How to build Recommender System using Machine Learning?

Introduction

With the increased accessibility to the internet, each business sector has expanded at an extreme speed. We must be aware of Amazon, Flipkart, YouTube, Instagram, and many such platforms. But h**ave we ever thought about the sole reason behind their success?** The critical point is hidden behind offering a better customer experience with the help of personalization. But now we must have been thinking about how they work on personalization? Companies use **Recommendation Systems** to personalize their users/customers better.

On **YouTube's** home page, we find similar content to the content which we enjoyed watching earlier. Similarly, other platforms like **Netflix, Spotify, Amazon, Instagram,** etc., try to increase their users' retention time over their platform. All this has become possible because of the advancement in the Recommendation System. So let's dive into the concept and its working principle of it.

Key features of this blog

In this blog, an end-to-end description of the Recommender System is presented. This blog answers these mentioned questions in a detailed manner:

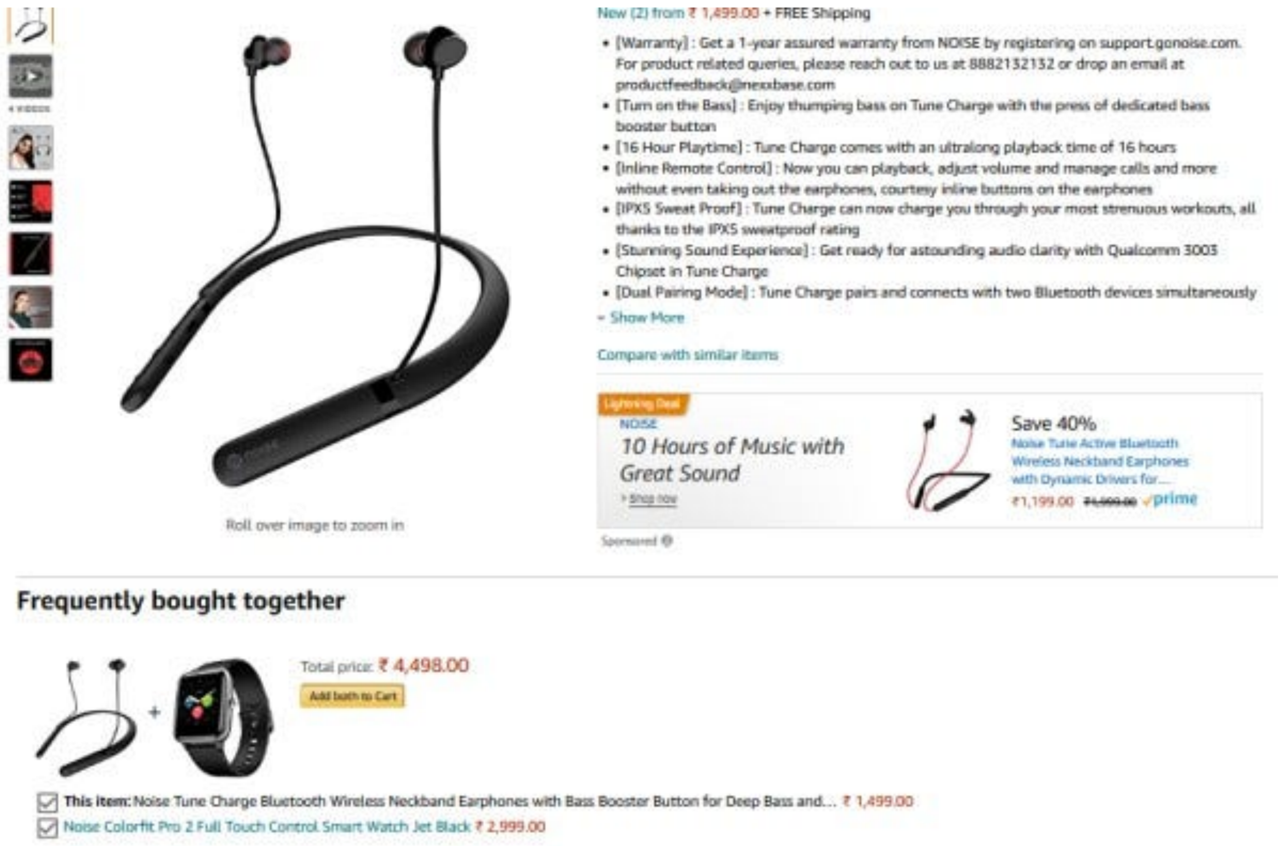
1. What is Recommender System and why they are used?
2. What are different types of Recommender Systems and their pros and cons?
3. How to implement content-based Recommender Systems?
4. How are tech-giant companies using recommender systems?

What is Recommender System?

It refers to a kind of system that could predict the future preference of users based on their previous behavior or by focusing on similar users' behavior. In a nutshell, **Recommender systems** are like salespeople who know us (our likings and dislikings) very well and suggest products that would attract us the most.

How are Recommender Systems revolutionizing the world?

Recommender systems have transformed the traditional way of focusing on what customers want to buy next. They are aimed to analyze customer choice better and constantly keep evolving itself based on users' interactions. As shown in Figure 1, if someone is looking to buy a Bluetooth neckband, then a recommender system will show them other neckbands and a list of more products with some lucrative offers.



