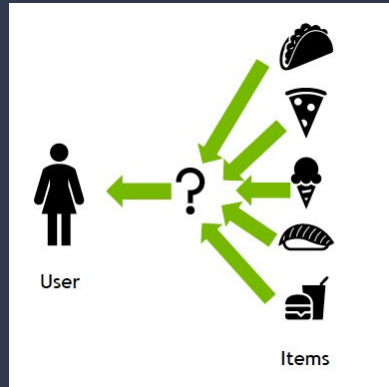


# Recommendation Systems

SALAZAR VILLEGAS LUIS SEBASTIAN  
JONATAN MORALES MENDOZA

A dark blue diagonal gradient bar that starts from the bottom left corner and extends towards the top right corner, covering the lower half of the slide.

# Introducción

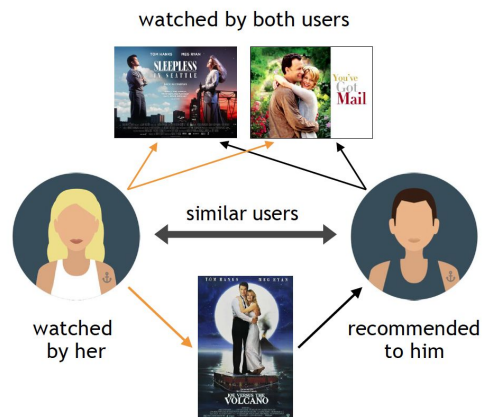


Un sistema de recomendación es un algoritmo de inteligencia artificial o IA, generalmente asociado con el aprendizaje automático, que utiliza Big Data para sugerir o recomendar productos adicionales a los consumidores.

Estos pueden basarse en varios criterios, incluyendo compras pasadas, historial de búsqueda, información demográfica y otros factores.

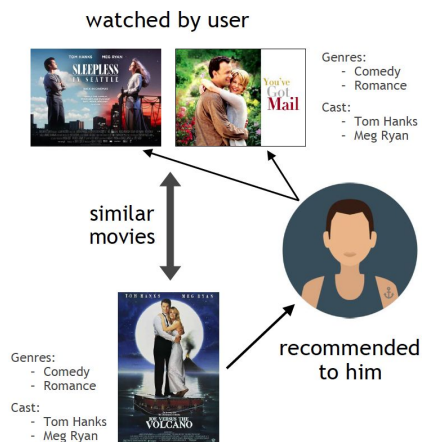
# Tipos de sistemas de recomendaciones

## Collaborative Filtering



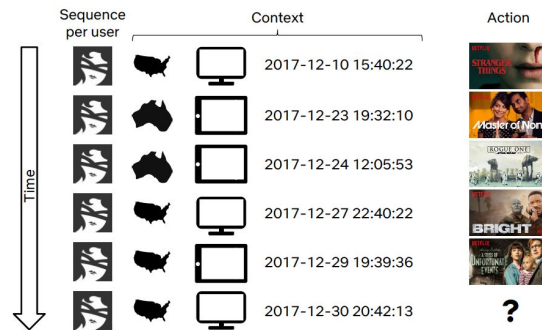
Filtrado colaborativo

## Content-based Filtering



Filtrado de contenido

## Contextual sequence data



Filtrado de contexto.

# Beneficios

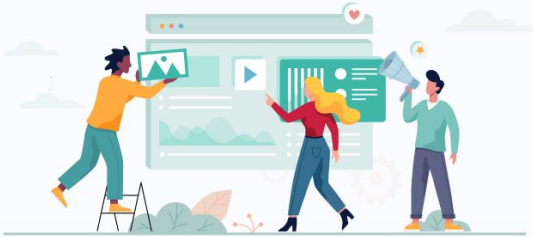
Los sistemas de recomendación son un componente crítico que impulsa experiencias de usuario personalizadas, un compromiso más profundo con los clientes y poderosas herramientas de soporte de decisiones en el comercio



- Mejora de la retención
- Aumento de las ventas
- Ayudar a formar hábitos y tendencias de los clientes
- Acelerar el ritmo de trabajo
- Impulsar el valor del carrito



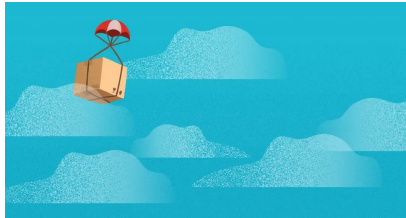
# Desafios usuales?



