A
Project Report
On
Approaches to handle Cold Start Problem In Recommendation System

(IT345-Software Group Project)

Prepared by
Kahani Patel (16IT076)

Under the Supervision of
Prof. Purvi Prajapati

Submitted to

Charotar University of Science & Technology (CHARUSAT)
For the Partial Fulfillment of the Requirements for the
Degree of Bachelor of Technology (B.Tech.)
In Information Technology (IT)
For 5th semester B.Tech

Submitted at

CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY
Accredited with Grade A by NAAC
Accredited with Grade A by KCG

cspit
Chandubhai S Patel Institute of Technology

DEPARTMENT OF INFORMATION TECHNOLOGY
(NBA Accredited)
Chandubhai S. Patel Institute of Technology (CSPIT)
Faculty of Technology & Engineering (FTE), CHARUSAT
At: Changa, Dist: Anand, Pin: 388421.
June, 2018

DECLARATION BY THE CANDIDATES

I hereby declare that the project report entitled "**Approaches to handle Cold Start Problem In Recommendation System** " submitted by me to Chandubhai S. Patel Institute of Technology, Changa in partial fulfilment of the requirement for the award of the degree of **B.Tech** in Information Technology, from Department of Information Technology, CSPIT/FTE, is a record of bonafide **IT345 Software Group Project** carried out by me under the guidance of **Prof. Purvi Prajapati**. I further declare that the work carried out and documented in this project report has not been submitted anywhere else either in part or in full and it is the original work, for the award of any other degree or diploma in this institute or any other institute or university.

16IT076

Prof. Purvi Prajapati
Assistant Professor

CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

Accredited with Grade A by NAAC
Accredited with Grade A by KCG

CERTIFICATE

This is to certify that the report entitled "**Approaches to handle Cold Start Problem in Recommendation System**" is a bonafied work carried out by **Kahani Patel (16IT076)** under the guidance and supervision of **Prof. Purvi Prajapati** for the subject **Software Group Project (IT345)** of 5th Semester of Bachelor of Technology in **Information Technology** at Chandubhai S. Patel Institute of Technology (CSPIT), Faculty of Technology & Engineering (FTE) – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred by the examiner(s).

Under the supervision of,

Prof. Purvi Prajapati
Assistant Professor
CSPIT- CHARUSAT, Changa

Dr. Parth Shah
Head -Department of Information Technology,
CHARUSAT, Changa, Gujarat.

Chandubhai S. Patel Institute of Technology (CSPIT)

Faculty of Technology & Engineering (FTE), CHARUSAT

At: Changa, Ta. Petlad, Dist. Anand, Pin: 388421. Gujarat.

Acknowledgement

I wish to warmly thank my guide, **Prof. Purvi Prajapati** for all her diligence, guidance, encouragement and help throughout the period of research, which have enabled me to complete the research work in time. Her constant inspiration and encouragement along with her valuable guidance has been instrumental in the successful completion of this research. She has always been willingly present whenever I needed the slightest support from her. I also thank her for the time that she spared for me, from her extreme busy schedule. I would also like to show my gratitude to my guide for sharing her pearls of wisdom with me during the course of this research. I would like to thank my guide, for her valuable and constructive suggestions during the planning and development of this research work.

- Kahani Patel (16IT076)

Abstract

Recommender system applies many knowledge discovery techniques to the problem of making personalized recommendation. The most popular technique is collaborative filtering (CF) that was expected to produce high quality recommendation as soon as possible. User based and item based CF is most popular approach and achieving widespread success on the web. User based CF produce high quality recommendation, but calculation complexity of prediction increase with the number of participant in the system. In the other hand, item based produce low quality recommendation but in the faster time. Commonly, CF have no feature about new item in the system. To address these challenges, we introduce a new approach of CF method, user-item based CF. User-item based CF use missing value algorithm to calculate the rating prediction and is not like other approach, user-item based eliminate the neighborhood formation. This report contains an idea of better solution of using naïve bayes classifier to make item based collaborative filtering more efficient.