New (2) from ₹ 1,499.00 + FREE Shipping

- [Warranty] : Get a 1-year assured warranty from NOISE by registering on support.gonoise.com. For product related queries, please reach out to us at 8882132132 or drop an email at productfeedback@nexxbase.com
- [Turn on the Bass] : Enjoy thumping bass on Tune Charge with the press of dedicated bass booster button
- [16 Hour Playtime] : Tune Charge comes with an ultralong playback time of 16 hours
- [Inline Remote Control] : Now you can playback, adjust volume and manage calls and more without even taking out the earphones, courtesy inline buttons on the earphones
- [IPX5 Sweat Proof] : Tune Charge can now charge you through your most strenuous workouts, all thanks to the IPX5 sweatproof rating
- [Stunning Sound Experience] : Get ready for astounding audio clarity with Qualcomm 3003 Chipset in Tune Charge
- [Dual Pairing Mode] : Tune Charge pairs and connects with two Bluetooth devices simultaneously

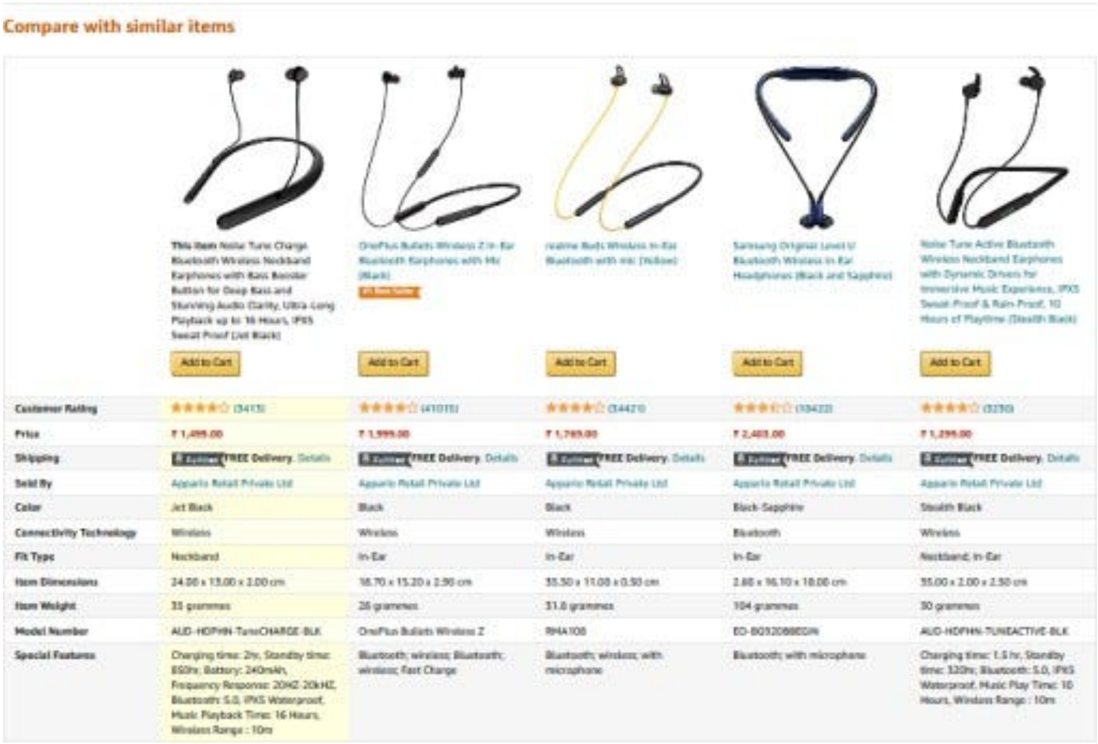
4 VIDEOS
Roll over image to zoom in
Show More
Compare with similar items
Lightning Deal
NOISE
10 Hours of Music with Great Sound
› Shop now
Save 40%
Noise Tune Active Bluetooth Wireless Neckband Earphones with Dynamic Drivers for...
₹1,199.00 ~~₹1,999.00~~ prime
Sponsored

Frequently bought together

Total price: ₹ 4,498.00

☑ **This item:** Noise Tune Charge Bluetooth Wireless Neckband Earphones with Bass Booster Button for Deep Bass and... ₹ 1,499.00
☑ Noise Colorfit Pro 2 Full Touch Control Smart Watch Jet Black ₹ 2,999.00

Traditionally people have to look across each product individually and go through the list of various individual items. If a customer searched for a neckband, they would be shown a neckband list from where they would pick one. But with the advent in place, now customers automatically be shown another similar list of neckbands and a comparison between all the possible combinations, as shown in Figure 2. This gives rise to a better and informed choice for the customer, and as a result, their trust and satisfaction increase with the seller and the platform.



