Research Report On Filtering Techniques

Abstract

On the Internet, where the number of choices is overwhelming, there is need to filter, prioritize and efficiently deliver relevant information in order to alleviate the problem of information overload, which has createda potential problem to many Internet users. **Recommender systems** solve this problem by searching through large volume of dynamically generated information to provide users with personalized content and services.

Keywords: Filtering, Recommender systems, Filtering techniques, Content-based filtering, Collaborative filtering, Hybrid filtering, Evaluation metrics for Recommender systems, Conclusion.

Introduction

Data filtering in Data science can refer to a wide range of strategies or solutions for refining data sets. This means the data sets are refined into simply what a user (or set of users) needs, without including otherdata that can be repetitive, irrelevant or even sensitive. Different types of data filters can be used to amend reports, query results, or other kinds of information results.

Typically, data filtering will involve taking out information that isuseless to a reader or information that can be confusing. Generated reports and query results from database tools often result in large andcomplex data sets. Redundant or impartial pieces of data can confuse or disorient a user. Filtering data can also make results more efficient.

In some other cases, data filters work to prevent wider access to sensitive information. For example, a data filtering program could scrub Social Security numbers, credit card numbers and other identifiers from complex client data sets coming into an employee’s workstation or, even more importantly, onto his or her mobile device. With the "bring your own device" (BYOD) movement emerging withinthe business world, data filtering can solve some security problems related to the information that employees need to do their jobs.

What is Filtering in Data Science?

- Data Filtering is **one of the most frequent data manipulation operation**. It is similar to filter in Python for selecting specific rowsbased on some conditions.

- Data filtering is **the task of reducing the content of noise or errors from measured process data**. It is an important task because measurement noise masks the important features in the data and limits their usefulness in practice.

Principle of Filtering :-

- Data filtering is **the process of choosing a smaller part of your data set and using that subset for viewing or analysis**.Filtering is generally (but not always) temporary – the complete data set is kept, but only part of it is used for the calculation.

Purpose of Filtering :-

It is **done to make it easier to focus on specific information ina large dataset or table of data**. Filtering does not remove or modify data.

A Quick Primer on Recommender Systems

- *Recommendation Engines are a subclass of **information filtering system** that seek to predict the 'rating' or 'preference' that user would give to an item.*
- A recommender system is a subclass of information filtering thatseeks to predict the "rating" or "preference" a user will give an item, such as a product, movie, song, etc.

- Recommender systems provide personalized information by learning the user's interests through traces of interaction with that user. Much like machine learning algorithms, a recommender system makes a prediction based on a user's past behaviours. Specifically, it's designed to predict user preference for a set of itemsbased on experience.
- Mathematically, a recommendation task is set to be:
  -Set of Users(U)
  -Set of items (I) that are to be recommended to U
  -Learn a function based on the user`s past interaction data thatpredicts the likeness of item I to U

- The implementation of the system consists of many sub-sections which are standard processes to be followed while solving any problem.
  These are as follows:

- ❖ Dataset
- ❖ Data Cleaning
- ❖ Model Analysis
- ❖ Model Building

How does a Recommendation Engine work?

A typical recommendation engine processes data through the followingfour phases namely collection, storing, analysing and filtering.

Collection of data :-

The first step in creating a recommendation engine is gathering data. Data can be either explicit or implicit data. Explicit data would consist ofdata inputted by users such as ratings and comments on products. And implicit data would be the order history/return history, Cart events, Pageviews, Click thru and search log. This data set will be created for every user visiting the site.

Behaviour data is easy to collect because you can keep a log of user activities on your site. Collecting this data is also straightforward becauseit doesn't need any extra action from the user; they're already using the application. The downside of this approach is that it's harder to analyse the data. For example, filtering the needful logs from the less needful ones can be cumbersome.

Since each user is bound to have different likes or dislikes about a product, their data sets will be distinct. Over time as you 'feed' the enginemore data, it gets smarter and smarter with its recommendations so that your email subscribers and customers are more likely to engage, click and buy. Just like how the Amazon's recommendation engine works withthe 'Frequently bought together' and 'Recommended for you' tab.

Storing the data :-

The more data you can make available to your algorithms, better the recommendations will be. This means that any recommendations projectcan quickly turn into a big data project.

The type of data that you use to create recommendations can help you decide the type of storage you should use. You could choose to use a NoSQL database, a standard SQL database, or even some kind of objectstorage. Each of these options is viable depending on whether you're capturing user input or behaviour and on factors such as ease of implementation, the amount of data that the storage can manage, integration with the rest of the environment, and portability.

When saving user ratings or comments, a scalable and managed database minimizes the number of tasks required and helps to focus onthe recommendation. Cloud SQL fulfils both of these needs and also makes it easy to load the data directly from Spark.

Analysing the data :-

How do we find items that have similar user engagement data? In orderto do so, we filter the data by using different analysis methods. If you want to provide immediate recommendations to the user as they are viewing the product then you will need a more nimble type of analysis.Some of the ways in which we can analyse the data are:

- Real-time systems can process data as it's created. This type of system usually involves tools that can process and analyse streamsof events. A real-time system would be required to give in-the- moment recommendations.
- Batch analysis demands you to process the data periodically. Thisapproach implies that enough data needs to be created in order tomake the analysis

relevant, such as daily sales volume. A batch system might work fine to send an e-mail at a later date.

- Near-real-time analysis lets you gather data quickly so you can refresh the analytics every few minutes or seconds. A near-real- time system works best for providing recommendations during thesame browsing session.

Filtering the data :-

Next step would be to filter the data to get the relevant data necessary to provide recommendations to the user. We have to choose an algorithm that would better suit the recommendation engine from the list of algorithms explained above. Like

- Content-based: A popular, recommended product has similarcharacteristics to what a user views or likes.
- Cluster: Recommended products go well together, no matter whatother users have done.
- Collaborative: Other users, who like the same products as anotheruser views or likes, will also like a recommended product.

