below.

A. *Collaborative Filtering (CF):*

CF provides suggestions to an online user or customer based on the concept that one who had shown some preferences or tastes for a product in the past, will continue to have similar tastes for the product in the future also. The workflow of a CF system can be typically classified into the following steps:

a) A user rates items (e.g. books, movies, etc.) available in the system to express his/her preferences or dislike for the item. These ratings can help analyze and understand about the user's interest in the corresponding domain.
b) The system compares the ratings of this user with other users available on the system and tracks people having the most “similar tastes”.
c) The system then recommends an item for a user based on these “similar tastes” of users such that all these similar users have rated the item highly but the latter has not yet been rated by this user.

Thus CF is a way of building usual predictions about the taste of a user by collecting preferences from many users. Moreover, CF can be further classified as memory-based algorithms and model-based algorithms. Memory-based operates over the whole user’s database to make predictions for users. These methods usually use similarity metrics to find the distance between two items or two users based on each of their ratios. Model-based algorithms, on the other hand, use the user’s database to learn and create a model which is then used for prediction of items. Model-based technique is mainly used to make predictions for real-time data. This paper focuses mainly on memory based algorithms.

Memory-based algorithms make use of the entire user-item database to make a prediction for recommendation. These systems try to find a set of users, known as *neighbors* that have a history of agreeing with the target user. This is based on how they have either rated different items similarly or they have bought similar sets of items. Once a neighborhood of users is created, these systems use different algorithms to produce top N recommendation for the active user. Basically the memory-based collaborative filtering can be represented in two ways:

1.1 K-Nearest Neighbor (KNN)
1.2 Top N Recommendation

1.1 K-Nearest Neighbor based Collaborative Filtering:

In order to obtain predictions and recommendations using neighbor–based CF we choose a subset of neighbors of active users. The criterion for choosing a subset is based on the similarity of users with active users. Then compute the weighted aggregate for their rating to generate recommendations. Neighbor–based collaborative filtering consists of the three main steps [10].

a) All users will be weighted and similarity is computed with respective to the active user.
b) Subsets of user called as predictors will be selected.
c) Rating is normalized and the weights of selected neighbors are combined with rating to make prediction.

The next subsection discusses about user based and item based filtering. Item-based collaborative filtering is a model-based algorithm for producing predictions to users. In this algorithm, the similarities between different items in the dataset are calculated by using one of a number of similarity measures, and then these similarity values are used to predict ratings for user-item pairs not present in the dataset .User based filtering is done based on the suggestion on similar users in a system.

1.1.1 User based K-Nearest Neighbor:

User-based K-nearest neighbor algorithms generate predictions for users based on ratings from similar users called as neighbors. This approach was proposed in the end of 1990s by the professor of University of Minnesota Jonathan L. Herlocker [8]. First, the similarities between the target user and all other users who have rated the target item are computed which is most commonly done by using the Pearson correlation. Here a set of users known as neighbours has to be find and once the neighbors are discovered then the system combines preferences of neighbors to produce predictions or top n recommendations for the active user.

A popular similarity measure used in user-based CF is Pearson correlation. In Pearson Correlation, the correlation between users a and b is calculated as follows:

$$w(a,b) = \frac{\sum_j (v_{aj} - \bar{v}_a)(v_{bj} - \bar{v}_b)}{\sqrt{\sum_j (v_{aj} - \bar{v}_a)^2 \sum_j (v_{bj} - \bar{v}_b)^2}}$$

The users a and b record votes over items by summing up over item j. The weight is calculated in such a way that it gives the value between -1 and 1. A value of -1 specifies that the two users have rated the item accurately opposite while the value of +1 specifies that the two users have rated the item in exactly the same way. Algorithm 1 below explains the step-wise procedure used for returning top-N items from a list, which is sorted by the average of the evaluations.