Compare with similar items

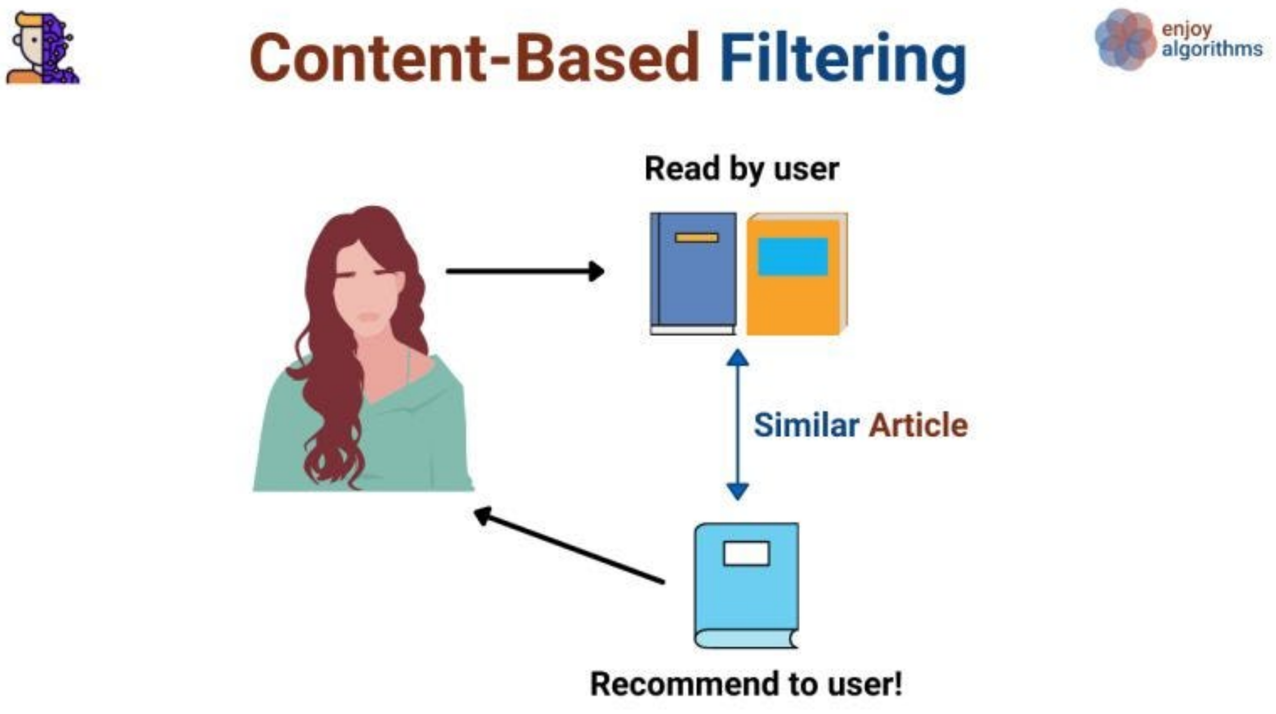
	This item Noise Tune Charge Bluetooth Wireless Neckband Earphones with Bass Booster Button for Deep Bass and Stunning Audio Clarity, Ultra Long Playback up to 16 Hours, IPX5 Sweat Proof (Jet Black)	OnePlus Bullets Wireless Z In-Ear Bluetooth Earphones with Mic (Black) #1 Best Seller	realme Buds Wireless In-Ear Bluetooth with mic (Yellow)	Samsung Original Level U Bluetooth Wireless In-Ear Headphones (Black and Sapphire)	Noise Tune Active Bluetooth Wireless Neckband Earphones with Dynamic Drivers for Immersive Music Experience, IPX5 Sweat-Proof & Rain-Proof, 10 Hours of Playtime (Stealth Black)
	Add to Cart	Add to Cart	Add to Cart	Add to Cart	Add to Cart
Customer Rating	★★★★☆ (3413)	★★★★☆ (41015)	★★★★☆ (34421)	★★★☆☆ (10422)	★★★★☆ (3230)
Price	₹ 1,499.00	₹ 1,999.00	₹ 1,769.00	₹ 2,403.00	₹ 1,299.00
Shipping	Fulfilled FREE Delivery. Details	Fulfilled FREE Delivery. Details	Fulfilled FREE Delivery. Details	Fulfilled FREE Delivery. Details	Fulfilled FREE Delivery. Details
Sold By	Appario Retail Private Ltd	Appario Retail Private Ltd	Appario Retail Private Ltd	Appario Retail Private Ltd	Appario Retail Private Ltd
Color	Jet Black	Black	Black	Black-Sapphire	Stealth Black
Connectivity Technology	Wireless	Wireless	Wireless	Bluetooth	Wireless
Fit Type	Neckband	In-Ear	In-Ear	In-Ear	Neckband; In-Ear
Item Dimensions	24.00 x 13.00 x 2.00 cm	18.70 x 15.20 x 2.90 cm	35.30 x 11.00 x 0.50 cm	2.80 x 16.10 x 18.00 cm	35.00 x 2.00 x 2.50 cm
Item Weight	35 grammes	28 grammes	31.8 grammes	104 grammes	30 grammes
Model Number	AUD-HDPHN-TuneCHARGE-BLK	OnePlus Bullets Wireless Z	RMA108	EO-BG920BBEGIN	AUD-HDPHN-TUNEACTIVE-BLK
Special Features	Charging time: 2hr, Standby time: 850hr, Battery: 240mAh, Frequency Response: 20HZ-20kHZ, Bluetooth: 5.0, IPX5 Waterproof, Music Playback Time: 16 Hours, Wireless Range : 10m	Bluetooth; wireless; Bluetooth; wireless; Fast Charge	Bluetooth; wireless; with microphone	Bluetooth; with microphone	Charging time: 1.5 hr, Standby time: 120hr, Bluetooth: 5.0, IPX5 Waterproof, Music Play Time: 10 Hours, Wireless Range : 10m