Collaborative filtering enables you to make product attributes theoretical and make predictions based on user tastes. The output of this filtering is based on the assumption that two users who liked the same products in the past will probably like the same ones now or in the future.

You can represent data about ratings or interactions as a set of matrices,with products and users as dimensions.

Filtering techniques for 'Recommender System'

Have we ever wondered why the ads on Facebook are so relevant to what we're interested in or how the "movie-match" on Netflix works? Is it magic?

No. In both cases, a recommendation engine or system makes predictions based on our historical behaviour.

OFFLINE RECOMMENDATION ENGINES :-

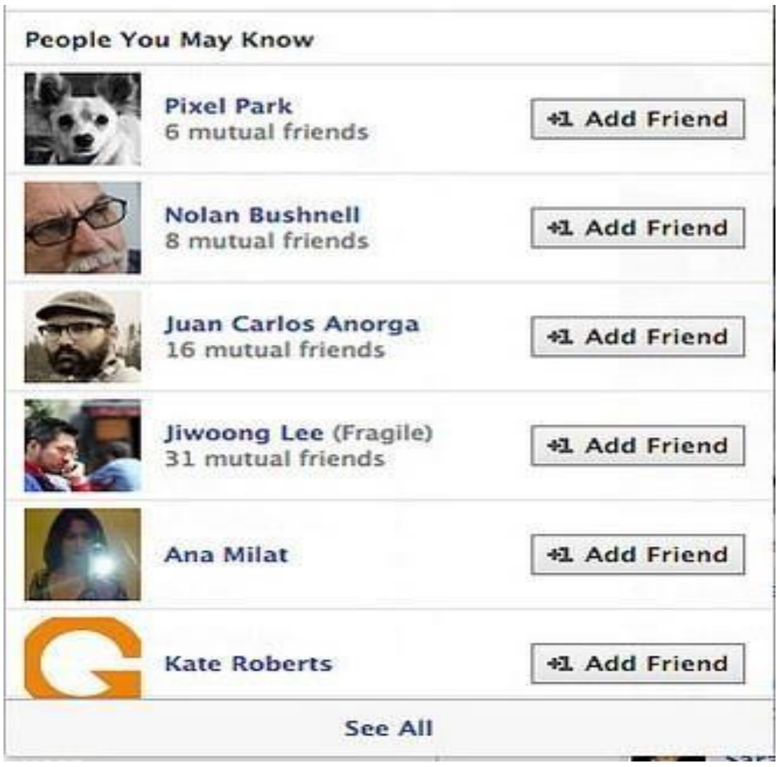
In the external world, we can think of the people around us as recommendation engines.

- **Your family and friends as clothes recommendation engines:** With the thousands of style options now available to us, we often rely on friends and family to recommend stores, styles andtell us what looks good on us.
- **Your Professors and book recommendation engines:** Whenwe want to research or better understand a concept, our Professors can lead us to the titles which best suit our needs
- **Your friends as movie recommendation engines:** If you have friends who know your cinematic tastes well, you're likely to trust their movie recommendations over a random stranger's picks.

Notice that all of these "offline recommenders" know somethingabout *you.* They know your style, taste or area of study, and thus can makemore informed decisions about what to recommendations would benefit you most. It is this personalisation- based on getting to "know" you - thatonline recommenders aim to emulate.

ONLINE RECOMMENDATION ENGINES :-

Facebook: "People You May Know"



People You May Know

Pixel Park
6 mutual friends
+1 Add Friend

Nolan Bushnell
8 mutual friends
+1 Add Friend

Juan Carlos Anorga
16 mutual friends
+1 Add Friend

Jiwoong Lee (Fragile)
31 mutual friends
+1 Add Friend

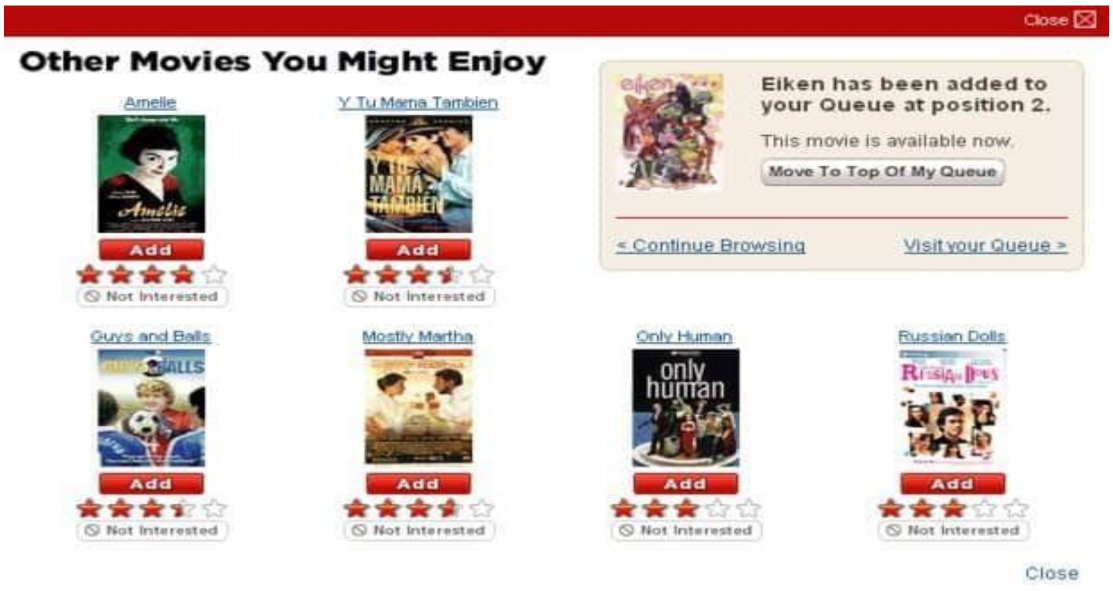
Ana Milat
+1 Add Friend

Kate Roberts
+1 Add Friend

See All

Facebook users a recommender system to suggest Facebook users you may know offline. The system is trained on personal data mutual friends,where you went to school, places of work and mutual networks (pages, groups, etc.), to learn who might be in your offline & offline network.