Algorithm 1: User-based CF

1. For each user n (that is not m)
   1.1. Compute the similarity between m and n
   1.2. Add the most similar users to the list L of neighbors
2. For each item i evaluated by a user in L but not evaluated by u
   2.1. For each user n in L that has evaluated i
      2.1.1. compute the similarity between m and n.
      2.1.2. multiply the evaluation of n for item i by weights($a_{n,i} = a_{n,i}$*s)
      2.1.3. include the weighted evaluation into the mean of the evaluations for item i
   2.2. Add the mean of the evaluations of item i to list A
3. Return top-N items from list A

1.1.2 *Item based K-Nearest Neighbor:*

This approach was proposed by the researchers of University of Minnesota in 2001 [12]. In this case, a model of user ratings is first developed and then item recommendations are provided by computing item-item similarities. The item-based approach looks into the set of items the target user has rated and determines how similar they are to the target item and then selects the most similar items. Once the process is over, the prediction is then calculated by taking a weighted average of the target user's ratings on these similar items. Algorithm 2 below explains the step-wise procedure used for a user to rate two items a and b and record the ratings as well as their similarities.

Algorithm 2: Item-based CF

1. For each item a
2. For each user U who rate a
3. For each item b rated by user U
   3.1 Record that a user rated a and b
4. For each item b
   4.1 Compute the similarity between a and b.

Cosine based similarity, also known as vector-based similarity, produces excellent outcomes in item-to-item filtering and the ratings are seen as vector in n-dimensional space. Similarity between the vectors can be calculated as the cosine of the angle between these vectors.

$$sim(\vec{a},\vec{b}) = \frac{\vec{a} \bullet \vec{b}}{|\vec{a}| * |\vec{b}|}$$

In item-based CF, item-to-item similarity is calculated along the columns whereas in user–based CF the similarity is calculated along the rows. It is a way of customizing figure of vector based similarity where few users might have rated highly in comparison to others. This drawback can be removed from vector-based similarity by subtracting average ratings for each user for the couple of items. The similarity measure for adjusted cosine can be calculated for two items a and b Here, U is a set of users who have rated both items a and b.

$$sim(\vec{a},\vec{b}) = \frac{\sum_{u \in U} (r_{u,a} - \bar{r}_u)(r_{u,b} - \bar{r}_u)}{\sqrt{\sum_{u \in U} (r_{u,a} - \bar{r}_u)^2} \sqrt{\sum_{u \in U} (r_{u,b} - \bar{r}_u)^2}}$$

The ratings can be predicted by calculating the weighted average. All the items which are similar to our target item are chosen first and from those similar items, pick up the items which the active user has rated. The users rating can be weighted for each of these items by computing the similarity between that and the target item.

$$pred(u,p) = \frac{\sum_{i \in ratedItem(u)} sim(i,p) * r_{u,i}}{\sum_{i \in ratedItem(u)} sim(i,p)}$$

1.2 *Top N Recommendations*

Top-N recommendation is to recommend a set of top-ranked items that will be of interest to a certain user. By designing a user-item rating matrix based on user browsed information and by computing the similarity between users or items recommendation is produced on top-N items. There are two types of top-N recommendations as discussed next.

1.2.1 User-based Top-N Recommendations:

User-based top-N recommendation algorithms identify the k most similar users (nearest neighbors) for an active user using the Pearson correlation or vector-space model. The similarity between the active user and other users are computed in N-dimensional item space where each user is treated as a vector. Once the most similar users have been discovered, the user-item matrix of corresponding rows are aggregated to discover a set of items purchased by the group collectively with their frequency. User-based CF techniques then recommend the top-most frequent items which the active user has not purchased. However, user-based top-N recommendation algorithms have limitations related to scalability and real-time performance.

1.2.2 Item-based Top-N Recommendations:

In this technique, the main task is to identify the k most similar item for the given set of items. Item-based top-N recommendation algorithms have been developed to deal with the scalability problem of user-based top-N recommendation algorithms. The algorithms firstly compute the most similar items for each item according to the similarities; then identify the set as candidates of recommended items by taking the union of the almost similar items and removing each of the items in the set that the user has already purchased. At last, calculate the similarities between each item of the set. The recommended item-based top-list is sorted in decreasing order of the similarity.