Types of Recommender System

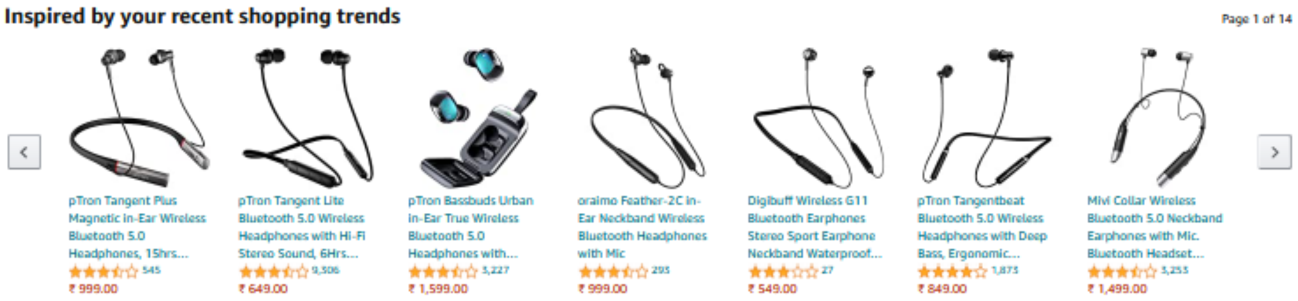
There are mainly **six types** of recommender systems based on their working principle. Some of the commonly used recommender systems are described below:

Content-Based Recommendation

Content-based recommender focuses on users' historical behaviors. The system keeps on adapting itself to be more and more personalized by accepting ratings or inputs from users. The following figure shows how this content-based recommendation system works.



Pros: In content-based systems, there is no cold start and popularity-based problem, and it can recommend items with rare features.
Cons: Content-based systems perform poorly as compared to other recommender systems and also suffer from the over-specialization problem. **Can you guess why?**



Inspired by your recent shopping trends

Page 1 of 14

pTron Tangent Plus Magnetic in-Ear Wireless Bluetooth 5.0 Headphones, 15hrs... 545
₹ 999.00

pTron Tangent Lite Bluetooth 5.0 Wireless Headphones with Hi-Fi Stereo Sound, 6Hrs... 9,306
₹ 649.00

pTron Bassbuds Urban in-Ear True Wireless Bluetooth 5.0 Headphones with... 3,227
₹ 1,599.00

oraimo Feather-2C in-Ear Neckband Wireless Bluetooth Headphones with Mic 293
₹ 999.00

Digibuff Wireless G11 Bluetooth Earphones Stereo Sport Earphone Neckband Waterproof... 27
₹ 549.00

pTron Tangentbeat Bluetooth 5.0 Wireless Headphones with Deep Bass, Ergonomic... 1,873
₹ 849.00

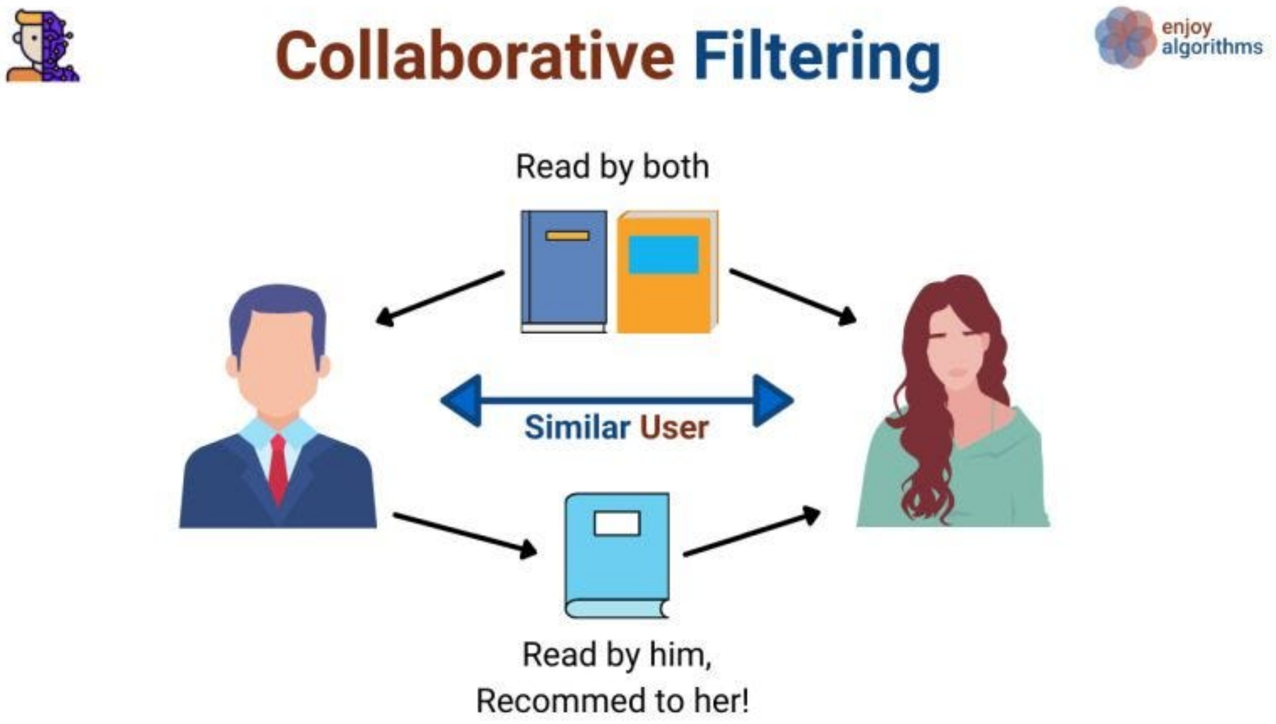
Mivi Collar Wireless Bluetooth 5.0 Neckband Earphones with Mic. Bluetooth Headset... 3,253
₹ 1,499.00