Netflix: "Other Movies You Might Enjoy"

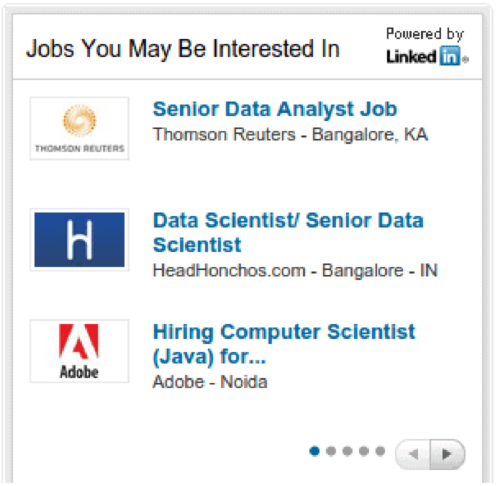


When you fill out your Taste Preferences or rate movies and TV shows, you're helping **Netflix** to filter through the thousands of selections to geta better idea of what you might like to watch. Factors that Netflix algorithm uses to make such recommendations include:

- The genre of movies and TV shows available
- Your streaming history, and previous ratings you've made.
- The combined ratings of all Netflix members who have similar tastesin titles to you.

LinkedIn: "Jobs You May be Interested In"

-


Jobs You May Be Interested In
Powered by Linked in

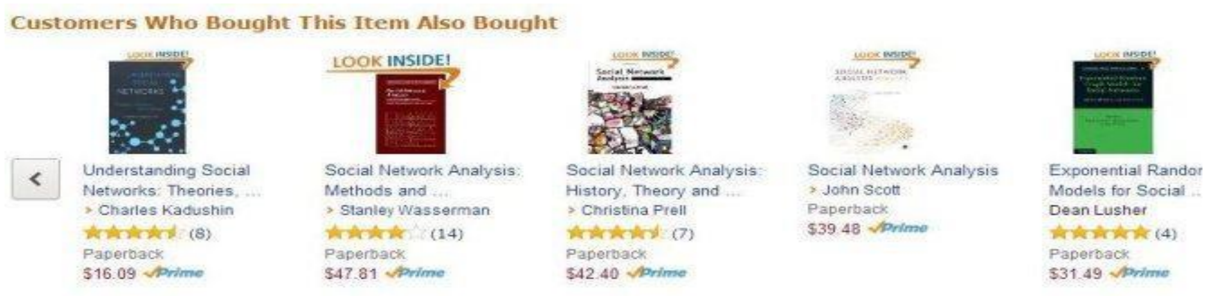
Senior Data Analyst Job
Thomson Reuters - Bangalore, KA

Data Scientist/ Senior Data Scientist
HeadHonchos.com - Bangalore - IN

Hiring Computer Scientist (Java) for...
Adobe - Noida

The Jobs You May Be Interested In feature shows jobs posted on **LinkedIn** that match your profile in some way. Theserecommendations shown based on the titles and descriptions in your previous experience, and the skills other users have "endorsed".

Amazon: “Customers Who Bought This Item Also Bought.


Customers Who Bought This Item Also Bought

Understanding Social Networks: Theories, ... › Charles Kadushin (8) Paperback $16.09 Prime

Social Network Analysis: Methods and ... › Stanley Wasserman (14) Paperback $47.81 Prime

Social Network Analysis: History, Theory and ... › Christina Prell (7) Paperback $42.40 Prime

Social Network Analysis › John Scott Paperback $39.48 Prime

Exponential Randor Models for Social ... Dean Lusher (4) Paperback $31.49 Prime

WHY SHOULD WE USE RECOMMENDATION ENGINES ?

In the immortal words of **Steve Jobs:** “A lot of times, people don’t knowwhat they want until you show it to them.” Customers may love your movie, your product, your job opening- but they may not know it exists. The job of the recommender system is to open the customer/user up to a whole new products and possibilities, which they would not think to directly search for themselves.

3 Key Techniques of Filtering areas follows :

- **Content based Filtering**
- **Collaborative Filtering**
- **Hybrid Filtering**