B. Content-Based Filtering:

Content-based filtering makes recommendations based on user's choices made in the past. A similarity can be established between objects as the basis for recommending items. It performs item recommendations by predicting the utility of items for a particular user based on how "similar" the items are to those that he/she liked in the past. E.g., in a movie recommendation application, a movie may have some features, such as, director, actors, genre, etc. Content-based recommendation approach analyzes a set of items which were previously rated by a user and makes a profile of user's choice depending on the characteristics of the objects rated by that user [12]. The user's interest or preference is represented by the same set of features often called as the user profile. Recommendations are made by comparing the user profile with candidate items expressed in the same set of features. The top-*k* best matched or most similar items are recommended to the user. The simplest approach to content-based recommendation is to compute the similarity of the user profile with each item.

For each item, item profile is needed to be created. A profile is a set of features for context specific (e.g. with films: actors, director, genre, title, etc.) and documents which is a set of important words. Important words are usually selected using the TF:IDF (Term Frequency times Inverse Doc Frequency) metric.

Most content-based recommender systems use simple retrieval models, such as keyword matching or the Vector Space Model (VSM) with basic TF-IDF weighting. VSM is a spatial representation of text documents represented as a vector in n-dimensional space. The overall vocabulary of a given collection of document corresponds to a term in each dimension. A vector of term weights is represented in every document where each weight represents the degree of union between the document and the term.

All of the content-based approaches represent documents by the important words in the documents. For example, Fab (Balabanovic 1997) represents a document in terms of the 100 words with the highest TF-IDF weights (Salton 1989) i.e., the words that appear frequently in this document, but infrequently in the whole document collection [22]. The TF-IDF is a statistic which reflects how important a word is to a document. The tf*idf weight, w(t,d), of a term t in a document d is a function of the frequency of t in the document (tft,d), the number of documents that contain the term (dft) and the number of documents in the collection (N).

$$w(t,d) = \frac{tf_{t,d}\log\left(\frac{N}{df_t}\right)}{\sqrt{\sum_i (tf_{t_i,d})^2 \log\left(\frac{N}{df_t}\right)^2}}$$

Let d = $\{d_1, d_2, \ldots, d_n\}$ denote a set of documents. Each document d_j is represented as a vector in a n-dimensional vector space, so $d_j = \{w_{1j}, w_{2j}, \ldots, d_{nj}\}$, where w_{kj} is the weight for term t_k in document d_j.

Also, a profile of the user's interests is used by most recommendation systems. It consists of various types of information:

a) A model of the user's preferences, i.e., a sketch of predicting likelihood for any item that the user is interested in.
b) A history of the user's interactions with the recommendation system. i.e., how the items have that a user has viewed together has been stored with other information about the user's interaction.

A similarity measure is required to determine the closeness between two documents. Many similarity measures have been derived to describe the proximity of two vectors; among those measures, cosine similarity is the most widely used:

$$sim(d_i, d_j) = \frac{\sum_k w_{ki} \cdot w_{kj}}{\sqrt{\sum_k w_{ki}^{2}} \cdot \sqrt{\sum_k w_{kj}^{2}}}$$

In content-based recommender systems relying on VSM, both user profiles and items are represented as weighted term vectors. Predictions of a user's interest in a particular item can be derived by computing the cosine similarity.

C. *Hybrid filtering:*

For better results hybrid recommender systems usually combine different techniques of collaborative approaches and content based approaches. Content-based recommender systems make recommendations by analyzing the content of textual information, such as documents, URLs, news messages, web logs, item descriptions, and profiles about users' tastes, preferences, and needs, and finding regularities in the content [21]. Collaborative filtering approaches build a model from a user's past behavior (items previously purchased or selected and/or numerical ratings given to those items) as well as similar decisions made by other users, then use that model to predict items (or ratings for items) that the user may have an interest in. One can avoid some limitations and problems of pure recommender systems, like the cold-start problem by using hybrid approaches. The combination of approaches can proceed in different ways [13]:

a) Partition implementation of algorithms and joining the results.
b) Make use of some rules of content-based filtering in collaborative approach.
c) Make use of some rules of collaborative filtering in content-based approach.
d) A combined recommender system can be created involving both approaches. Robin Burke categorized the taxonomy of hybrid recommender systems [14].