List of Figures

TABLE OF CONTENTS

Chapter 4: Naïve bayes classifier

Chapter 5: Conclusion

Approaches to handle Cold Start Problem In Recommendation System

Chapter 1: Introduction to Recommendation System

1.1 Introduction

In the era of internet, there are numbers of choices and transactions available for users. Hence the system must there, which filtered and prioritize the information or choice or transaction user may like to get better performance and convenient result for user. This system is known as Recommendation System.

These systems are either suggests user for the same websites in order to engage user to their website only or suggests user for other websites for the finance. [14] Besides that, these systems can rely on different algorithms for what to suggest and how to suggest. Usually these systems can take the input for the suggestions from the user feedback or like or dislike or wish lists. Also these systems can suggests accordingly by finding the similarities between two users and if one user like something then there is a chance that the other one might like that too.

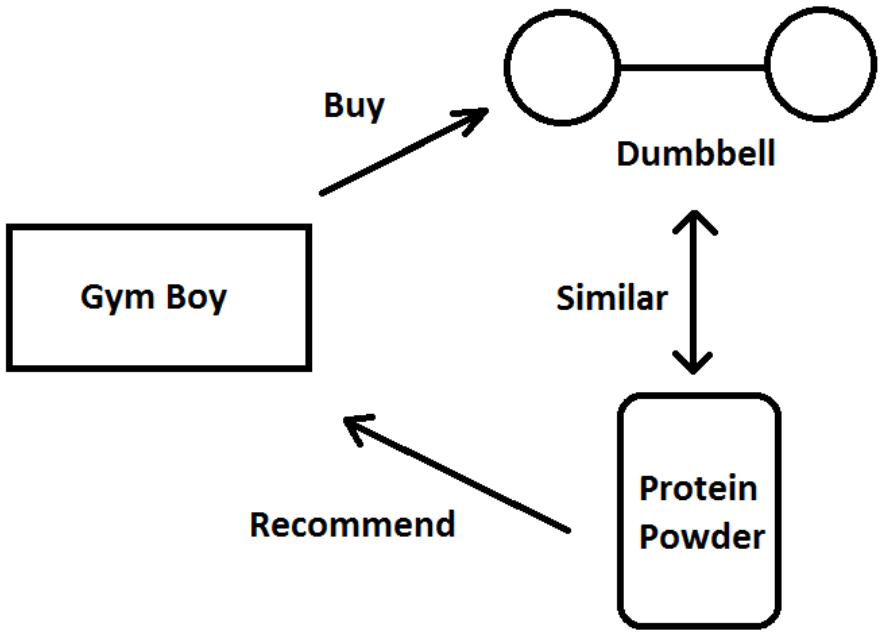


Fig. 1.1 – Recommendation System [15]

Figure 1.1 says that a gym boy bought dumbbell online. Dumbbell and protein power are similar type or products from gym. So Recommendation System Recommend Protein

Power to that gym boy. As per the survey, 75% of people choose from the Recommendation by Ads or any other resources. Recommendation System has big value in MNC's like Google, Amazon and Netflix And so on. The current web pages are not the way to judge any websites just because it shows relevant to the personal interest. The day was far too apart when automated combining process combines user's interest with machine. [19] If any system would come in future then system would give user more benefits who might get frustrated with scrolling.

1.2 Importance of Recommendation System

In the 21st century, we have many choices to select, wear, and read and so on. To fulfill that choices or to choose among them, these systems are designed into practically any device and platform. We are used to constantly receiving recommendations, but the way in which they have settled in our daily life is a long and interesting history marked by ups and downs. [8] The main goal of recommendation system is to increase the production of sales of merchant. By recommending that means selecting items for user, recommendation system gives relative items. In this information era, people want something more and accurate information of everything. [10] So this is the most important to introduce recommendation system. Recommendation system gives relevant information to user within short time.