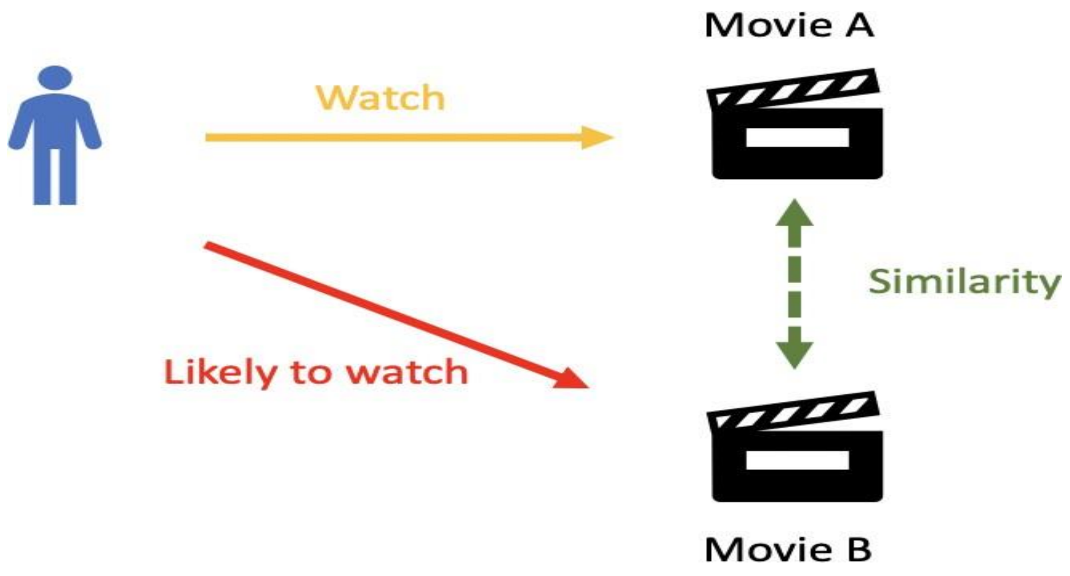
Content-Based Filtering :-

What is content-based filtering & How does it works?

Content-based filtering is a type of recommender system that attempts to guess what a user may like based on that user’s activity.

Content-based filtering makes recommendations by using keywords and attributes assigned to objects in a database (e.g., items in an online marketplace) and matching them to a user profile. The user profile is created based on data derived from a user’s actions, such as purchases, ratings (likes and dislikes), downloads, items searched for on a website and/or placed in a cart, and clicks on product links.

A Content-based recommendation system tries to recommend items to users based on their profile. The user’s profile revolves around that user’spreferences and tastes. It is shaped based on user ratings, including the number of times that user has clicked on different items or perhaps even liked those items. The recommendation process is based on the similaritybetween those items. Similarity or closeness of items is measured based on the similarity in the content of those items. When we say content, we're talking about things like the items category, tag, genre, and so on.



For example, suppose you're recommending accessories to a user that just purchased a smartphone from your website and has previously bought smartphone accessories. Aside from keywords such as the smartphone manufacturer, make, and model, the user profile indicates prior purchases include phone holders with sleeves for credit cards. Based on this information, the recommender system maysuggest similar phone holders for the new phone with attributes suchas an RFID blocking fabric layer to help prevent unauthorized credit card scanning. In this example, the user would expect recommendations for similar phone holders, but the RFID blocking feature may be something they didn't expect yet appreciate

Assigning attributes :-

Content-based filtering relies on assigning attributes to database objects so the algorithm knows something about each object. These attributes depend primarily on the products, services, or content you're recommending.

Assigning attributes can be a monumental undertaking. Many companies resort to using subject-matter expert teams to assignattributes to each item manually. For example, Netflix has hired screenwriters to **rate shows** on aspects ranging from shooting locations and actors to plotlines, tone, and emotional effects. The resulting tags, used by the recommender, are algorithmically combined to group films together that share similar aspects.

Building a user profile :-

User profiles are another element crucial to content-based recommender systems. Profiles include the database objects the user has interacted with—purchased, browsed, read, watched, or listened to—as well as their assigned attributes.

Attributes appearing across multiple objects are weighted more heavily than those that show up less often. This helps establish a degree of importance because not all of an object's attributes are equal to the user. User feedback is also critical when weighting items, whichis why websites that provide recommendations are continually asking you to rate products, services, or content.

Based on attribute weightings and histories, the recommender system produces a unique model of each user's preferences. The model consists of attributes the user is liable to like or dislike based on past activities, weighted by importance. User models are compared against all database objects, which are then assigned scores based on their similarity to the user profile.

Here's an example: Let's say you've listened to Taylor Swift's "The Last Time," Shakira's "Can't Remember to Forget You," and "Me, Myself and I" by Beyoncé. A recommender system might recognize that you like female pop artists and breakup songs. You could expect to receive recommendations for more breakup songs by these and other female pop artists, such as Miley Cyrus's "Slide Away."

The recommender system may also suggest different types of songs by Miley Cyrus because you appear to like female pop artists. Still, sinceyou didn't choose to listen to this artist or songs not associated with breakups before, these selections would receive a lower assigned score.

Why it is used?

If you are a fan of science fiction movies and have watched Star Wars, therecommendation engine may suggest that you watch Avatar. This methodis known as **content-based filtering** because it analyses the content ofeach item and finds similar items. While very useful, it requires a thoroughknowledge of each item in order to find similar items.

- The idea here is to recommend similar items to the ones you liked before. The system first finds the similarity between all pairs of articles and then uses the articles most similar to the articles already evaluated by a user to generate a list of recommendations.

- No data from other users is required to start making recommendations. Unlike collaborative filtering, content- based filtering doesn't need data from other users to create recommendations. Once a user has searched on and browsed a few items and/or completed some purchases, a content-based filtering system can begin making relevant recommendations.

This makes it ideal for businesses that don't have an enormous pool of users to sample. It also works well for sellers that have many users but a small number of user interactions in specific categories or niches.

- Recommendations are highly relevant to the user. Content- based recommenders can be highly tailored to the user's interests, including recommendations for niche items, because the method relies on matching the characteristics or attributes of a database object with the user's profile. For instance, content-based filtering will recognize a specific user's preferences and tastes, such as hot sauces made in Texas with organic Scotch bonnet peppers and recommend products with the same attributes. Content-based filtering is also valuable for businesses with extensive libraries containing a single type of product, such as smartphones, where recommendations need to be based on many discrete features.

- Recommendations are transparent to the user. Highly relevant recommendations project a sense of openness to the user, bolstering their trust level in offered recommendations. Comparatively, with collaborative filtering, instances are more likely to occur where users don't understand why they see specific recommendations. For example, let's say a group of users who purchased an umbrella also happen to buy down puffer coats. A collaborative system may recommend down puffer coats to other users who bought umbrellas but are uninterested in and have never browsed or purchased thatproduct.