Collaborative Filtering

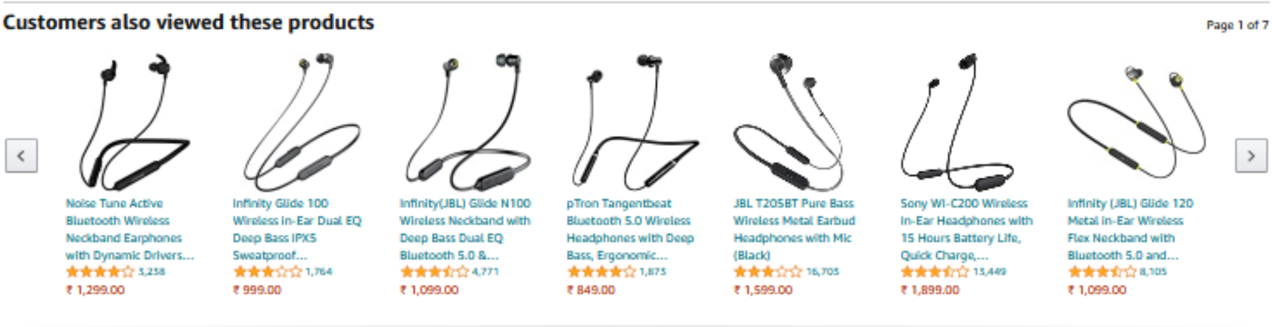
Collaborative filtering focuses on the similarity in the behavior of users. It is assumed that people who have similar behavior or who have agreed in the past about some common things will agree in the future. The system tries to determine users/items based on their rating history. That history is to be grouped and assigned based on their neighborhood.

The collaborative filtering approach does not rely on analyzable machine content, and hence it can accurately recommend complex items.

Collaborative Filtering working is shown below.



Pros: No feature selection is required; hence it works for any item.
Cons: Collaborative Filtering method suffers from cold start problems, sparsity, and popularity bias problems. **Can you guess why?**



Customers also viewed these products

Page 1 of 7

Noise Tune Active Bluetooth Wireless Neckband Earphones with Dynamic Drivers...
★★★★☆ 3,258
₹ 1,299.00

Infinity Glide 100 Wireless In-Ear Dual EQ Deep Bass IPX5 Sweatproof...
★★★☆☆ 1,764
₹ 999.00

Infinity(JBL) Glide N100 Wireless Neckband with Deep Bass Dual EQ Bluetooth 5.0 &...
★★★½☆ 4,771
₹ 1,099.00

pTron Tangentbeat Bluetooth 5.0 Wireless Headphones with Deep Bass, Ergonomic...
★★★★☆ 1,873
₹ 849.00

JBL T205BT Pure Bass Wireless Metal Earbud Headphones with Mic (Black)
★★★☆☆ 16,703
₹ 1,599.00

Sony WI-C200 Wireless In-Ear Headphones with 15 Hours Battery Life, Quick Charge,...
★★★½☆ 13,449
₹ 1,899.00

Infinity (JBL) Glide 120 Metal In-Ear Wireless Flex Neckband with Bluetooth 5.0 and...
★★★½☆ 8,105
₹ 1,099.00

Hybrid Recommender System

Both Content-based and Collaborative filtering-based recommender systems have some pros and cons. So it would be better to leverage both content-based and collaborative data and use it as a Hybrid system for recommendations. **Netflix** recommendations are solely based on the Hybrid Recommender systems. Whenever a new user subscribes to their service, they must have to rate the already seen content. Once the user began using the service, collaborative filtering suggests similar content to their customers.

Implementation of a Content-Based Recommender System

Let’s first look at the high-level view of implementing any simple Recommender system, and then we’ll dive a little deeper and look into the actual dataset. So let’s get started :).