1.3Applications of Recommendation System

Application developments of recommender systems, clusters their applications into eight main categories: e-government, e-business, e-commerce/e-shopping, e-library, e-learning, e-tourism, e-resource services and e-group activities, and summarizes the related recommendation techniques used in each category. [20]

- Entertainment – Recommend for Movies, Music and so on.
- Content – Recommend for documents, articles, web pages, applications and so on.
- E-Commerce – Recommend for Books, Camera, and PC and so on.
- Services – Recommend for consultancy, travel services, houses for rent or maternity services. [12]
- Querying and computational advertisement.

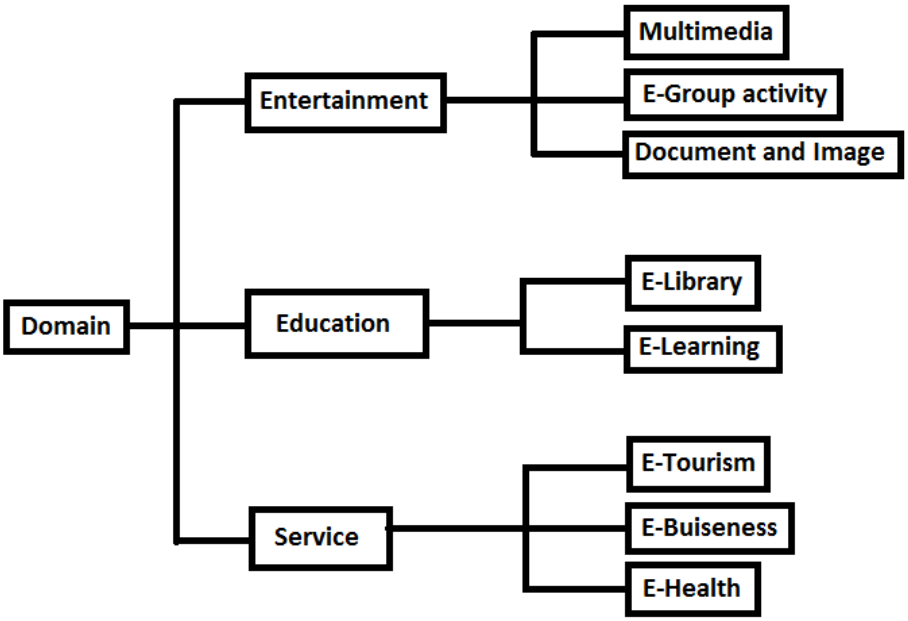


Fig. 1.2 – Applications of Recommendation System

Figure 1.2 states that there are many fields in which Recommendation System is must needed. Entertainment, Education and Services are major part where these are roots of all other applications.

Chapter 2: Recommendation Approaches

Starting with the basic algorithmic approaches, we exemplify the functioning of the algorithms and discuss criteria that help to decide which algorithm should be applied in which context. [14, 17] In this chapter, we introduce the basic approaches of collaborative filtering, content-based filtering, and knowledge-based recommendation.

