Review of Content-based Recommendation System

Prajakta A. Holey¹, Dr. S.S. Prabhune²

¹M.E., C.E. Dept. of Comp. Sci. & Engg.
Shri Sant Gajanan Maharaj College of Engineering, Shegaon, India.

²Professor & Head, Dept. of Information Technology,
Shri Sant Gajanan Maharaj College of Engineering, Shegaon, India.

Abstract-- Content-based recommendation systems try to recommend items similar to those a given user has liked in the past. Recommender systems guide users in a personalized way to interesting objects in a large space of possible options. Indeed, the basic process performed by a content-based recommender consists in matching up the attributes of a user profile in which preferences and interests are stored, with the attributes of a content object (item), in order to recommend to the user new interesting items. It provides an overview of content-based recommender systems, with the aim of imposing a degree of order on the diversity of the different aspects involved in their design and implementation.

Keywords: Recommendation, aggregation, utility matrix.

I. INTRODUCTION

The abundance of information available on the Web and in Digital Libraries, in combination with their dynamic and heterogeneous nature, has determined a rapidly increasing difficulty in finding what we want when we need it and in a manner which best meets our requirements. As a consequence, the role of user modeling and personalized information access is becoming crucial: users need a personalized support in sifting through large amounts of available information, according to their interests and tastes. Many information sources embody recommender systems as a way of personalizing their content for users [3]. Recommender systems have the effect of guiding users in a personalized way to interesting or useful objects in a large space of possible options. Recommendation algorithms use input about a customer’s interest to generate a list of recommended items. At Amazon.com, recommendation algorithms are used to personalize the online store for each customer, for example showing programming titles to a software engineer and baby toys to a new mother [4]. The problem of recommending items has been studied extensively, and two main paradigms have emerged. Content-based recommendation systems try to recommend items similar to those a given user has liked in the past, whereas systems designed according to the collaborative recommendation paradigm identify users whose preferences are similar to those of the given user and recommend items they have liked [2].

Here, a comprehensive and systematic study of content-based recommender systems is carried out. The intention is twofold:
• to provide an overview of state-of-the-art systems, by highlighting the techniques which revealed the most effective, and the application domains in which they have adopted.
• to present trends and directions for future research which might lead towards the next generation of content-based recommender systems.

The application of content-based recommendation system is ‘Admission Recommendation System’ for important role in Education System which allot the best colleges to the student according to their merit score for pursuing higher education in reputed institutes e.g. (B.E./M.E) courses based on the cutoff marks at institute level in which they are eligible. Relevance Feedback model will be used with information retrieval system in education system.
II. MECHANISM

Basics of Content-based Recommender Systems

Systems implementing a content-based recommendation approach analyze a set of documents and/or descriptions of items previously rated by a user, and build a model or profile of user interests based on the features of the objects rated by that user. The profile is a structured representation of user interests. The recommendation process basically consists in matching up the attributes of the user profile against the attributes of a content object. The result is a relevance judgment that represents the user’s level of interest in that object. If a profile accurately reflects user preferences, it is of tremendous advantage for the effectiveness of an information access process. For instance, it could be used to filter search results by deciding whether a user is interested in a specific Web page or not, and, in the negative case, preventing it from being displayed.

A. A High Level Architecture of Content-based Systems

Content-based Information Filtering (IF) systems need proper techniques for representing the items and producing the user profile, and some strategies for comparing the user profile with the item representation. The high level architecture of a content-based recommender system is depicted in Figure (a). The recommendation process is performed in three steps, each of which is handled by a separate component:

- **Content Analyzer** – The main responsibility of the component is to represent the content of items (e.g., documents, Web pages, news, product descriptions, etc.) coming from information sources. This representation is the input to the Profile Learner and Filtering Component.
- **Profile Learner** – This module collects data representative of the user preferences and tries to generalize this data, in order to construct the user profile. Usually, the generalization strategy is realized through machine learning techniques i.e. user interests starting from items liked or disliked in the past.
- **Filtering Component** – This module exploits the user profile to suggest relevant items by matching the profile representation against that of items to be recommended.

Typically, it is possible to distinguish between two kinds of relevance feedback: positive information (inferring features liked by the user) and negative information (i.e., inferring features the user is not interested in). Two different techniques can be adopted for recording user’s feedback.

![Fig. (a): Architecture of a Content-based Recommender](image)

When a system requires the user to explicitly evaluate items, this technique is usually referred to as “explicit feedback”; the other technique, called “implicit feedback”, does not require any active user involvement, in the sense that feedback is derived from monitoring and analyzing user’s activities. Explicit evaluations indicate how relevant or interesting an item is to the user. There are three main approaches to get explicit relevance feedback.

