Analytical Study on Algorithms for Content Based Mobile Phone Recommendation System

Abstract - Recommendation of anything helps in filtering out unwanted and irrelevant products from the entire work set which are not of any use or do not add any importance or value to the current task set to be accomplished. Mobile phone recommendation systems could be a probable revolution in the near future, as the telecommunication and mobile phone production industries are increasing exponentially in the market. Such systems would extract description of similar mobile phones which are most correlated with the mobile phone of user interest. The proposed system carries out this task taking the dataset which comprises of mobile phones and its features extracted from the E-commerce website '*Flipkart'*. The dataset comprises of 6917 records in total each one corresponding to a unique mobile phone in the website. The proposed model finds the similarities and does the predictions based on the input features in the dataset.

Keywords - Natural Language Processing, Tokenizing, Cosine Similarity, Corpus, *LSI* model, Similarity Index.

1 Introduction

These days every customer is confronted with various choices. For example, if a customer searching for a book to examine with no particular thought of what he needs, there is a wide extent of chances for how his interest may work out. He may consume lot of time examining around the internet content and trawling through. In such cases, he tends to take suggestions/recommendations from other people. This has led to the idea of developing recommendation systems instead of getting it from other people. The force of recommendation engines is being benefited by most associations these days. These systems wipe out data over-burden issues i.e., the normal state of a web customer, of having an exorbitant measure of information to choose a decision or remain instructed about a subject. A recommendation system is a kind of data sifting framework which endeavours to anticipate the inclinations of a client, and make recommendations based on these inclinations. There are a wide variety of applications for these systems. They have become progressively notable in the course of the most recent years and are presently used in most online stages are used. Frequently, these systems can gather data about a client's decisions and can utilize this data to improve their recommendations later on.

One of the two primary methodologies that are broadly utilized in recommendation systems is the content-based filtering, where it is attempted to profile the clients' interests utilizing data gathered, and suggest things dependent on that profile. Content based recommendation is the approach that has been applied in this work and is experimented for mobile phone recommendation.

The rest of the paper is organized as follows – the Section 2 to present the works related to this work, the Section 3 to discuss the design of experiments used for this study, the Section 4 details the architecture of the proposed system, the Section 5 to discuss the results and the Section 6 to conclude the work.

2 Related Works

The proposed architecture involves computing the Similarity index using both Count Vectorizer and *TF-IDF* to recommend mobile phones to the user. As there are no research works reported on content-based recommendation of mobile phones, the research works related to recommending products from e-commerce sites like *Flipkart* and *Amazon* are discussed here. Also, the research works on content-based filtering that have been done over the past and where some prominent exploration is needed are highlighted.

Techniques involved in content-based filtering (*Count Vectorizer* and *TF-ID*) for calculating the similarity index were presented in [1]. The author of [2] suggested combining recommender systems innovations with mobile phone search can become beneficial for mobile users, as the mobile phones are becoming an essential device for accessing the information. According to [3], mobile recommenders form a new exploration field that has not been efficiently studied and further analysis is needed to concentrate on the implementation of recommendation systems in mobile phones. The authors of [4] explained the recommendation process's multiple phases, namely Data collection, Learning, and Recommendation. According to [5] use of recommender systems is crucial for the service providers as well as for the users. The reasons that the providers use such systems are - increase in sales, employ a decent variety of items, increase in client satisfaction, and increase in faithfulness. Using specifications of an item to recommend additional items with similar properties by the model in content-based filtering was proposed in [6]. Various methods involved in pre-processing the dataset like tokenization, stop word removal, stemming, and Dictionary was explained in [7]. The authors of [8] proposed an approach to build a recommender system that can be adapted to customizable products such as computers and home theatre system. The authors expanded the means of recommendation systems, so that they recommend the products alongside their customizable parts to customers and provide them a more significant alternative in looking. Various challenges of recommender systems like Cold Start, Sparsity, Scalability were explained in [9]. The authors of [10] described how the implementation of recommender system has enlarged on the internet, that demonstrated its use in various areas like recommending movies, applications, gadgets, songs, websites and wherever the user will provide ratings to an item of his choice. According to [11], for improving the retrieval method of Latent Semantic Indexing (*LSI*) stop words are enforced to spot normally used words (which can be neglected before or when the processing of a natural language) and removing them plays a crucial role. The authors of [12] explained why it is vital to reduce the dimension of the vectors used to represent the search space for retrieving textual information. Representation of vectors in high dimension will cause scattered vectors, which will affect the efficiency of retrieval of information.