2.1 List of Techniques

i. Collaborative Filtering System
ii. Content-based System
iii. Hybrid System

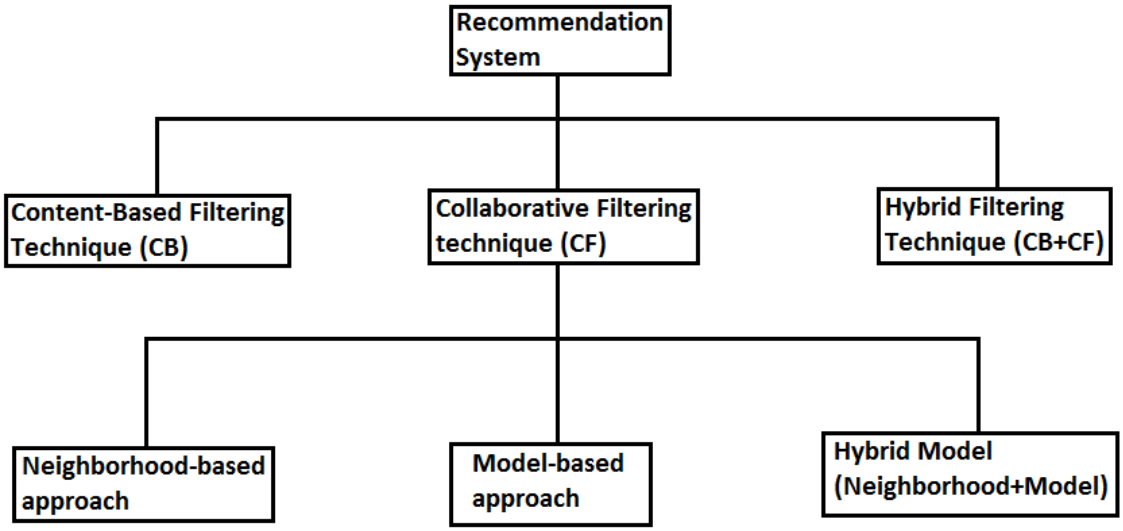


Fig. 2.1 – Approaches of Recommendation System

Figure 2.1 shows us the different approaches are listed in the form of hierarchy of Recommendation System are drawn in the above figure. As one can see Collaborative, Content-based and Hybrid filtering are main three approaches for Recommendation System implementation.

2.2 Brief Description of each Technique

The three basic approaches of collaborative filtering, content-based filtering, and knowledge-based recommendation exploit different sources of recommendation knowledge and have different strengths and weaknesses.

2.2.1 Collaborative filtering

In the traditional manner, user ask their family members or friends to choose on their behalf for user's personal decision making. These people plays a major role for recommend items or select among more for user. In collaborative filtering, user fills feedback form or gives rating for their shopping item and it will become the information for the system to recommend other users. [9, 10] Collaborative filtering replace the family members and friends by nearest neighbors who are other users with same preference or behavior like the current user.

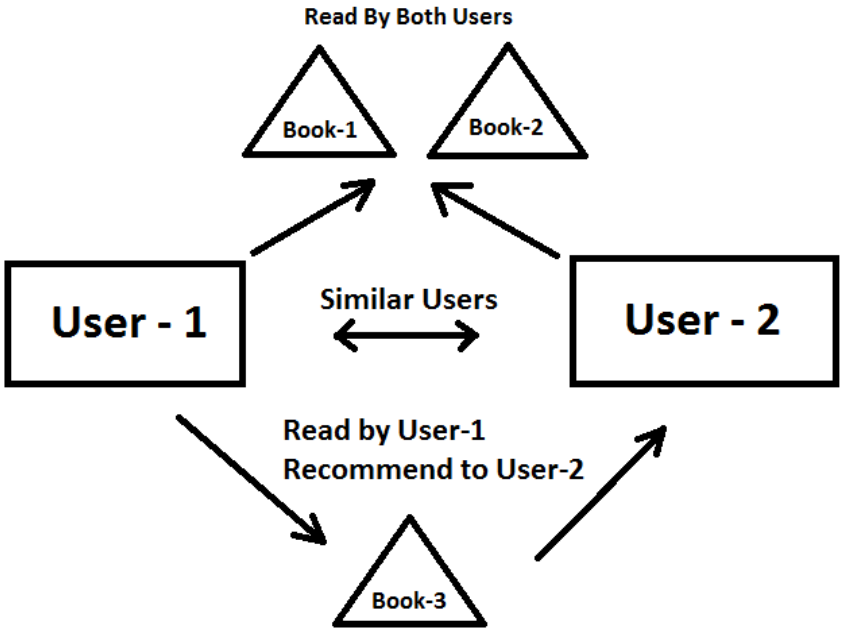


Fig. 2.2 – Collaborative Filtering

Figure 2.2 consists of two users and three books. There is a possibility that User-1 have read book-1 and User-2 have read book-2 and if both users have some similarities and user-1 have read book-3 then it will recommend to User-2.