- **like/dislike** – items are classified as “relevant” or “not relevant” by adopting a simple binary rating scale.
- **ratings** – a discrete numeric scale is usually adopted to judge items. Alternatively, symbolic ratings are mapped to a numeric scale, such as in Syskill & Webert [6], where users have the possibility of rating a Web page as hot, lukewarm, or cold;
- **text comments** – Comments about a single item are collected and presented to the users as a means of facilitating the decision-making process.
The literature proposes advanced techniques from the affective computing research area to make content-based recommenders able to automatically perform this kind of analysis. Explicit feedback has the advantage of simplicity. Implicit feedback methods are based on assigning a relevance score to specific user actions on an item, such as saving, discarding, printing, bookmarking, etc.

B. Advantages and Drawbacks of Content-based Filtering

The adoption of the content-based recommendation paradigm has several advantages when compared to the collaborative one:

• **User Independence** - exploit solely ratings provided by the active user to build her own profile.

• **Transparency** - Explanations on how the recommender system works can be provided by explicitly listing content features or descriptions that caused an item to occur in the list of recommendations.

• **New item** - Content-based recommenders are capable of recommending items not yet rated by any user.

• **Limited Content Analysis** - Content-based techniques have a natural limitation in the number and type of features that are associated.

• **Over-Specialization** - Content-based recommenders have no inherent method for finding something unexpected.

• **New User** - Enough ratings have to be collected before a content-based recommendersystem can really understand user preferences and provide accurate recommendations.

### III. METHODS

**State of the Art of Content-based Recommender Systems**

As the name implies, content-based filtering exploits the content of data items to predict its relevance based on the user’s profile. Research on content-based recommendersystems takes place at the intersection of many computer science topics, especially Information Retrieval [5] and Artificial Intelligence.

From Information Retrieval (IR), research on recommendation technologies derives the vision that users searching for recommendations are engaged in an information seeking process. In IR systems the user expresses a one-off information need by giving a query (usually a list of keywords), while in IF systems the information need of the user is represented by her own profile. Items to be recommended can be very different depending on the number and types of attributes used to describethem. Each item can be described through the same small number of attributes withknown set of values, but this is not appropriate for items, such as Web pages, news, emails or documents, described through unstructured text. From an Artificial Intelligence perspective, the recommendation task can be cast as a learning problem that exploits past knowledge about users. At their simplest, user profiles are in the form of user-specified keywords or rules, and reflect the longterm interests of the user.

**A. Item Representation**

Items that can be recommended to the user are represented by a set of features, also called attributes or properties. For example, in a movie recommendation application, features adopted to describe a movie are: actors, directors, genres, subject matter, . . . [7].

In most content-based filtering systems, item descriptions are textual features extracted from Web pages, emails, news articles or product descriptions. Unlike structured data, there are no attributes with well-defined values. String matching suffers from problems of:

• **POLYSEMY**, the presence of multiple meanings for one word.

• **SYNONYMY**, multiple words with the same meaning.

The result is that, due to synonymy, relevant information can be missed if the profile does not contain the exact keywords in the documents while, due to polysemy, wrong documents could be deemed relevant. Items are represented as follows:

**a. Keyword-based Vector Space Model**

VSM is a spatial representation of text documents. In that model, each document is represented by a vector in an-dimensional space, where each dimension corresponds to a
term from the overall vocabulary of a given document collection.

b. Review of Keyword-based Systems

Several keyword-based recommender systems have been developed in a relatively short time, and it is possible to find them in various fields of applications, such as news, music, e-commerce, movies, etc. Each domain presents different problems, that require different solutions.

c. Semantic Analysis by using Ontologies

Semantic analysis allows learning more accurate profiles that contain references to concepts defined in external knowledge bases. The main motivation for this approach is the challenge of providing a recommender system with the cultural and linguistic background knowledge which characterizes the ability of interpreting natural language documents and reasoning on their content.

d. Semantic Analysis by using Encyclopedic Knowledge Sources

Common-sense and domain-specific knowledge may be useful to improve the effectiveness of natural language processing techniques by generating more informative features than the mere bag of words. The process of learning user profiles could benefit from the infusion of exogenous knowledge (externally supplied), with respect to the classical use of endogenous knowledge (extracted from the documents themselves).