The proposed system used *LSI* in order to reduce the dimension of the term space into a lower-dimensional *LSI* space, which is more compact. Thus, it could identify similarities that go beyond term similarity [13]. Considering, the massive gap between the users and service providers of mobile phone recommendation systems, this work focuses on exploring various algorithms for mobile phone recommendation systems and thus filling the gap between users and service providers.

3 Design of Experiments

With late progressions in innovation, a mobile phone can do all the tasks a personal computer is prepared to do and furthermore the greater part of the web traffic is originating from mobile phone. It is a clear indication that the interest and use of mobile phones are expanding at a tremendous measure. Prior, the quantity of mobile phone user in a family was restricted to few but these days every single individual have a mobile phone. Because of which demand of mobile phones is going up and the mobile phone industries are enhancing the specification and configuration of them to pull in an ever-increasing number of clients. Hence, mobile phone variations are additionally expanding at an exponential rate. This makes a great deal of disarray for the purchasers and they may wind up purchasing an unseemly item. To beat these issues of data over-burden a recommender system is suggested. Because of quick development in the varieties of mobile phone dependent on their structure and functionalities, it is getting hard for the client to choose the best mobile phone as per their prerequisites and inclinations. The primary issue in mobile phone determination is the examination between the various things, which is very complicated and physically it is impossible. So, an efficient methodology is necessary for the complicated examination between the things and to recommend the client most proper thing as per their inclinations. Therefore, Recommender System helps in doing an examination and assessment over enormous assortments of mobile phone functionalities, highlights, structure, and brand to give a productive recommendation to the user.

3.1 Dataset

The proposed system used a dataset which consists of over 6900 mobile phone of various brands with their specifications, descriptions, user ratings on a 5-point scale and reviews. This information was scraped from the *Flipkart* website for the corresponding mobiles as *Flipkart* is one of the most popular sites and the data here is more organized when compared to other websites.

3.2 Content Based Implementation

Data Scraping - To obtain data consisting of mobile phones and their Specifications, the *Flipkart* website page content were accessed using the *Beautiful Soup* python library and *Requests* module.

BeautifulSoup is a tool for parsing *HTML* code and grabbing exactly the information you need. *Requests* is a module in python which can be used to send *HTTP requests***.** Once the structure of the website was identified, a script to automatically extract the relevant information about a mobile phone was developed. The details extracted are

Product Name. *Samsung Galaxy A50 (Blue, 64 GB)* .

Ram & Rom-*6 GB RAM | 64 GB ROM | Expandable Upto 512GB*

Price – *18999,* **Rating** -*4.4,*

Number of Ratings and Reviews- *10235 Ratings and 920 Reviews.*

Features -*6 GB RAM | 64 GB ROM | Expandable upto 512 GB,16.26 cm (6.4 inch) Full HD+ Display 25MP + 5MP + 8MP | 25MP Front Camera,4000 mAh Lithium-ion Battery, Exynos 9610 Processor, Super AMOLED Display, Brand Warranty of 1 Year Available for Mobile and 6 Months for Accessories.*

Description - Do a lot more than just text and call your friends with the *Samsung Galaxy A50* smartphone. The *Exynos 9610 Octa-core Processor* makes multitasking a breeze. Take photography a notch higher with its revolutionary *Triple Camera System* that comprises a *25 MP Low Light Camera*, *8 MP Ultra-wide* Camera and *5 MP Live Focus Camera.*

[('***5Star***', '6430'), ('***4Star***', '2371'), ('***3Star***', '675'), ('***2Star***', '215'), ('***1Star***', '544')] i.e ,6430/10235 users rated 5/5 for this Product.

[('***Camera***', '3.6'), ('***Battery***', '3.0'), ('***Display***', '3.7'), ('***Value for Money***', '4.2')] –Ratings out of 5.

All the above details were considered while recommending the mobile phone to the users. A training model proposed to use all the details extracted and to recommend the mobiles phone.

4 Proposed System

The architecture of the proposed mobile phone recommendation system is presented in Figure 1.