2.2.2 Content-based System

Based on item description and user preference, Content-based algorithm recommend the item to user. [22] There are several advantages of content-based system like relevant to user's interest, quick to search, transparency, easy to implement etc. User gives the input explicitly by rating or implicitly by click on the links, the data will go through API and will

generate your profile then which is used to suggest other users as a recommendation. [10] The more you input or make your actions, more your profile in API become more accurate.

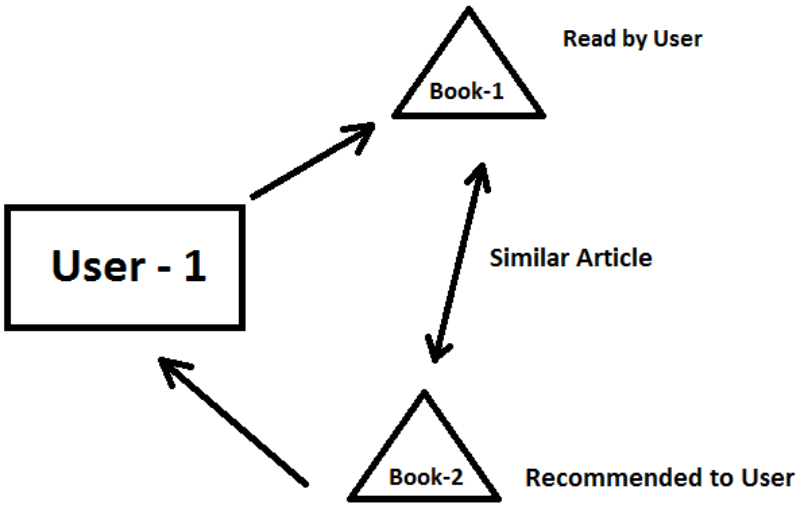


Fig. 2.3 – Content-based filtering

Figure 2.3 shows this scenario is different from the above one and totally independent from collaborative filtering for recommendation. If two books, book-1 and book-2 are of similar type and User-1 have read book-1 then he/she will be recommend the book-2 by recommendation system.

2.2.3 Hybrid System (Content-based + Collaborative based)

Hybrid system is combination of collaborative filtering system and content-based system. Collaborative filtering system recommends user with matrix recommendation and Content-based system recommends user by information and user's activities. [11] If one of the users cannot provide rating then in that case the Collaborative filtering system does not work so like this type of problems, we have to go with the Content-based system.

