Building a News Recommender System

IDC 410 Group Project

Department of Physical Sciences
IISER Mohali

April 24, 2022
Team: HISP

Authors-
Himanshu Jain (MS19026)
Ira Sharma (MS19037)
Shreya Pipraiya (MS19107)

Instructor- Dr. Sarabjot Singh Anand

1 Web Scraping using RSS Feed

RSS stands for Really Simple Syndication. We extracted the RSS feed from 6 different news websites namely BBC, CNN, The Hindu. Economic Times, India today and NDTV; to scrape News Headlines, News Summary, Datetime, URLs. We wrote the code for Web Scraping using the RSS Feed.

	A	B	C	D	E	F	G	H	I	J	K	L
1		Category	Headlines	Summary	DateTime	Url						
2	1	Business	Hauliers want priority for perishable g	Drivers say deliveries of produce	Sun, 10 Apr 2022 0	https://www.bbc.co.uk/news/business-61053402?at_medium=RSS&at_ca						
3	2	Business	AI-assisted digital out of home media	Firm eyes listing through merger	Sat, 09 Apr 2022 2	https://www.thehindu.com/business/ai-assisted-digital-out-of-home-me						
4	3	Business	Elon Musk Proposes Dogecoin For Twi	Elon Musk, Twitter Inc's biggest	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/dogecoin-for-twitters-blue-premium-sub						
5	4	Business	New Russia Sanctions Include Crypto \	The European Union on Friday ta	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/new-russia-sanctions-include-crypto-wa						
6	5	Business	No Plans To Regulate Metaverse, Web	The government has made it cle	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/no-plans-to-regulate-metaverse-web-3-						
7	6	Business	Reliance Capital's Lenders And Admin	Differences have emerged betw	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/rbi-appointed-admin-and-lenders-disagr						
8	7	Business	FX Markets Tell Central Banks' "Go Ha	The trades in foreign exchange n	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/fx-markets-tell-central-banks-go-hard-o						
9	8	Business	Jump In Gold Imports By A Third To $4	India's gold imports, which have	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/gold-imports-surge-to-usd-46-14-bn-in-2						
10	9	Business	IT Refund: Here's How To Know The St	Income Tax Refund: The income	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/how-to-know-your-income-tax-refund-s						
11	10	Business	Earnings And Macro Data Will Drive Hc	Quarterly earnings from IT majo	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/q4-earnings-macro-data-global-trends-t						
12	11	Business	NDA Government's Development Expe	Finance Minister Nirmala Sithara	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/nda-governments-development-expendi						
13	12	Business	RBI Revises Inflation Forecasts. Here's	The CEO of Kotak Mahindra Ban	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/as-rbi-revises-inflation-forecasts-uday-k						
14	13	Business	Government Working On FAQs On The	The government is working on a	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/faq-on-taxation-of-crypto-virtual-digital						
15	14	Business	Forget Ashneer Grover, BharatPe Post	Putting behind the controversy a	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/bharatpe-puts-behind-grover-episode-p						
16	15	Business	After 6 Months, Foreign Portfolio Inve	FPI Investment: After a six-mont	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/fpis-turn-net-buyers-in-april-so-far-inves						
17	16	Business	Over Rs 1 Lakh Crore Market Cap Loss	The combined market valuation	Sun, 10 Apr 2022 1	https://www.ndtv.com/business/four-of-top-10-cos-lose-over-rs-1-lakh-						

2 Text Preprocessing

The scraped data was then preprocessed including:

- Making all the characters lowercase
- Clearing all the numerical and special characters
- Stemming
- Lemmatization
- Removing all the NAN value entries

3 Vectorizing the news dataset

From our news stories dataset, we created our vocabulary set and then used Countvectorizer for vectorizing the different news in the dataset. We also tried the TF-IDF method and found this one better than the Countvectorizer as it gave a larger cosine value for the recommended news articles.

4 Content-based filtering

In content based filtering, first some random news from each category will be shown to the user. And after that when user clicks and read the news the time for which the user reads the news gets recorded in the csv file of that user. And then 2-8 news out of 10 get recommended to that user in the next session.
The main criteria to find the similarity between the vectors used is **"Cosine similarity"**

5 Cosine Similarity

By the inner product we have:

$$\vec{a}.\vec{b} = |\vec{a}||\vec{b}| \cos\theta$$
$$|\cos\theta| = \frac{|\vec{a}.\vec{b}|}{|\vec{a}||\vec{b}|}$$

So the aim is to find the row vectors from the ranking matrix with which the new vector has the largest $\cos\theta$ value and then the same news feed will be recommended to that user.