Outline

- We will start with the user and find out the set of items that they liked or purchased; we will build an **item profile** (a description of the item).
- From these Item profiles, we could infer a **user profile**, a description of the user.
- Once we have a user profile, we use it to match the pool of item profiles using any similarity measures and recommend the most similar items to the user.

How to build an Item profile?

For each item, the Item profile consists of a set of important features given below:

1. **Movies:** Author, Title, Actor, etc.
2. **Images:** Tags, Metadata, etc.

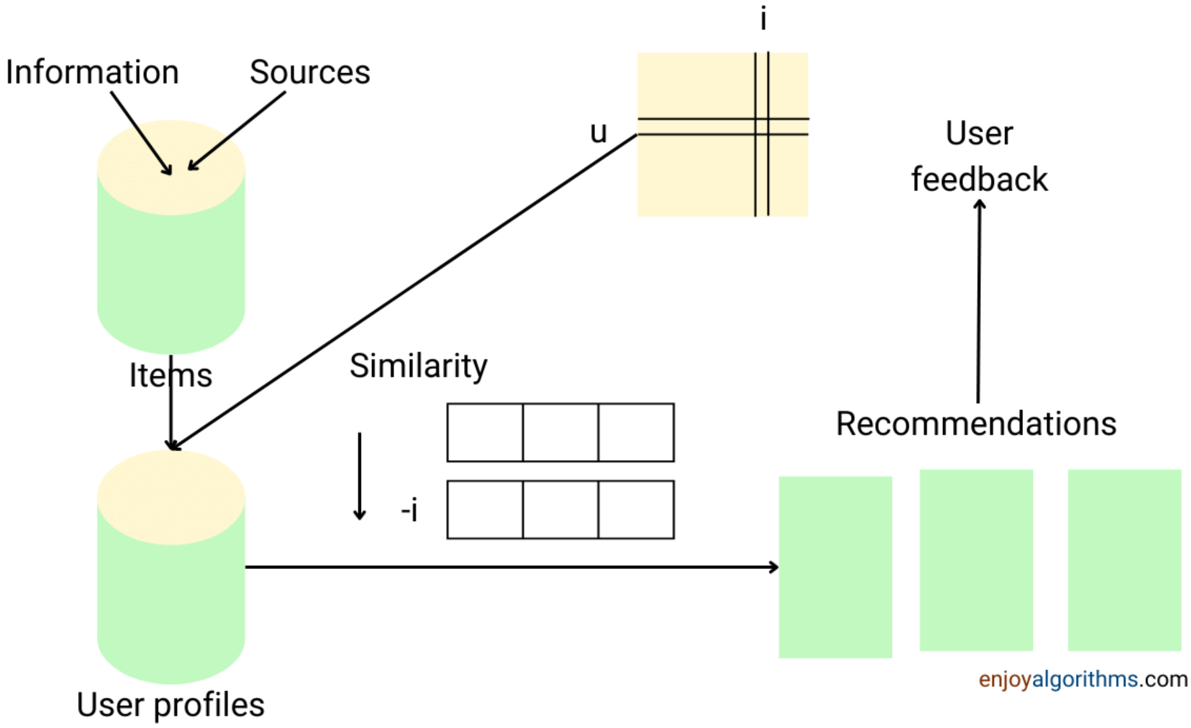
How to build a User profile?

Suppose a user has rated items with profiles **i1,i2,i3…..in**. The user profile is simply the normal or weighted average of the rated item profiles. Various aggregation methods are possible.

Once we have the User and Item profile ready, we can use them to make predictions using **Cosine Similarity.** Suppose we have User profile **x** and Item profile **i,** we have to estimate the maximum value of U(x, i), which is given by:

$$U(x, i) = cos(\theta) = \frac{dot\_product(x, i)}{|x||i|}$$

|a| represents the magnitude of vector a. Greater the value of cos(θ), the smaller the θ, and, hence, the closer the x and i. In this manner, we recommend items to users using Content-Based Recommender Systems. The figure below describes the complete overview of the content-based recommender system works.

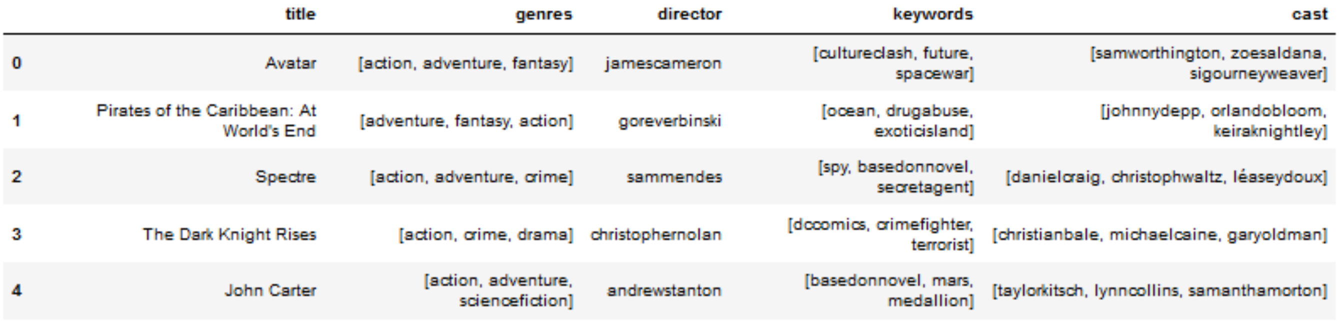


Implementation Steps

Now we must have got a basic understanding of how a recommender system works. Here in this section, we'll look into the actual dataset and working of the recommender system on some real-world examples of recommending movies. Here we are using TMDB 5000 movie dataset to build a Content-Based Recommender System.