Arranque frío de usuario se refiere al hecho de cuando los nuevos usuarios entran en un sitio web o app por primera vez.



La recopilación y el uso de datos personales para hacer recomendaciones pueden plantear preocupaciones éticas y de privacidad.



Ingreso de nuevos productos

# Empresas que usan



# Casos de uso más recurrentes



- E-Commerce & Retail



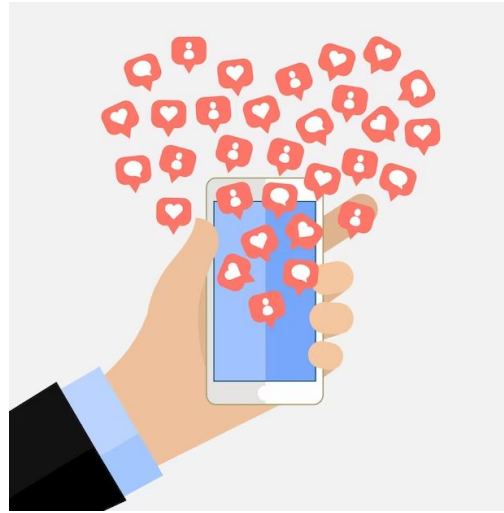
- Entidades Bancarias



- Media & Entertainment

# Tipos de sistemas de recomendación

- POPULARIDAD
- BASADO EN CONTENIDO
- COLABORATIVO

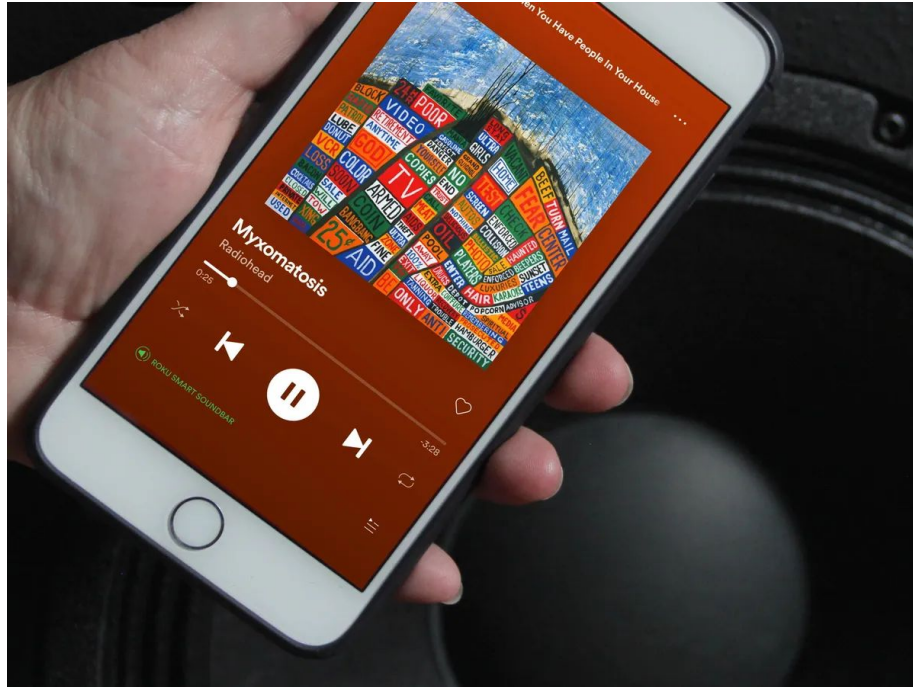




# Popularidad

	1. Stray 6 days   Cats Top 250 #49 \$29.99  	8.61 98% 42,655 votes
	2. God of War Jan 2022   Action Top 250 #75 \$49.99  	8.56 97% 53,101 votes
	3. PowerWash Simulator 11 days   Simulation Top 250 #81 \$24.99  	8.55 98% 23,539 votes
	4. Teardown Apr 2022   Destruction Top 250 #143 \$19.99  	8.47 96% 45,845 votes
	5. Dorfromantik Apr 2022  City Builder Top 250 #144 \$13.99  	8.47 97% 16,424 votes
	6. Neon White Jun 2022   Action Top 250 #164 \$24.99  	8.45 98% 5,187 votes
	7. The Looker Jun 2022   Puzzle Top 250 #198 FREE  	8.43 98% 6,096 votes

# Basado en contenido



Se basan en los atributos de los productos.