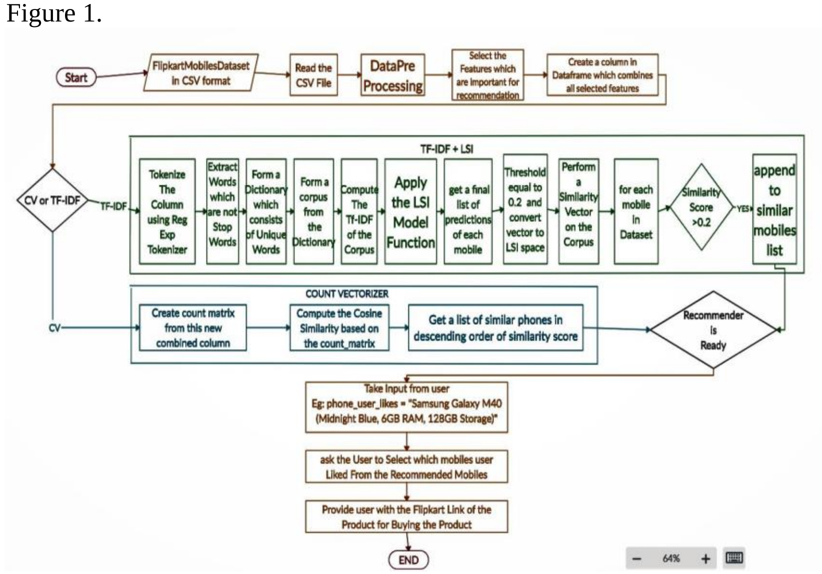

Fig. 1. Architecture diagram – Proposed recommendation system.

4.1 Prediction using *LSI* model

After extracting the combined features for each mobile phone in the dataset as mentioned above, tokenizing the combined features column was carried out using *RegExpTokenizer*. After tokenization, extracted the words which are not ‘stop words’. Stop words are the common words in a language which do not add value to the learning process and they don't add to the uniqueness factor of the feature set. Then these extracted unique words i.e., non-stop words were combined to form a dictionary. The dictionary consists of a combination of distinctive words.

In the next step, a corpus is formed from the dictionary. The corpus is the group of gathered information of all synopses pre-processed and transformed using the dictionary. Next, the term frequency-inverse document frequency (*tf-idf*) transformation was performed. The *tf-idf* transformation gives an indication of weights or importance of words in the corpus. The *tf-idf* value on the count of number of words in a document and also on the number of times it appears in the corpus. The “*tf-idf*” function transforms the corpus from a grouped integer counts format into *tf-idf* real values weight. Then, computed the *tf-idf* of the Corpus which was extracted earlier (corpus-*tf-idf*). Next, the "*LSI* Model" was applied to transform *tf-idf* corpus into a latent n-dimensional space. The *LSI* model is present in the *Gensim* library of python. After transforming the corpus to *LSI* space, it was indexed. This was done using *MatrixSimilarity* function in similarities library. To get a final list of predictions of each mobile phone, a threshold was set to 0.2 initially. Then, iterated over each element of the list which stored the *tf-idf* of the corpus. During the iteration, the current element of the corpus *tf-idf* was converted to LSI space. Then, performed a similarity vector construction against the corpus to find the index of the *LSI* space transformed vector. Now, for each mobile phone in the dataset, it was checked that the corresponding index in the similarity vector (Similarity score) is greater than the threshold 0.2. If yes, it is appended to the similar mobile phones list of the current mobile in the iteration. So, at the end, a list was generated where for each mobile phone in the dataset its corresponding row showed all the similar mobile phones (the mobiles with similarity index > 0.2), Thus, a list of recommended mobile phones based on the content based prediction was obtained.

4.2 Prediction using Cosine Similarity

Cosine similarity is the measure of the similarity between two documents by counting the frequencies of similar terms and considering them as vectors. It is calculated by the cosine of the angle between two vectors and it ranges from 0 to +1 in the case of documents, because the frequencies of the terms cannot be negative. This measure suffers less from the curse of dimensionality than Euclidean distance. Document vectors tend to get long because there will be many words in a given document. So *cosine similarity* is a way to avoid the curse of dimensionality, which is measured by the Equation (1).

$$\textit{Cosine Similarity}\ (|\mathrm{x.y}|) = \frac{x.y}{||x||.||y||} \tag{1}$$

where $||x||$ is the norm or magnitude of vector $x = (x_1, x_2, x_3, \ldots\ldots x_n)$ *is*defined as $\sqrt{\mathrm{x}_1^2 + \mathrm{x}_2^2 + \mathrm{x}_3^2 + \cdots \mathrm{x}_n^2}$.

Basically, it is the Euclidean Distance of the vector from origin. Similarly, $||y||$ is the magnitude of vector y. If the *cosine similarity* value is closer to 1 then angle between the vectors is smaller and similarity between the vectors is greater.