Step 1: Dataset Description

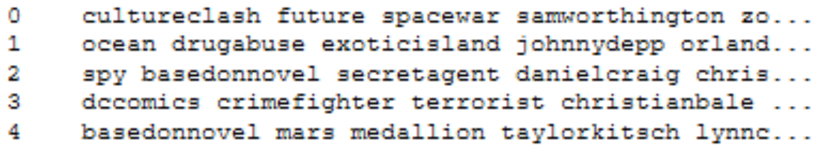
In this system, we use the movies' contents, such as title, genre, cast, directors, etc., as the features to recommend similar movies. The dataset looks like as shown below.



	title	genres	director	keywords	cast
0	Avatar	[action, adventure, fantasy]	jamescameron	[cultureclash, future, spacewar]	[samworthington, zoesaldana, sigourneyweaver]
1	Pirates of the Caribbean: At World's End	[adventure, fantasy, action]	goreverbinski	[ocean, drugabuse, exoticisland]	[johnnydepp, orlandobloom, keiraknightley]
2	Spectre	[action, adventure, crime]	sammendes	[spy, basedonnovel, secretagent]	[danielcraig, christophwaltz, léaseydoux]
3	The Dark Knight Rises	[action, crime, drama]	christophernolan	[dccomics, crimefighter, terrorist]	[christianbale, michaelcaine, garyoldman]
4	John Carter	[action, adventure, sciencefiction]	andrewstanton	[basedonnovel, mars, medallion]	[taylorkitsch, lynncollins, samanthamorton]

Step 2: Text Pre-processing

We need to pre-process the data in this step, i.e., converting to lower cases, removing punctuations, stemming, tokenization, etc. We can use the porter stemmer algorithm for this task. After performing data-cleaning, our dataset would look like this:



```
0    cultureclash future spacewar samworthington zo...
1    ocean drugabuse exoticisland johnnydepp orland...
2    spy basedonnovel secretagent danielcraig chris...
3    dccomics crimefighter terrorist christianbale ...
4    basedonnovel mars medallion taylorkitsch lynnc...
```

Step 3: Generate Recommendations using TF-IDF and Cosine Similarity

After all the text-processing, now it’s time to generate word vectors of each feature. For this, we can use TF-IDF (Term Frequency-Inverse Document Frequency) to generate a matrix in which each column represents a word in the overview vocabulary. Each row represents a movie that will be used for calculating the similarity scores.

Here, Term Frequency is the relative frequency that a given word is present in a document and Inverse Document Frequency gives the relative count of documents containing a particular word. Details about TF-IDF can be found in this article.

```
TF=(given word/total words)
```

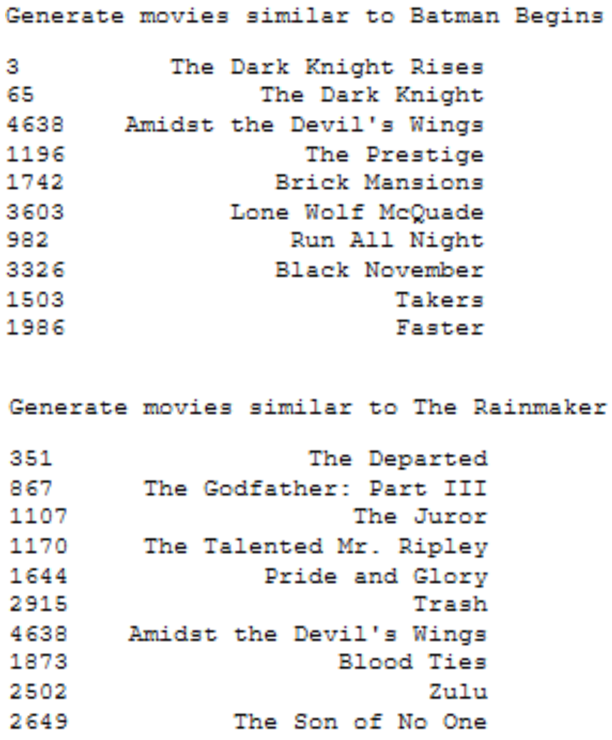
```
IDF=log(number of documents/documents with given word)

Each word's overall importance to the documents in which
they appear is equal to TF * IDF.
```

Note: We can also use **CountVectorizer()** instead of **TF-IDF** as TF-IDF down-weight an actor if he/she has acted in more movies.

Once we have this importance matrix, we can use it to generate similarity measures. Here we are using **Cosine Similarity** for calculating the measures.