La idea básica es que a cada usuario le va gustar un elemento similar a lo que le ha gustado en el pasado.

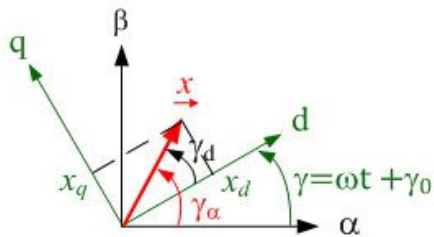
# Basado en contenido

## Content-Based

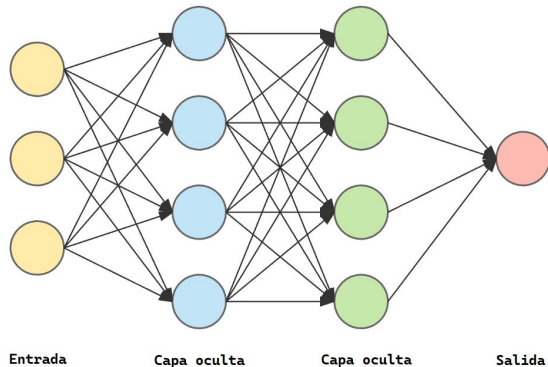


# Basado en contenido

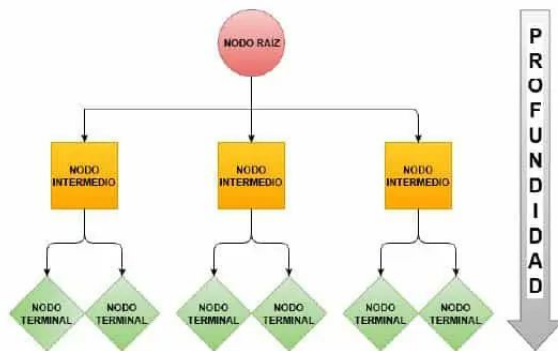
## VECTORES ESPACIALES



## REDES NEURONALES



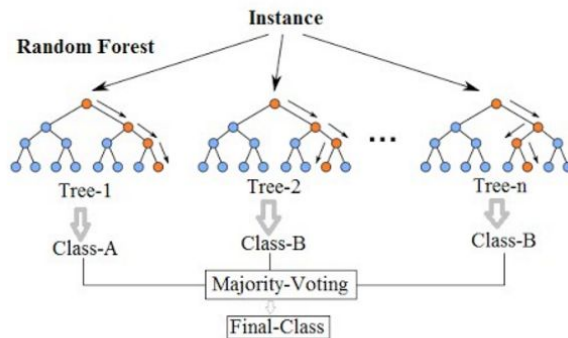
## ÁRBOL DE DECISIÓN



## RANDOM FOREST

## ÁRBOLES DE DECISIÓN

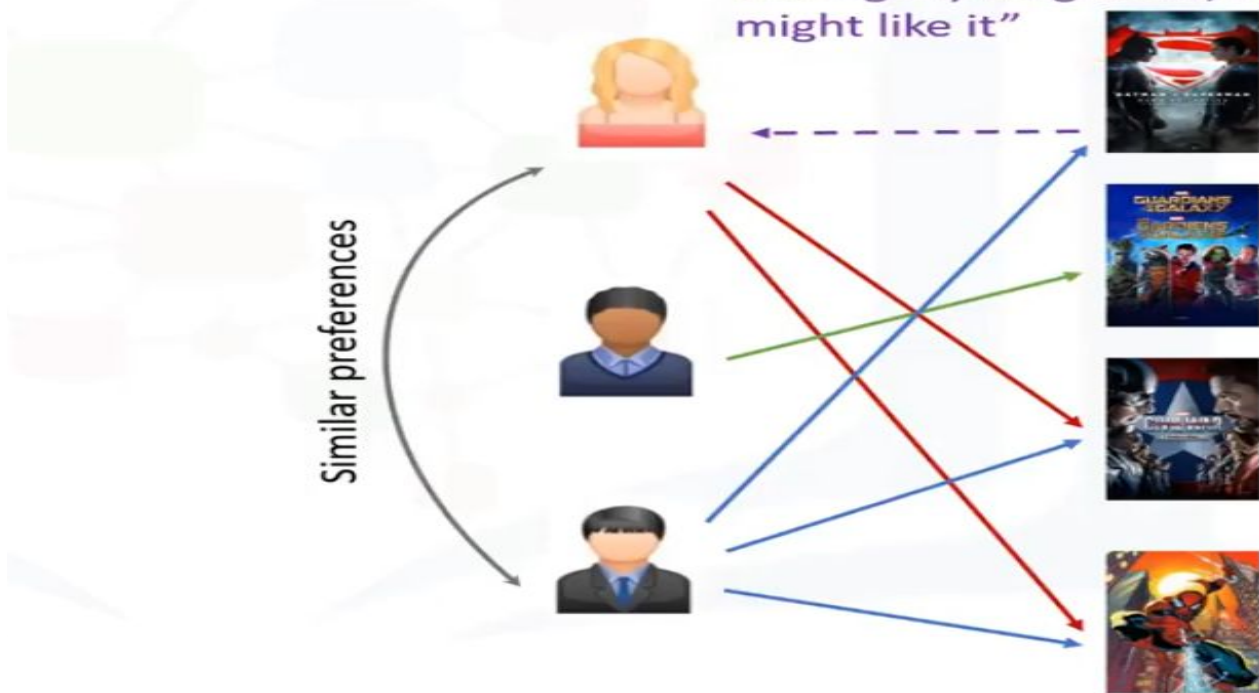
### Random Forest Simplified



# Colaborativo

## Collaborative Filtering

"Tell me what's popular among my neighbors, I also might like it"