After extracting the combined features for each mobile phone in the dataset, all the selected features are combined together. A count matrix was created from this new combined features. Now, *cosine similarity* was computed based on the count matrix. After this process, input was taken from the user. Based on the input given by the user, index of the mobile phone is retrieved from its name. A list of similar mobile phones was generated in descending order of similarity score, then names and details of first 20 similar mobile phones corresponding to the mobile phone the user had chosen were displayed. Then, the user can choose any of the mobile phone from the given top 20 recommendations and then a link to buy that mobile phone is given to the user.

5 Results and Discussions

The final result obtained based on the content based filtering is shown as an image in Figure 2. In Figure 2, the column -*'Mobile Name'* is the mobile phone which the user likes, and the columns *Mob1,Mob2…..Mob20* are the first 20 similar mobile phones recommended to the user and *Score1*, *Score2…..Score20* are the similarity indices of the top 20 recommendation, respectively. The set of recommendations for additional set of user inputs is depicted in Figure 3, for reference.

	Unnamed: 0	Mobile Name	Mob1	Score1	Mob2	Score2	Mob3	Score3	Mob4	Score4	...	Score16	Mob17	Score17	Mob18	Score1
0	0	Acer Z630S (Black & Gold/Black - G, 32 GB)	Acer Liquid Z530 (Black, 16 GB)	0.949191	Panasonic P90 (Black, 16 GB)	0.742600	Yu Yunicorn (Gold Rush, 32 GB)	0.738004	Lava X50 4G (Blue Silver, 8 GB)	0.731686	...	0.675502	Samsung Galaxy A8 (Black, 32 GB)	0.664698	Honor 4X (White, 8 GB)	0.66127
1	1	Acer Liquid Z530 (Black, 16 GB)	Acer Z630S (Black & Gold/Black - G, 32 GB)	0.949191	Lava X81 4G with VoLTE (Space Grey, 16 GB)	0.772154	Panasonic P95 (Blue, 16 GB)	0.736302	Panasonic P95 (Grey, 16 GB)	0.733992	...	0.678175	Yu Yunicorn (Rush Silver, 32 GB)	0.675824	Alcatel Idol 4 (Dark Grey, 16 GB)	0.67337
2	2	Adcom J1	Adcom A-J1	0.985032	Gfive Classic	0.663438	Gfive Audi	0.635590	Mesee M3	0.632486	...	0.527197	Diamond D09	0.525580	Snexian Splender	0.52392
3	3	Adcom AJ2	Adcom AJ2	0.715455	InFocus F110	0.703642	Adcom A101 Voice Changer	0.688862	M-tech G24	0.672982	...	0.536631	Gfive Classic	0.533683	Karbonn K88	0.52246

Fig. 2. Results for content based filtering.