6 Simulation of Ranking Matrix

6.1 Finding the meantime to read the particular news

Using **readtime** module in python, we have found the time to read that particular news assuming the speed of 200 wpm. Using this as mean, we have selected values from the normal distribution as the time which the reader will take to read that news.

6.2 Generating the fake data

We randomly generated news articles of 10 out of 12 different categories for displaying it to 100 fake users, from which the user randomly clicked on 2 to 7 articles and also used a normal distribution of the time spent whose mean was the average reading time of that particular news article. After that we applied Content-Based Recommendation on the clicked news and generated the User data containing the UserID, ArticleID, Category, Time Spent.

	User ID	Article ID	Category	Time Spent
1	0	806	Technology	3.075140834
2	0	112	Entertainment	7.736988716
3	0	259	India	1.949288416
4	0	665	Sports	9.773501955
5	0	113	Entertainment	9.425633729
6	0	345	Lifestyle	17.11667849
7	0	379	Politics	4.556894943
8	0	523	Sports	4.030412574
9	0	436	Politics	5.415538594
10	1	470	Science	5.887012553
11	1	351	Market	10.9803104
12	1	800	Technology	2.023171973
13	1	122	Entertainment	2.877910675
14	1	456	Science	7.325650873
15	1	438	Politics	2.273187124
16	1	607	Sports	1.347502653
17	2	789	Technology	4.603195634
18	2	109	Entertainment	3.468692164
19	2	96	Education & Family	4.640578287
20	2	488	Science	3.122939673
21	2	241	Health	3.554339385
22	2	110	Entertainment	3.286668693
23	3	870	World	3.239313768
24	3	270	India	15.29127881
25	3	345	Lifestyle	4.462320033
26	3	453	Science	0.423646726
27	3	59	Business	5.820605091

6.3 Generating the Ranking Matrix

Then we ranked each category for each of the 100 users based on the ratio of the sum of time spent on the articles of a particular category divided by the total sum of the average reading time of the articles of a particular category displayed to each of the 100 users.
For i^{th} user,

$$\text{the rank of } j^{th} \text{ category} = \frac{\text{Time the } i^{th} \text{ user spent on reading the news of } j^{th} \text{ category}}{\text{Total mean time of the news of the } j^{th} \text{ category}}$$

	Business	Education & Family	Entertainment	Health	India	Lifestyle	Market	Politics	Science	Sports	Technology	World
0	0	0	1.606585975	0	0.302308998	1.05532565	0	0.908567195	0	0.908949596	0.379522071	0
1	0	0	0.242670527	0	0	0	1.190664758	0.095085909	0.732544636	0.397415254	0.212637189	0
2	0	0.60788293	0.701636981	0.234868241	0	0	0	0	0.394775789	0	0.259004218	0
3	1.041797507	0	0	0	1.580221717	0.759889448	0.503374924	0.09337151	0.075534303	0	0	0.363806393
4	0	0	0	0	0	0.647785657	0	0	0	1.175630732	0	0.113528241
5	0	0	0	0	0.504919943	0.158835779	0	0	0.286879993	0	0.564469494	0.356832644
6	0	0.776396678	0.78188887	0	0.267805707	0	0	0	0.129978078	0	0.288317639	0.188037921
7	0	0	0	0.94261313	0.576622199	0.504822034	0	2.371810472	0.475703928	0.656629345	1.00646957	0
8	0.606192741	0	1.128915827	1.269023811	0	0.907856421	1.218717119	0.249758413	0	0	0	0
9	0.389647957	0	0.863517499	0	0.158353165	0	0.527507348	0.476185188	0	1.208712038	0	0.489783045
10	0	0	0	0	0.767928777	0.307423339	0.405945367	0	0.319108682	0.665232209	0	0.870365953
11	0	0	0.705501615	0	2.627370797	0.89387855	0	0	0	0	0.014600818	0.337098821
12	0	1.676012926	1.012575031	0	0.315776498	0	0	0.354930229	0.019395217	0	0.724582372	1.675447738
13	0	0	0.653415665	0	0.257279912	0.303960324	0	0	0.582449093	0.351967296	0	0
14	0.160120606	0.712314081	0.206024828	0	0	0.219857302	0.25246956	0	0.867137881	0.371845186	0.780349708	0
15	0.102265756	0.503955268	0	0.331988221	0	0	0.29967727	0	0	0.153046962	0	0
16	0	0	0	0	0.086736575	0	0.96157096	0.334737494	0	0.190787002	0	0
17	0.486903195	0.468912171	1.803929868	1.466868458	0	0.956483732	0.36273932	0.455871944	0	0.196423013	2.499058604	0
18	0	0.277908252	1.020381282	1.082675334	0	0.871788213	0.593237655	0.252856559	0	0.93924053	0.072782442	0
19	0	0	0	0.409168103	0.421292784	0	0	0	0	0.592037349	0	0
20	0.116075219	0	0	0	0	0	0	0.004781085	0.177161155	1.534928552	0.014695552	0
21	1.076416121	0	0	0	0	0	0	0	0	0	0	0.524421575
22	0	0	0	0.392229303	0	0	0.38029729	0.619179154	0.918847254	0.269677308	0.174152502	0
23	0.585194719	0	1.088603625	0.182107064	0	0	0	0	0	0.471108344	0.244171486	0.625242906
24	0.525746562	0.36875784	0	0	0.319330897	1.462134876	0.138672756	0	1.02988138	0	0	1.082332284
25	0	0	2.476117711	0.438126344	0.76737143	0.251700377	0.668980729	0	0	0.309797191	0	0