- You avoid the "cold start" problem. Collaborative filtering creates a potential cold start scenario when a new website or community has few new users and lacks user connections. Although content-based filtering needs some initial inputs from users to start making recommendations, the quality of early recommendations is generally better than a collaborative system that requires the addition and correlation of millions of data points before becoming optimized.

- Content-based filtering systems are generally easier to create. The data science behind a content-based filtering system is relatively straightforward compared to collaborative filtering systems intended to mimic user-to-user recommendations. The real work in content-based filtering is assigning the attributes.

Advantages of content-based filtering :-

- This has the capability of mentioning unrated objects
- It is easy to explain the work of recommender system by cataloguethe content features of a item.
- Content-based recommender systems require only the rating of the concerned client, no one else's.
- Operator of the system.

How can you find similarities between items ?

To compare two items, we need to transform them into mathematical objects such as vectors, on which we can compute metrics suchas *Euclidean distance, Pearson's coefficient* or *cosine similarity*. Below are expressed the formulas of the different metrics exposed below:

$$Euclidean\ distance(u,p) = \sum_i (u_i - p_i)^2$$

$$Pearson's\ coefficient(u,p) = \frac{\sum_i (u_i - \bar{u})(p_i - \bar{p})}{\sqrt{\sum_i (u_i - \bar{u})^2}\sqrt{\sum_i (p_i - \bar{p})^2}}$$

$$Cosine\ similarity(u,p) = \frac{\sum_i u_i p_i}{\sqrt{\sum_i {u_i}^2}\sqrt{\sum_i {p_i}^2}}$$

A commonly used technique is the **TF-IDF** (frequency term — inverted document frequency). It is a statistical measure that evaluates the relevance of a word to a document in a set of documents. This is done by multiplying two measures: the number of times a word appears in a document (term frequency) and the inverse of the number of documents in which the word appears (inverse document frequency). If a film is a science fiction film like Star Wars is, the science fiction word may appeara lot in the film description, so the term frequency will be high, and if thereare not many science fiction films in the film corpus, the inverse of the termfrequency will also be high. Finally, after normalization, the value of TF-IDF will be close to one.

If we apply this technique to a set of films, after normalization, we can obtain the following table. The numbers in this table are random plausible values.

Movies	Science-fiction	Action	Romantic
Star Wars	0.92	0.87	0.35
Avatar	0.93	0.7	0.5
Titanic	0.05	0.2	0.9

Now that we have vectors that describe a movie, we can calculate the *Euclidean distance, the Pearson coefficient, or the Cosinesimilarity* between two of these vectors. For example, the Euclidean distance between the Star Wars and Avatar vectors is 0.05, while the distance between Star Wars and Titanic is 1.5. The closer the result is to 0,the more similar the elements are. Thus, we can clearly see how close StarWars and Avatar are according to the characteristics we have chosen (Science Fiction, Action, Romantic) and how far apart Star Wars and Titanic are.

As you can see, it is essential to know the content of each element for thismethod.

Challenges of content-based filtering :-

Like all recommender systems, content-based filtering has both prosand cons.

- There's a lack of novelty and diversity. There's more to recommendations than relevance. Suppose you liked the movie *Tenet*. Chances are you'll like *Inside Man,* too. But there's a high probability you don't need a recommender system to tell you this. So, to be of value, recommendation engines must come up with diverse and unexpected results.
- Scalability is a challenge. Every time a new product or service or new content is added, its attributes must be defined and tagged. The arduous, never-ending nature of attributeassignments can make scalability difficult and time-consuming.
- Attributes may be incorrect or inconsistent. Content-based recommendations are only as good as the subject-matter experts tagging items. Potentially millions of items may need attributes assigned, and since attributes can be subjective, many may be incorrectly tagged. A process that ensures attributes are applied consistently and accurately is paramount. Otherwise, a content-based recommender system will not function as intended.

Disadvantage of content-based filtering :-

- A new user who has not rated any item will not work for them, as enough ratings are required content-based recommender assesses the client inclinations and gives exact proposals.
- No proposals of unexpected objects.
- Restricted Content Study - If the system fails to distinguish the itemsthe recommender does not work.
- A client likes items that he/she does not like

Skills and Technology you need to build a content-based filtering system :-

Building a recommendation engine is a classic machine learning exercise. Not only should **data scientists** have experience with the tools of statistical analysis, but they should also be familiar with tools and frameworks that provide an infrastructure for building recommendation engines. These include programming languages like Python and Scala, and libraries and frameworks such as Hadoop/Spark Matplotlib, and Neo4. Which ones are appropriate for your project will depend on what exactly you're trying to accomplish.

Next steps :-

Recommender systems such as content-based filtering benefit both sellers and buyers. Buyers can spend less time searching through pages of different products in a digital marketplace. Sellers can better understand customer preferences, provide a more personalized buyer experience, increase sales, and build brand loyalty by using content-based filtering.

Consider staff augmentation if you don't have the available staff or staff with the appropriate expertise to create or implement content- based filtering. Upwork enables you to hire independent recommender systems specialists with the confidence and ease of using the world's work marketplace.