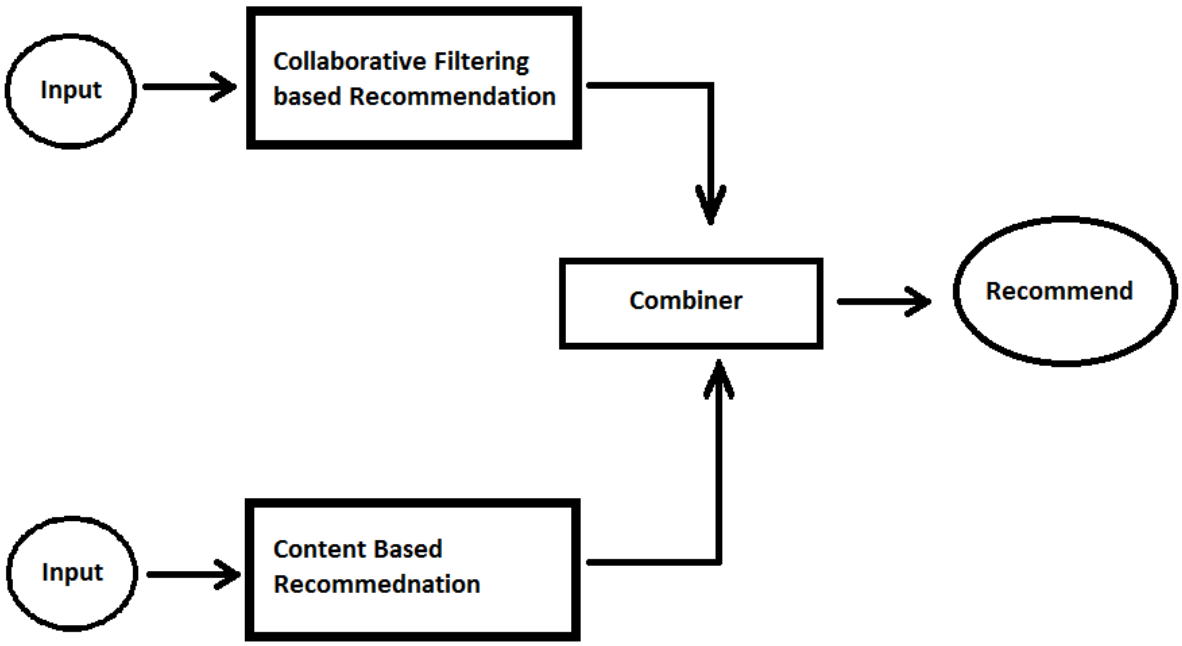


Fig. 2.4 – Hybrid System of Filtering

Figure 2.4 describe the Hybrid system is consist of both collaborative filtering as well as content-based filtering. When there is lack of information to recommend by collaborative then system will go with content-based filtering.

Chapter 3: Cold Start Problem

Cold start is a potential problem in computer-based information systems which involve a degree of automated data modelling. Specifically, it concerns the issue that the system cannot draw any inferences for users or items about which it has not yet gathered sufficient information.

3.1 Introduction to cold start problem

So what is the cold start problem? The term derives from cars. When it's really cold, the engine has problems with starting up, but once it reaches its optimal operating temperature, it will run smoothly. With recommendation engines, the "cold start" simply means that the circumstances are not yet optimal for the engine to provide the best possible results. In eCommerce, there are two distinct categories of cold start: product cold start and user cold starts. Here we're going to examine both.

Recommendation engines that run on collaborative filteringrecommend each item (products advertised on your site) based on user actions. The more user actions an item has, the easier it is to tell which user would be interested in it and what other items are similar to it. As time progresses, the system will be able to give more and more accurate recommendations.
This, however, brings a major contradiction and difficulty to classified sites and their recommendation engines. Even though the newest ads are actually the most relevant ones, a recommendation system has far less confidence in recommending them to your users than it has with older items, but it's just simply not a good idea to let older ads dominate the recommendation process.

3.2 Types of cold start problem

There are three cases of cold start :

1. **New context**: refers to the start-up of the recommender, when, although a catalogue of items might exist, almost no users are present and the lack of user interaction makes very hard to provide reliable recommendations
2. **New item**: a new item is added to the system, it might have some content information but no interactions are present

3. **New user**: a new user registers and has not provided any interaction yet, therefore it is not possible to provide personalized recommendations

3.3 New user problem

The new user case refers to when a new user enrolls in the system and for a certain period of time the recommender has to provide recommendation without relying on the user's past interactions, since none has occurred yet. This problem is of particular importance when the recommender is part of the service offered to users, since a user who is faced with recommendations of poor quality might soon decide to stop using the system before providing enough interaction to allow the recommender to understand his/her interests. The main strategy in dealing with new users is to ask them to provide some preferences to build an initial user profile. A threshold has to be found between the length of the user registration process, which if too long might indice too many users to abandon it, and the amount of initial data required for the recommender to work properly.

3.4 New Item problem

The item cold-start problem refers to when items added to the catalogue have either none or very little interactions. This constitutes a problem mainly for collaborative filtering algorithms due to the fact that they rely on the item's interactions to make recommendations. If no interactions are available then a pure collaborative algorithm cannot recommend the item. In case only a few interactions are available, although a collaborative algorithm will be able to recommend it, the quality of those recommendations will be poor.This arises another issue, which is not anymore related to new items, but rather to *unpopular items*. In some cases (e.g. movie recommendations) it might happen that a handful of items receive an extremely high number of iteractions, while most of the items only receive a fraction of them. This is referred to as popularity bias.

3.5 New context problem

The new community problem, or systemic bootstrapping, refers to the startup of the system, when virtually no information the recommender can rely upon is present.This case presents the disadvantages of both the New user and the New item case, as all items and users are new. Due to this some of the techniques developed to deal with those two cases are not applicable to the system bootstrapping.