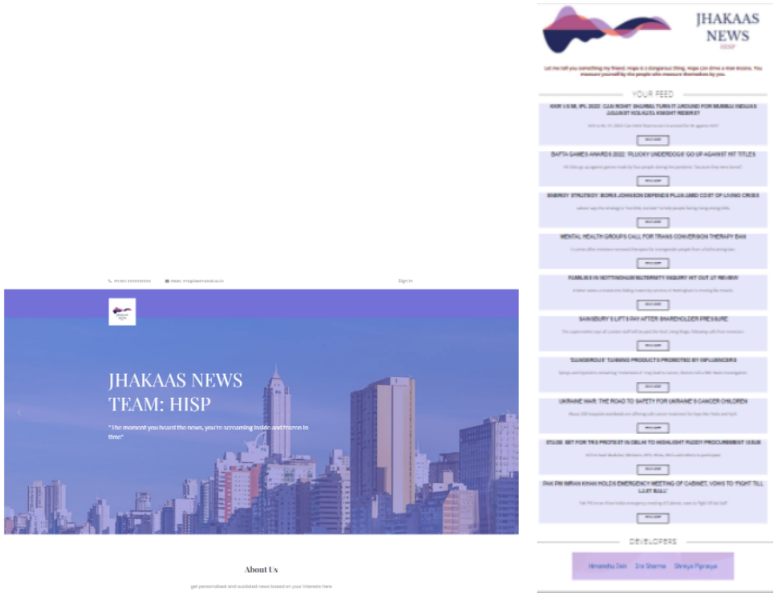
7 Collaborative Filtering

On this Category Ranking Matrix, we have used Cosine Similarity to find how close the two vectors are. If the user-based vector is closer to one of the existing vectors then θ will be smaller and thus $\cos\theta$ will be larger (near to 1).
The aim is to find the row vectors from the ranking matrix with which the new vector has the largest $\cos\theta$ value and then the same news feed will be recommended to that user.

8 Web App Using Flask

8.1 Implementation of timer to generate clickstream data

On clicking the news, a JavaScript function has been called which creates an **"Xml Http request"** which calls the **route function** of flask. When the news has been clicked for the first time, it initializes to record the time. When the next news has been clicked, then the time of first news has been recorded completely and the time to record the second news gets initialized. When the user closes the website or click the back button or refresh the page then final time of the last news gets recorded.
This is done using the unloading event in the flask using the route function, **Navigator.sendBeacon()**. Both these route functions return html 204 response.



9 Thing that we have planned but could not implement

- We had planned to use Latent Semantic Analysis for vectorizing the documents but due to time constraints we are not able to implement it.
- We had planned to implement better security checks in the website but could not implement that completely.

10 GitHub link for the code

`https://github.com/jaihimanshu/IDC410-News-Recommender`