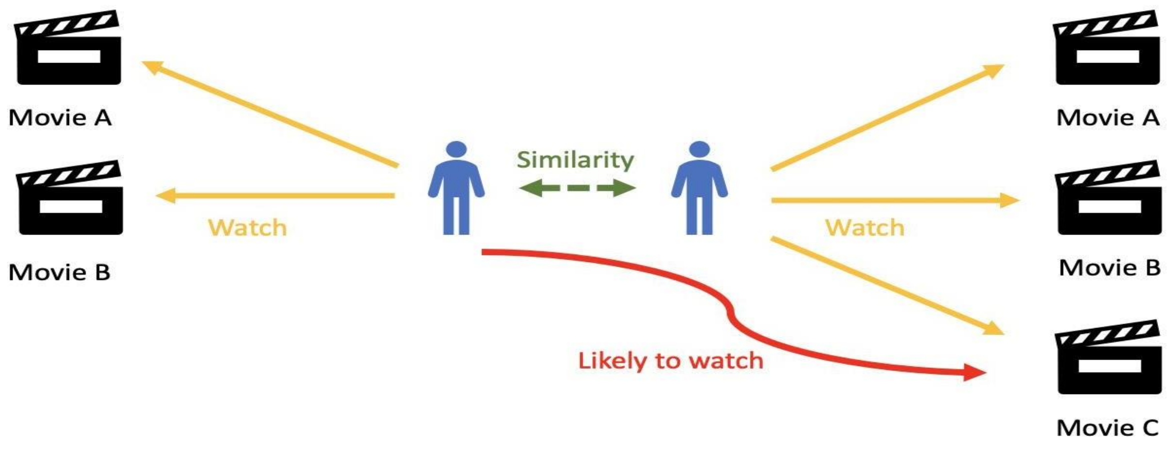
Collaborative Filtering :-

What is Collaborative filtering & How does it works ?

Collaborative filtering (CF) is a **technique used by recommender systems**. In the newer, narrower sense, collaborative filtering is a method

of making automatic predictions (filtering) about the interests of a user by collecting preferences or taste information from many users (collaborating).

Collaborative filtering methods are based on collecting and analysing a large amount of information on users' behaviours, activities or preferences and predicting what users will like based on their similarity toother users. A key advantage of the collaborative filtering approach is thatit does not rely on machine analysable content and therefore it is capable of accurately recommending complex items such as movies without requiring an "understanding" of the item itself. Many algorithms have been used in measuring user similarity or item similarity in recommendersystems. For example, the k-nearest neighbour (k-NN) approach and the Pearson Correlation.



Why it is used?

Collaborative filtering is based on the fact that relationships exist between products and people’s interests. Many recommendation systemsuse collaborative filtering to find these relationships and to give an accurate recommendation of a product that the user might like or be interested in.

Collaborative filtering doesn’t need anything else but users’ historical
preference on a set of items.

The standard method of Collaborative Filtering is known as the **NearestNeighbourhood** algorithm. We have an n × m matrix of ratings, that wewill call R, the user matrix is denoted as U and the item matrix as P. The user i is represented by the vector u_i and they are n users so i = 1, …n. The

item is represented by the vector p_j and they are m items so j=1, …m. Now
we want to predict the rating r_{ij} if target user i did not watch/rate an item
j. The process is to calculate the similarities between target user i and all other users, select the top X similar users, and take the weighted average of ratings from these X users with similarities as weights. The rating r_{ij} is defined as:

$$r_{ij} = \frac{\sum_k Similarities(u_k, u_i) * r_{kj}}{number\ of\ ratings}$$

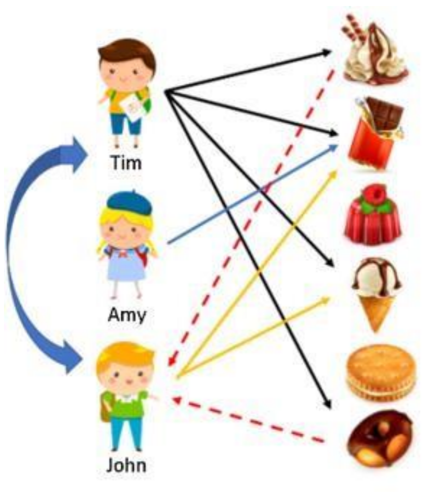
There are two classes of Collaborative Filtering:

-***User-based :-***

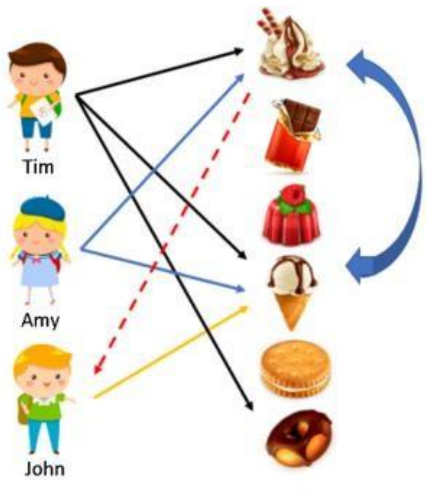
Which measures the similarity between target users and other users.
In user-based collaborative filtering, we have an active user for whom therecommendation is aimed. The collaborative filtering engine first looks forusers who are similar. That is users who share the active users rating patterns.
Collaborative filtering basis this similarity on things like history, preference, and choices that users make when buying, watching, or enjoying something.

-***Item-based :-***

Which measures the similarity between the items that target users rate orinteract with and other items.
In the item-based approach, similar items build neighbourhoods on thebehaviour of users.



(a) User-based filtering



(b) Item-based filtering

Advantage of collaborative filtering :-

- It relies upon the connection between clients which infers that it is content-autonomous.
- Collaborative Filtering recommender systems can propose unexpected items by detecting like-minded person's behaviour.
- They can make genuine quality evaluation of objects by considering other person's experience

Challenges of Collaborative Filtering :-

-*Data sparsity*

Data sparsity happens when you have a large data set of users who generally rate only a limited number of items. As mentioned, collaborativebased recommenders can only predict scoring of an item if there are otherusers who have rated it. Due to sparsity, we might not have enough ratingsin the user item dataset which makes it impossible to provide proper recommendations.