III. RELATED WORK

J. Bobadilla et. al. proposed a survey on recommender systems [2]. The paper gives us an overview of recommender systems as well as collaborative filtering methods and algorithms. Recommender systems have developed in parallel with the web. RS make use of different sources of information for providing users with predictions and recommendations of items. RS have proved to be a useful tool for addressing a portion of the information overload phenomenon from the Internet.

Another study on neighborhood-based algorithms for large-scale collaborative-filtering has been proposed by Yuke Fang et. al. [4]. In this paper, the collaborative filtering for personalized recommendation is studied based on K-Nearest Neighbor. By selecting the top k neighbors, the KNN modeling procedure becomes slow and limited. The paper proposed an improved algorithm which recommends items to users basing on global information.

Another directly related work on RS is found in [7]. Here, the authors proposed a system that suggests an item to a user depending upon the description of the item and a profile of the user's interests. It can be used in various sectors ranging from recommending news articles, webpage, television programmes, etc. Content-based recommendation systems share a general way for describing the items that may be recommended, a way for creating a profile of the user that shows the different types of items the user likes, and a way of comparing items to the user profile to decide what to recommend.

The paper on Amazon.com recommendation [23] gives a description on input about a customer's interests to generate a list of recommended items. Many applications use only the items that customers purchase and explicitly rate to represent their interests, but they can also use other attributes, including items viewed, demographic data, subject interests, and favorite artists. It provides an effective form of targeted marketing by creating a personalized shopping experience for each customer. There are three common approaches which are used in this paper: traditional collaborative filtering, cluster models, and search-based methods.

Another paper work [24] focused on how recommender systems help Ecommerce sites to increase their sales, and analyze different sites that use recommender systems. A categorization of recommender system is created where different techniques are used for creating recommendations. The products which are suggested to their customer can be recommended based on the top overall sellers on a site based on the demographics of the customer.

IV. Challenges and Issues

Collaborative filtering systems have been very successful in past, but their prevalent use has shown some potential challenges such as:

a. **Cold-start Problem** – There is a problem in recommending items to new users if his/her profile is almost empty and he/she has not rated any items yet because of which his/her taste is unknown to the system. This is called the cold-start problem. This problem can be solved by creating a user profile in some recommender system. Thus, if a user who is totally new to the system and have not rated before, the cold start problem arises. This problem occurs for both new user and new community. Both these can be resolved using hybrid approaches

b. **Scalability -** When there are many numbers of users and items, the system needs more resources for giving out information and forming recommendations. A large amount of resources is required for determining users with similar tastes and items with similar descriptions. It can be solved by mixing of different types of filters and physical improvement of systems.

c. **Sparsity -** There are users who have rated just a few items in online shops despite of having large amount of users and items. Using collaborative and other approaches, recommender system generally create neighbourhood of users using their profiles. Hence, it is very difficult to determine the user's taste if he/she has evaluated just only a few items. This could lead him/her to the wrong neighbourhood. Due to lack of information, sparsity problem arises [12].

d. **Privacy -** Privacy has been the most important problem in dealing with RS. The system should have the required amount of information about the user to get the correct and accurate recommendation. Sometimes, it includes data about the location of a particular user and also the demographic information. Hence, the confidentiality, reliability and security of the given information do possess a problem. By utilizing specialized algorithms, various online shops provide effective protection of privacy for the users.

V. Conclusion

Recently, RS implementation in the Internet has increased at a very fast rate, which has facilitated its use in diverse areas. This paper discusses about social RS to move a step ahead with the study of RS in social networks and gain insights and new ideas about the same. However, research in RS requires using a representative set of public databases to facilitate investigations on the algorithms, methods and techniques developed by researchers in the field.

References

[1] P. Resnick and H.R. Varian, "Recommender systems", Commun. ACM, vol. 40, pp. 56-58, March 1997.

[2] J. Bobadilla, F. Ortega, A. Hernando and A. Gutierrez, "Recommender systems survey", Knowledge-Based Systems vol. 46, pp. 109–132, 2013.

[3] Robin Burke, "Hybrid Recommender Systems: Survey and Experiments", California State University, Fullerton, Department of Information Systems and Decision Sciences, 2002.