3.6 Basic solutions

1. Asking the user at the beginning to rate some items.

2. Asking the users explicitly to state their taste in aggregate and suggesting items to the user based on the collected demographic information.

3. User demographic information can be used to know about the location, zip-code along with interactions of the new user with the system in order to recommend items on the basis of ratings provided by other similar users having similar demographic information.

Chapter 4: Naïve bayes classifier

In machine learning, naive Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naive) independence assumptions between the features.

Naive Bayes has been studied extensively since the 1950s. It was introduced under a different name into the text retrieval community in the early 1960s,and remains a popular (baseline) method for text categorization, the problem of judging documents as belonging to one category or the other (such as spam or legitimate, sports or politics, etc.) with word frequencies as the features. With appropriate pre-processing, it is competitive in this domain with more advanced methods including support vector machines. It also finds application in automatic medical diagnosis.

Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression, which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers.

4.1 Introduction

Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. There is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlations between the color, roundness, and diameter features.

For some types of probability models, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without accepting Bayesian probability or using any Bayesian methods.

Despite their naive design and apparently oversimplified assumptions, naive Bayes classifiers have worked quite well in many complex real-world situations. In 2004, an analysis of the Bayesian classification problem showed that there are sound theoretical reasons for the apparently implausible efficacy of naive Bayes classifiers.Still, a comprehensive comparison with other classification algorithms in 2006 showed that Bayes classification is outperformed by other approaches, such as boosted trees or random forests.

An advantage of naive Bayes is that it only requires a small number of training data to estimate the parameters necessary for classification.

4.2 How it works?

A class's prior may be calculated by assuming equiprobable classes (i.e., priors = 1 / (number of classes)), or by calculating an estimate for the class probability from the training set (i.e., (prior for a given class) = (number of samples in the class) / (total number of samples)). To estimate the parameters for a feature's distribution, one must assume a distribution or generate nonparametric models for the features from the training set.[8]

The assumptions on distributions of features are called the *event model* of the Naive Bayes classifier. For discrete features like the ones encountered in document classification (include spam filtering), multinomial and Bernoulli distributions are popular. These assumptions lead to two distinct models, which are often confused.

Following is the formula to find posterior in naïve bayes theorem:

$$p(C_k \mid \mathbf{x}) = \frac{p(C_k)\, p(\mathbf{x} \mid C_k)}{p(\mathbf{x})}$$

Here Ck is the class which we want our sample document to be classified in.

X is the other attributes in which all the total documents fall.

Example:

For given dataset:

Person	height (feet)	weight (lbs)	foot size(inches)
male	6	180	12
male	5.92 (5'11")	190	11
male	5.58 (5'7")	170	12
male	5.92 (5'11")	165	10
female	5	100	6
female	5.5 (5'6")	150	8
female	5.42 (5'5")	130	7
female	5.75 (5'9")	150	9

Given sample value is:

Person	height (feet)	weight (lbs)	foot size(inches)
sample	6	130	8

By applying the formula i.e:

$$\text{posterior (male)} = \frac{P(\text{male})\, p(\text{height} \mid \text{male})\, p(\text{weight} \mid \text{male})\, p(\text{foot size} \mid \text{male})}{evidence}$$

$$\text{posterior (female)} = \frac{P(\text{female})\, p(\text{height} \mid \text{female})\, p(\text{weight} \mid \text{female})\, p(\text{foot size} \mid \text{female})}{evidence}$$

- Posterior(male)=6.193*10^-9
- Posterior(female)=5.3778*10^-5
- As here posterior(female)>posterior(male) so, the sample is predicted to be female.