-*Cold start*

Another issue to keep in mind is something called cold start. Cold start refers to the difficulty the recommendation system has when there is a newuser, and as such a profile doesn't exist for them yet. Cold start can also happen when we have a new item which has not received a rating.

-Scalability

Scalability can become an issue as well. As the number of users or itemsincreases and the amount of data expands, collaborative filtering algorithms will begin to suffer drops in performance, simply due to growthand the similarity computation.

Disadvantage of content-based filtering :-

- Early rating problem: Collaborative filtering systems cannot suggest for fresh items as there are no client ratings on which to base a forecast.
- Gray sheep: In order to Collaborative filtering-based system to work, group with related features are required. It will be very tough to suggest clients who do not steadily agree or disagree to these groups, if such groups exist.
- Sparsity problem: Mostly, the number of items exceeds the number of clients by a huge margin which makes it tough to finditems that are rated by ample individuals.

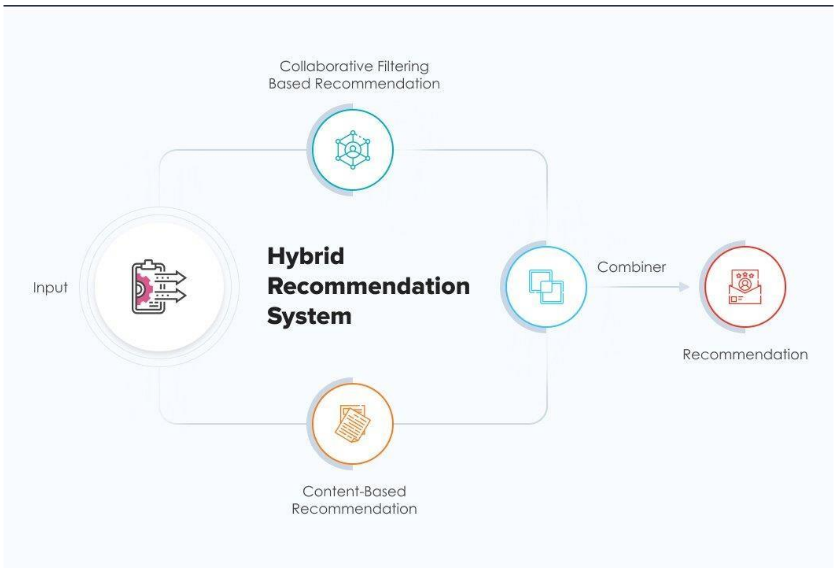
Hybrid Filtering :-

What is Hybrid filtering & How does it work?

The hybrid filtering is a blend of content-based or collaborative filtering methods. Content based or collaborative filtering methods are executed individually and the outputs of both methods are recommended to jointlysuggest better.

The spread of distance between feature vectors can also be used to calculate the similarity of two objects:

- **Content-based similarity**
  - ***Weight from collaborative network***


Collaborative Filtering Based Recommendation
Input
Hybrid Recommendation System
Combiner
Recommendation
Content-Based Recommendation

Recent research has demonstrated that a hybrid approach, combining collaborative filtering and content-based filtering could be more effectivein some cases. Hybrid approaches can be implemented in several ways, bymaking content-based and collaborative-based predictions separately andthen combining them, by adding content-based capabilities to a collaborative-based approach (and vice versa), or by unifying the approaches into one model.

Hybrid recommendation systems came into existence due to thelimitations of each of the previous kinds. Hence there have been several strategies to combine Collaborative Filtering and Content-Based Filteringand is called Hybrid Recommender Engine. Amongst the various approaches used, **weighted method** is the most common. In the beginning, the combination of the recommendation results is obtained from each and equal weights are distributed to each of these results and gradually the weights are adjusted after evaluating the responses from theusers to the recommendations. **Feature Combination method** is another popular approach where the User profile from Content-Based Filtering is combined with user-item ratings information and a new strategy is considered to build a Hybrid Recommendation System.

Why it is used?

Several studies empirically compare the performance of the hybrid with the pure collaborative and content-based methods and demonstrate that the hybrid methods can provide more accurate recommendations than pure approaches. These methods can also be used to overcome some of the common problems in recommendation systems such as cold start andthe sparsity problem.

Types of Hybrid filtering methods :-

Hybrid filtering methods combine CBF and CF methods to overcome the drawbacks of both CBF and CF and benefit from their advantages. To combine CBF and CF the following three methods are suggested:

Weighted combination:

At first, CBF and CF methods are applied separately, then their resulting predictions are combined to achieve the final predictions.

Mixed combination:

At first, CBF and CF methods are applied to obtain recommendations separately, then the two lists of recommendations are combined together to achieve the final list of recommendations.

Sequential combination:

Initially user features are formed by CBF methods. When there are enoughratings for the users, CBF methods can be replaced by CF methods to obtain the final recommendations.

Netflix is a good example of a hybrid filtering system. They makerecommendations by comparing the watching and searching habits of similar users (i.e. collaborative filtering) as well as by offering movies thatshare characteristics with films that a user has rated highly (content-basedfiltering).

Evaluation metrics for Recommender systems

Given a choice between two or more different recommender systems, howdoes one select the best system? How does one compare and contrast thestrengths and weaknesses of a variety of systems?
One can potentially use one or more criteria that are generally used for systems deployed in production, such as memory and CPU footprint, as well as the type, quality and availability of the data that is fed as input intothe system.
For example, let us consider a movie recommender system that considers only the ratings a user has provided over the past month versus a movie recommender system that considers all ratings the user has ever made. When the user base is small, (< 1M users), there probably isn't much of a difference in the time taken to retrieve 1 month's worth of dataversus the time taken to retrieve all the ratings for a particular user. However, if the user base is sufficiently large, this difference increases. Inthis case, it would be more prudent to choose the system that relies on a smaller quantum of data to make its predictions.
However, recommender systems are primarily gauged by their predictionaccuracy.