B. Methods for Learning User Profiles

The problem of learning user profiles can be cast as a binary text categorization task: each document has to be classified as interesting or not with respect to the user preferences. Therefore, the set of categories is $C = \{c^+, c^−\}$, where $c^+$ is the positive class (user-likes) and $c^−$ the negative one (user-dislikes).

a. Probabilistic Methods and Naïve Bayes

Naïve Bayes is a probabilistic approach to inductive learning, and belongs to the general class of Bayesian classifiers. These approaches generate a probabilistic model based on previously observed data.

b. Relevance Feedback and Rocchio’s Algorithm

Relevance feedback is a technique adopted in Information Retrieval that helps users to incrementally refine queries based on previous search results. It consists of the users feeding back into the system decisions on the relevance of retrieved documents with respect to their information needs. Relevance feedback and its adaptation to text categorization, the well-known Rocchio’s formula, are commonly adopted by content-based recommender systems. The general principle is to allow users to rate documents suggested by the recommender system with respect to their information need. The Rocchio’s method is used for inducing linear profile-style classifiers. This algorithm represents documents as vectors, so that documents with similar content have similar vectors.

C. Other Methods

Other learning algorithms have been used in content-based recommendation systems. Decision trees, Decision rule classifiers, nearest neighbor algorithms are some other methods used in content-based recommendation systems.

Decision trees are trees in which internal nodes are labeled by terms, branches departing from them are labeled by tests on the weight that the term has in the test document, and leaves are labeled by categories. Decision trees are used in the Syskill & Webert recommender system.

Decision rule classifiers are similar to decision trees, because they operate in a similar way to the recursive data partitioning approach. An advantage of rule learners is that they tend to generate more compact classifiers than decision trees learners.

Nearest neighbor algorithms, also called lazy learners, simply store training data in memory, and classify a new unseen item by comparing it to all stored items by using a similarity function. Daily Learner and Quickstep use the nearest neighbor algorithm.

IV. TRENDS AND FUTURE RESEARCH

A. The Role of User Generated Content in the Recommendation Process
Web 2.0 is a term describing the trend in the use of World Wide Web technology that aims at promoting information sharing and collaboration among users. According to Tim O'Reilly, the term “Web 2.0” means putting the user in the center, designing software that critically depends on its users since the content, as in Flickr, Wikipedia, Del.icio.us, or YouTube, is contributed by thousands or millions of users. That is why Web 2.0 is also called the “participative Web”. O'Reilly also defined Web 2.0 as “the design of systems that get better the more people use them”.

One of the forms of User Generated Content (UGC) that has drawn more attention from the research community is folksonomy. The freely chosen keywords according to interest are called tags.

Folksonomies provide new opportunities and challenges in the field of recommendersystems.

a. Social Tagging Recommender Systems

Several methods have been proposed for taking into account user tagging activity within content-based recommender systems.

The user profile is represented in the form of a tag vector, with each element indicating the number of times a tag has been assigned to a document by that user. The matching of profiles to information sources is achieved by using simple string matching.

B. Beyond Over-specialization: Serendipity

Content-based systems suffer from over-specialization, since they recommend only items similar to those already rated by users. One possible solution to address this problem is the introduction of some randomness. For example, the use of genetic algorithms has been proposed in the context of information filtering. In certain cases, items should not be recommended if they are too similar to something the user has already seen, such as a different news article describing the same event. In summary, the diversity of recommendations is often a desirable feature in recommender system.

It is useful to make a clear distinction between novelty and serendipity. Novelty occurs when the system suggests to the user an unknown item that she might have autonomously discovered. A serendipitous recommendation helps the user to find a surprisingly interesting item that she might not have otherwise discovered (or it would have been really hard to discover). The example of the difference between novelty and serendipity, consider a recommendation system that simply recommends movies that were directed by the user’s favorite director. If the system recommends a movie that the user was not aware of, the movie will be novel, but probably not serendipitous. On the other hand, a recommender that suggests a movie by a new director is more likely to provide serendipitous recommendations. Recommendations that are serendipitous are by definition also novel.

V. CONCLUSIONS

In this content-based recommender systems, is overviewed and provided by the most important characterizing systems. Although there is a bunch of recommender systems in different domains, they share in common a means for representing items to be recommended and user profiles. Here the main issues related to the representation of items, starting from simple techniques for representing structured data, to more complex techniques coming from the Information Retrieval research area for unstructured data are discussed. The main content recommender systems developed in the last 15 years, by highlighting the reasons for which a more complex “semantic analysis” of content is needed in order to go beyond the syntactic evidence of user interests provided by keywords are analyzed. A review of the main strategies (and systems) adopted to introduce some semantics in the recommendation process is carried out, by providing evidence of the leading role of linguistic knowledge, even if a more specific knowledge is mandatory for a deeper understanding and contextualization of the user interests in different application domains.
The issues presented here will contribute to stimulate their search community about the next generation of content-based recommendation technologies.

REFERENCES