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum\limits_{i=1}^{n} A_i B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2}\sqrt{\sum\limits_{i=1}^{n} B_i^2}},$$

Hence after calculating the similarity measures, we can easily recommend movies. Here are examples of how our recommender system is recommending movies to the user.



```
Generate movies similar to Batman Begins

3          The Dark Knight Rises
65               The Dark Knight
4638    Amidst the Devil's Wings
1196                The Prestige
1742              Brick Mansions
3603           Lone Wolf McQuade
982                Run All Night
3326              Black November
1503                      Takers
1986                      Faster

Generate movies similar to The Rainmaker

351                 The Departed
867      The Godfather: Part III
1107                   The Juror
1170     The Talented Mr. Ripley
1644             Pride and Glory
2915                       Trash
4638    Amidst the Devil's Wings
1873                  Blood Ties
2502                        Zulu
2649           The Son of No One
```

Performance Evaluation of our Recommender System

There are two ways to evaluate a recommender system. They are **Offline way** and **Online way**.

Offline Way: Offline way measures the system's performance by splitting the data into training and validation sets. Offline Evaluators are of two types: Implicit and Explicit. Some metrics for offline evaluation are:

- RMSE
- F1 Score

Online Way: It is employed to track down the recommendations and validate those through customer interaction after deploying the model. Metrics for online evaluation are:

- Customer Lifetime Value
- Click-Through Rate
- Return On Investment

Many other ways are used to build Recommender Systems. Nowadays, deep learning-enabled models such as auto-encoder are also used to develop recommender systems. Moreover, Reinforcement based recommender systems are also deployed by major tech giants to give a state-of-the-art performance.

How are companies using Recommender System?

Nowadays, almost every company focuses on better user personalization and longer retention time. Recommender Systems offers a better consumer experience and also boosts the company's revenue. Let’s see how companies relied on this:

YouTube

We all must have used YouTube, but have we ever noticed that our home feed begins to show similar content if we search or watch any specific kind of video. If we haven’t noticed that yet, please do check, we will be surprised how using this method; they are trying to increase their users' retention time. **YouTube** heavily personalizes **recommendations** based on a user's viewing history and hence offers a better user experience.

Social Media (Instagram, TikTok)

If we are familiar with Instagram, TikTok, or the new Reel feature of Instagram, we must have noticed that the search section will show similar content if we like or spend some time over a certain kind of content. Tracking the user's interaction tries to predict their behavior, which might be very useful for advertising any other third-party products.

Amazon

We must have interacted with Amazon, right! Whenever we buy anything or search for any product, we must have noticed some sections suggesting like "**People who bought this also bought….**" or other sections offering some lucrative offers to purchase more add-on items with the searched product. All these are possible because of Recommender Systems. Hence such systems proved a boon to both the consumers and the company. Consumers now have more and more options to choose from, and Companies are generating more revenue by offering more and more cheesy offers :) Yummy!

Netflix

Recently Netflix announced two days of free access to Netflix content. We must have tried that, right! But have we ever thought about what their underlying intuition behind this is? They are trying to figure out how the unreachable customers are searching from the large content pool and what kind of content is most demanding. Hence by observing such behavior, they will create new content and recommendations based on our behavior. Doesn't it looks interesting, right! So yeah, in a nutshell, Netflix heavily relies on Recommendation systems for creating new content and increasing its revenue. We can think of the importance of the recommender system to Netflix by the amount of prize money they offered to a team that beat Netflix's recommender system by 10%. So let's dive into the details of "The Netflix Challenge."

Case Study: The Netflix Prize

"The Netflix Prize" is the competition launched by Netflix between 2003 to 2006. This challenge motivated the researchers to develop a novel recommender system that could beat the Netflix system by more than 10%. Netflix sponsored the competition and offered a grand prize of US $1,000,000 to the team who could surpass Netflix's existing recommender system with 10% more accuracy. For this competition, a dataset of 100 million movies was offered to the teams to work on. On 21 September 2009, BellKor's Pragmatic Chaos team was awarded the grand prize of US $1,000,000.

Netflix's challenge boomed the research focused on recommender systems, and as a result, many companies came up with their own recommender systems to focus more on customer prospects.

Possible Interview Questions

- What are the steps that you took to pre-process your data?
- Why is a hybrid approach more beneficial in a recommendation system?
- How will you evaluate your recommendation system?
- What is the Porter stemmer algorithm?
- How is cosine similarity a good measure to know the similarity between two choices?

Conclusion

Recommender Systems have a very significant role in current industries. Using recommender systems, companies are now focusing more and more on users' behavior for better user personalization and longer retention time. Recommender Systems are evolving day by day based on user interactions and their behaviors. They are the backbone of the E-Commerce industry.

This article has just given the use cases of the Recommender Systems and how it plays a key role in various real-world scenarios and a better-personalized experience. We hope you all liked it. Please do suggest in the comment your views on this blog. Till then,

If you have any ideas/queries/doubts/feedback, please share in the message below or write us at **contact@enjoyalgorithms.com**. Enjoy learning, Enjoy algorithms!