There are three categories of metrics, depending on the kind ofprediction :-

Rating prediction accuracy :-

Netflix rating predictions exemplifies this. The ratings can be binary(thumbs up/thumbs down) or on a scale (1 star to 5 stars).

Root mean squared error (RMSE) :

This is perhaps the most popular metric used in evaluating accuracy of predicted ratings. The system generates predicted ratings for a test set of user-item pairs for which the true ratings are known. Typically, are knownbecause they are hidden in an offline experiment, or because they were obtained through a user study or online experiment. The RMSE between the predicted and actual ratings is given by:

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

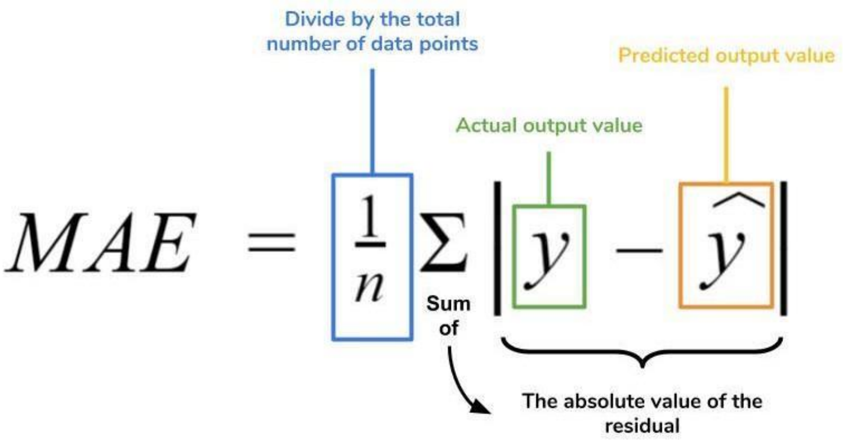
$$\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n \text{ are predicted values}$$

$$y_1, y_2, \ldots, y_n \text{ are observed values}$$

$$n \text{ is the number of observations}$$

Mean absolute error (MAE) :-

- It is a popular alternative
- Compared to MAE, RMSE disproportionately penalizes large errors.For instance, given a test set with four hidden items, RMSE would prefer a system that makes an error of 2 on three ratings and 0 on the fourth over a system that makes an error of 3 on one rating and0 on all three others. MAE would prefer the second system.
- MAE is one of the many metrics for summarizing and assessing the quality of a machine learning model. Here, error refers to the subtraction of Predicted value from Actual Value as below.
- $Prediction\ Error = Actual\ Value - Predicted\ Value$
- This prediction error is taking for each record after which we convert all error to positive. This is achieved by taking Absolute value for each error as below:
- $Absolute\ Error \rightarrow |Prediction\ Error|$
- Finally, we calculate the mean for all recorded absolute errors **(Average sum of all absolute errors).**


Divide by the total number of data points
Predicted output value
Actual output value
$MAE = \frac{1}{n}\sum\left|y - \hat{y}\right|$
Sum of
The absolute value of the residual

Usage prediction accuracy :-

Several applications do not require predictions of the user's preferences in the form of ratings, but they utilize lists of items thatthe user may choose from. For example, on a shopping website, based on a user's purchase history, the website might want to recommend a list of items the user might be interested in purchasing. Here, the number of items from the list the user actuallybought measures the success of the recommendations.

Thus, *precision* would be a good measure of usage prediction.

Precision= #TruePositives/#TruePositives+#FalsePositives

A more useful metric is *precision at n*, which is defined as the value of precision considering only the top-*n* results. A broaderpicture of the system can be obtained by measuring precision at *n* for several values of *n*.

Conclusion

Recommender systems have the capability to provide users with customized and tailored recommendations. Thus, it resolves the issue of information overload that plagues the users at the modern age. Recently, various approaches for building recommendation systems have been developed, which can utilize either CollaborativeFiltering, Content Based Filtering or hybrid filtering.

- **Collaborative Filtering technique** has reached a certain maturity and is one of the most commonly implemented systems. A multitude of different application areas has adopted Collaborative Filtering based recommendation systems.
- One of them is 'GroupLens', a news-based architecture that assists users to locate articles from huge news databases usingcollaborative methods.
- Amazon improved its recommendation system by implementing topic diversification algorithms.
- On the other hand, **Content-Based Filtering techniques** focus finding the similarities between content properties and user characteristics. Content-Based Filtering techniques normally base their predictions on user's information, and they ignore contributions from other users as with the case ofcollaborative techniques.
- Letizia predicts the pages that a user may be interested in by tracing his movements though the sites and thus uses Content-Based Filtering method.
- Despite the widespread success of these two filtering techniques, they have several limitations.
- While Content-Based Filtering techniques have issues likelimited content analysis, overspecialization and sparsity of data, **collaborative approaches** have issues like cold-start, sparsity, and scalability. These issues make it difficult to use these systems 27 in live production.
- **Hybrid filtering,** using the combination of two or more filtering techniques in different ways to increase the performance and accuracy of recommender systems has thus been proposed to alleviate these issues.
- These hybrid systems try to leverage the strength of each method without sacrificing any capability due to inherentweakness.

Reference

- ✓ www.displayr.com
- ✓ www.dummies.com
- ✓ www.towarsdatascience.com
- ✓ www.medium.com
- ✓ www.dataconomy.com
- ✓ www.packtpub.com
- ✓ www.researchgate.net
- ✓ www.sciencedirect.com
- ✓ www.projectspro.io