[4] Yuke Fang, Yan Fu, Junlin Zhou, Siyu L, "Improved Neighborhood-based Algorithms for Large-scale Collaborative-filtering", Journal of Computational Information Systems vol. 7, pp. 2695-2702, 2011.

[5] G. Karypis, "Evaluation of item-based top-N recommendation algorithms", in Proceedings of the International Conference on Information and Knowledge Management (CIKM '01), pp. 247–254, USA, November 2001.

[6] Manolis Vozalis, Konstantinos G. Margaritis, "On the enhancement of collaborative filtering by demographic data", Web Intelligence and Agent Systems, vol. 4, no. 2, pp. 117–138, 2006.

[7] Michael J. Pazzani1and Daniel Billsus, "Content-Based Recommendation Systems", P. Brusilovsky, A. Kobsa, and W. Nejdl (Eds.): The Adaptive Web, LNCS 4321, pp. 325 – 341, © Springer-Verlag Berlin Heidelberg 2007.

[8] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms", in Proceedings of the 10th International conference on World Wide Web, ACM New York, USA, pp. 285–295, 2001.

[9] Z. Sun and N. Luo, "A new user-based collaborative filtering algorithm combining data-distribution", in Proceedings of the 2010 International Conference of Information Science and Management Engineering (ISME'10), Washington, DC, USA: IEEE Computer Society, pp. 19–23, 2010.

[10] M. Vozalis and K. Margaritis, "Using SVD and demographic data for the enhancement of generalized collaborative filtering," Information Sciences, vol. 177, no. 15, pp. 3017–3037, 2007.

[11] Y. Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model", in Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '08), pp. 426–434, Las Vegas, USA, August 2008.

[12] O'Connor, M., Cosley, D., Konstan, J.A., Riedl, J. PolyLens, "A Recommender System for Groups of Users", in Proceedings of the 7th Conference on European Conference on Computer Supported Cooperative Work, Kluwer Academic Publishers, pp. 199–218, 2001.

[13] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms," in Proceedings of the 10th

International conference on World Wide Web (WWW'01), New York, USA, ACM, pp. 285–295, 2001.

[14] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions", IEEE Trans. on Knowledge and Data Eng., vol. 17, pp. 734–749, June 2005.

[15] R. Burke, "The adaptive web", P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds. Berlin, Heidelberg: Springer-Verlag, ch. Hybrid web recommender systems, pp. 377–408, 2007

[16] J. Bobadilla, F. Serradilla, J. Bernal, "A new collaborative filtering metric that improves the behaviour of recommender systems", Knowledge-Based Systems, vol. 23, pp. 520–528, 2010.

[17] Ronak Patel, Priyank Thakkar, K Kotecha, "Enhancing movie recommender system", International Journal of Advanced Research in Engineering and Technology (IJARET), ISSN 0976 – 6499(Online) vol. 5, issue 1, pp. 73-82, 2014.

[18] Mohd Abdul Hameed, Omar Al Jadaan, S. Ramachandram, "Collaborative Filtering Based Recommendation System: A survey", International Journal on Computer Science and Engineering (IJCSE), ISSN: 0975-3397, vol. 4, no. 5, May 2012.

[19] Daniar Asanov, "Algorithms and Methods in Recommender Systems", Berlin Institute of Technology, Berlin, Germany, 2011

[20] Katarzyna Musiał, "Recommendation system for online social network", Master Thesis Software Engineering Thesis no: MSE-2006, July 2006.

[21] M. J. Pazzani, "A framework for collaborative, content-based and demographic filtering", Artificial Intelligence Review, vol. 13, pp. 393–408, 1999.

[22] M. Balabanovic, and Y. Shoham, "Fab: Content-Based Collaborative Recommendation", Communications of the ACM vol. 40, no. 3, March 97.

[23] Greg Linden, Brent Smith, and Jeremy York, "Amazon.com Recommendations Item-to-Item Collaborative Filtering", IEEE Computer Society, January - February 2003.

[24] J. Ben Schafer, Joseph Konstan, John Riedl, "Recommender Systems in E-Commerce", E-COMMERCE 99, Denver, Colorado ©1999 ACM 1-58113-176-3/99/0011.