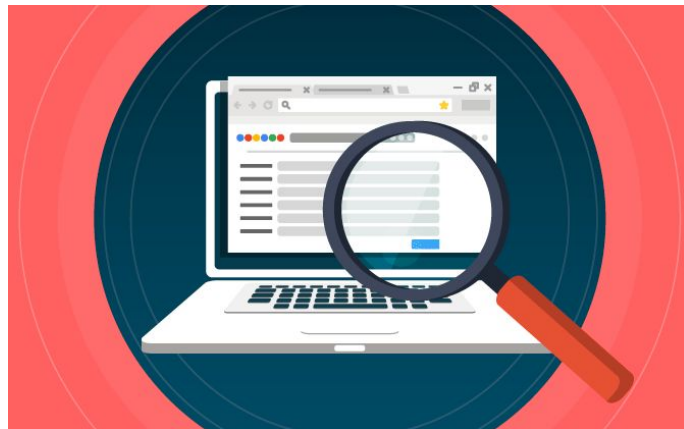


# Colaborativo

## EXPLÍCITO



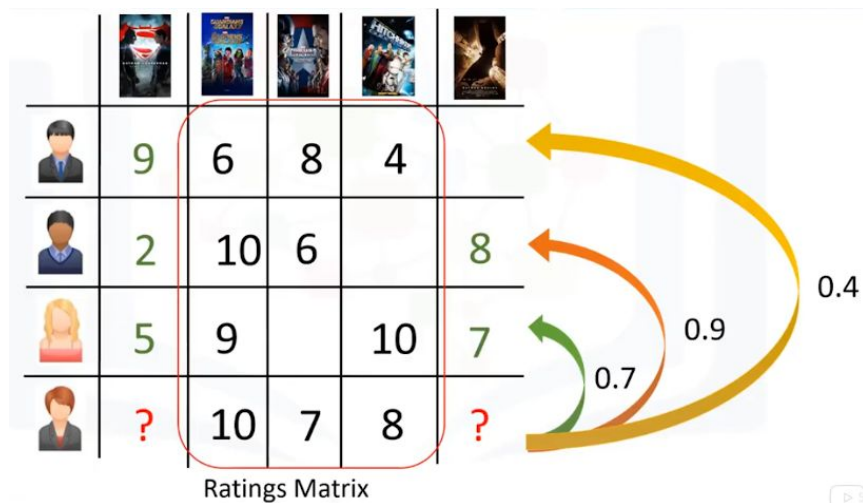
## IMPLÍCITO





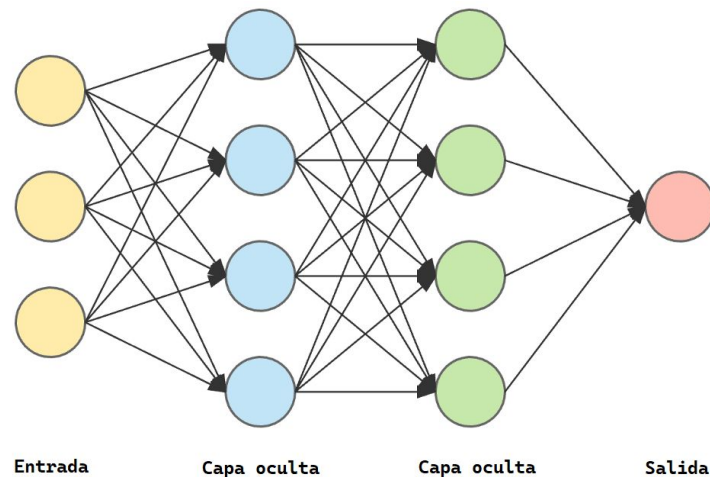
# Colaborativo

## BASADO EN MEMORIA



## BASADO EN MODELOS

Matrices de factorización, grafos, modelos bayesianos, redes neuronales



# Implementación

```
df = pd.read_csv('titles.csv')  
df.head()
```

Python														
id	title	type	description	release_year	age_certification	runtime	genres	production_countries	seasons	imdb_id	imdb_score	imdb_votes	tmdb_popularity	tmdb_score
ts21930	Lupin the Third	SHOW	Arsene Lupin III is the grandson of the master...	1971	TV-14	23	['sci-fi', 'animation', 'action', 'comedy', 'cr...	['JP']	6.0	tt0159175	7.9	2116.0	45.829	8.0
ts25177	Mobile Suit Gundam	SHOW	Mobile Suit Gundam is a televised anime series...	1979	TV-Y7	25	['sci-fi', 'war', 'animation', 'action', 'drama...	['JP']	1.0	tt0159172	7.9	2281.0	41.250	8.1

Obtención de datos, el dataset se obtuvo de los datos de Crunchyroll

Análisis de la data

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 1081 entries, 0 to 1080  
Data columns (total 15 columns):  
#   Column                Non-Null Count  Dtype    
--  --  
0   id                     1081 non-null  object   
1   title                  1081 non-null  object   
2   type                   1081 non-null  object   
3   description            1050 non-null  object   
4   release_year          1081 non-null  int64    
5   age_certification      750 non-null   object   
6   runtime                1081 non-null  int64    
7   genres                 1081 non-null  object   
8   production_countries   1081 non-null  object   
9   seasons                1080 non-null  float64   
10  imdb_id                927 non-null   object   
11  imdb_score             907 non-null   float64   
12  imdb_votes             907 non-null   float64   
13  tmdb_popularity         1068 non-null  float64   
14  tmdb_score             1006 non-null  float64   
dtypes: float64(5), int64(2), object(8)
```



# Obtención de las recomendaciones

```
def get_recommendations(title, cosine_sim=cosine_sim):  
    idx = indices[title]  
    sim_scores = list(enumerate(cosine_sim[idx]))  
    sim_scores = sorted(sim_scores, key=lambda x: x[1], reverse=True)  
    sim_scores = sim_scores[1:11]  
    anime_indices = [i[0] for i in sim_scores]  
    recommended_animes = df['title'].iloc[anime_indices].tolist()  
    return recommended_animes
```