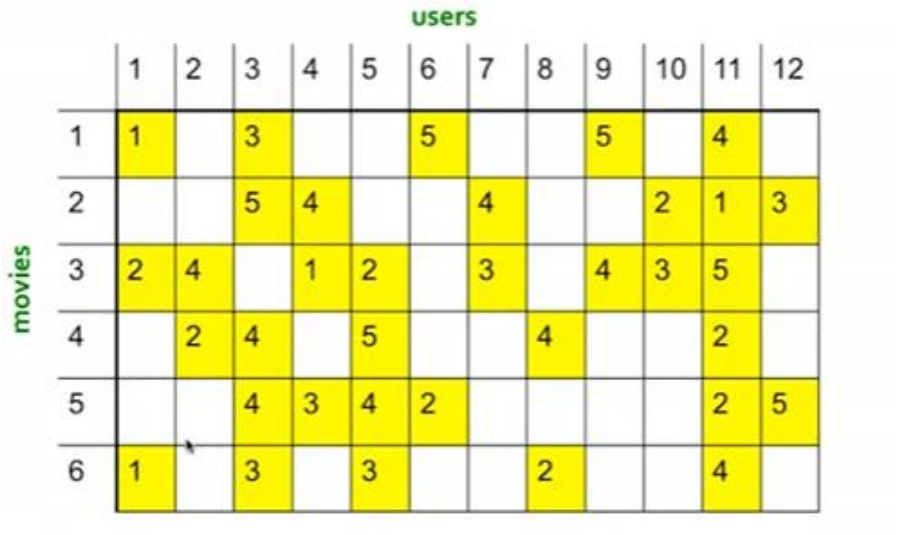
4.3 Use of naïve bayes to resolve cold start problem

Basically, naïve bayes classifier is used to classify data precisely i.e. if there is any missing attribute value in our dataset and we want to find out the class value we can use naïve bayes to figure it out. So we can use naïve bayes to figure out the missing values in our dataset for an accurate recommendation. We can use the output we get after using naïve bayes as an input for item based collaborative filtering. By using naïve bayes if there is new item problem in recommendation then we can get wanted information for the new item.

4.4 Dataset description for naive bayes classifier

Attribute	Attribute value
Outlook	Rainy, overcast, sunny
Temperature	Hot, mild, cool
Humidity	Normal, high
Windy	True, false
Play golf	Yes, no

4.5 Dataset description for item based collaborative filtering



movies \ users	1	2	3	4	5	6	7	8	9	10	11	12
1	1		3			5			5		4	
2			5	4			4			2	1	3
3	2	4		1	2		3		4	3	5	
4		2	4		5			4			2	
5			4	3	4	2					2	5
6	1		3		3			2			4	

Here, by using item based filtering we can predict the missing ratings.

Chapter 5: Conclusion

5.1 Conclusion

In order to handle the large volume of dataset, recommendation systems are used to deliver exactly what any user wants. But there are several problems in the system which makes the recommendation less efficient and one such problem is cold start problem. We can solve it using collaborative filtering techniques but again if there is any missing attribute or class in the dataset then filtering would not be efficient. So we can use naïve bayes classifier to fill the empty gaps in the dataset and use the same for item based collaborative filtering to get accurate recommendations.

5.2 Future work

In the future, implementation of algorithm by using particular dataset as needed in naïve bayes classifier and further implementation of the same for item based collaborative filtering. After implementing code an application of movie suggestion can be made.

References

1. Recommendation system:https://www.sciencedirect.com/science/article/pii/S1110866515000341
2. User-based CF and item-based CF:https://www.youtube.com/watch?v=6mGMBipt7kU
3. Naïvebayes-classifier algorithm:https://eprints.soton.ac.uk/268483/1/IMECS2010_MustansarAliGhazanfar.pdf
4. Combining item based CF and naïve bayes classifier , algorithm:https://eprints.soton.ac.uk/268483/1/IMECS2010_MustansarAliGhazanfar.pdf
5. Dataset referances: https://toolbox.google.com/datasetsearch
6. Python code for item based collaborative filtering: https://github.com/revantkumar/Collaborative-Filtering
7. Research paper for collaborative filtering:
   1. https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0184516
   2.https://www.researchgate.net/publication/280576923_Research-paper_recommender_systems_A_literature_survey