Mobile Name	Mob1	Score1	Mob2	Score2	Mob3	Score3	Mob4	Score4	Mob5	Score5	Mob6	Score6	Mob7	Score7	Mob8	Score8	Mob9
OPPO F15 (Light	OPPO F15 (Unicorn	0.99930245	Realme X2 (Pearl Wl	0.866089	OPPO A52 (Stre	0.84243	OPPO Reno2 (	0.839641	Realme 5 Pro	0.828523	Realme 5 Pro (S	0.82676	Realme 5 Pro (Cr	0.826084	Realme X50	0.814911	OPPO Ren
OPPO K1 (Piano	OPPO K1 (Piano Bla	1	OPPO K1 (Astral Blu	0.898959	OPPO A1K (Red,	0.767842	OPPO F11 (Ma	0.72053	OPPO F11 (Je	0.71967214	OPPO F11 Pro (	0.709833	OPPO A5s (Gold,	0.704337	OPPO A5s (	0.704301	OPPO A5s
OPPO F11 (Marb	OPPO F11 (Marble	1	OPPO F11 (Marble G	0.968495	OPPO F11 (Fluo	0.955917	OPPO F11 Pro	0.923688	OPPO F11 Pro	0.90648144	OPPO A1K (Red	0.897017	OPPO F9 Pro (Su	0.77572	OPPO A5s (	0.765102	OPPO A5s
OPPO A7 (Glarin	OPPO A7 (Glaring G	0.99999994	OPPO A7 (Glaze Blu	0.998804	OPPO A7 (Glarir	0.988488	OPPO A7 (Gla	0.987396	OPPO A5s (Bl	0.9077667	OPPO A5s (Blue	0.906894	OPPO A5s (Black,	0.906089	OPPO A5s (	0.906055	OPPO A5s
OPPO F11 Pro (A	OPPO F11 Pro (Aur	1	OPPO F11 Pro (Wat	0.971895	OPPO F11 (Marl	0.950457	OPPO F11 (Ma	0.923688	OPPO F11 (Je	0.92232174	OPPO A1K (Blac	0.87172	OPPO F9 Pro (Su	0.871	OPPO F15 (	0.779923	OPPO R17
OPPO A7 (Glarin	OPPO A7 (Glaring G	1	OPPO A7 (Glaze Blu	0.998811	OPPO A7 (Glarir	0.988488	OPPO A7 (Gla	0.987203	OPPO A5s (Bl	0.8854592	OPPO A5s (Blue	0.884524	OPPO A5s (Black,	0.883472	OPPO A5s (	0.883181	OPPO A5s
OPPO A7 (Glaze	OPPO A7 (Glaze Blu	1	OPPO A7 (Glaring Gc	0.998811	OPPO A7 (Glaze	0.988494	OPPO A7 (Gla	0.987396	OPPO A5s (Bl	0.8871393	OPPO A5s (Blac	0.887013	OPPO A5s (Blue,	0.88513	OPPO A5s (	0.884974	OPPO A5s
OPPO A7 (Glaze	OPPO A7 (Glaze Blu	0.9999999	OPPO A7 (Glaring Gc	0.998804	OPPO A7 (Glaze	0.988494	OPPO A7 (Gla	0.987203	OPPO A5s (Bl	0.9093351	OPPO A5s (Blac	0.909142	OPPO A5s (Blue,	0.907612	OPPO A5s (	0.907412	OPPO A5s
OPPO Reno2 Z (S	OPPO Reno2 Z (Sky	1	OPPO Reno2 (Ocear	0.900154	OPPO F15 (Blazi	0.818827	OPPO F15 (Lig	0.805521	OPPO A52 (Tv	0.7490188	OPPO F9 Pro (S	0.74024	OPPO F9 Pro (Sta	0.740028	Realme XT (	0.732774	OPPO F11
OPPO A5 2020 (I	OPPO A5 2020 (Mir	1	OPPO A5 2020 (Dazz	0.999471	OPPO A9 2020 (	0.980609	OPPO A9 202(	0.979974	OPPO A9 202	0.91733825	OPPO A52 (Twil	0.702163	OPPO Reno2 Z (S	0.646654	Realme 5s (	0.633565	Realme 5s
OPPO F9 Pro (Su	OPPO F9 Pro (Sunri	1	OPPO F11 Pro (Auro	0.871	OPPO F11 Pro (	0.863965	OPPO F11 (Ma	0.826601	OPPO F11 Pro	0.8133728	Realme 3 (Blac	0.781326	OPPO F11 (Jewel	0.777772	OPPO F11 (	0.77572	Realme 3 (
OPPO A5s (Gree	OPPO A5s (Green, 6	1	OPPO A5s (Black, 32	0.997845	OPPO A5s (Blac	0.997739	OPPO A5s (Blu	0.997084	OPPO A5s (Bl	0.99692434	OPPO A5s (Red,	0.996904	OPPO A7 (Glaze	0.907271	OPPO A7 (G	0.906055	OPPO A1K
OPPO A5s (Blue,	OPPO A5s (Blue, 32	1	OPPO A5s (Blue, 32	0.999893	OPPO A5s (Blac	0.999348	OPPO A5s (Re	0.998123	OPPO A5s (Gr	0.99692434	OPPO A5s (Gol	0.996342	OPPO A7 (Glaze	0.909335	OPPO A7 (G	0.906894	OPPO A1K
OPPO A5s (Red,	OPPO A5s (Red, 32	1	OPPO A5s (Black, 32	0.99922	OPPO A5s (Blue	0.998287	OPPO A5s (Blu	0.998123	OPPO A5s (Gr	0.996904	OPPO A5s (Gol	0.996442	OPPO A7 (Glaze	0.906609	OPPO A7 (G	0.905528	OPPO A1K
OPPO A5s (Blue,	OPPO A5s (Blue, 32	0.9999999	OPPO A5s (Blue, 32	0.999893	OPPO A5s (Blac	0.999294	OPPO A5s (Re	0.998286	OPPO A5s (Gr	0.99708414	OPPO A5s (Gol	0.99651	OPPO A7 (Glaze	0.907612	OPPO A7 (G	0.905229	OPPO A1K
OPPO A5s (Black	OPPO A5s (Black, 3	1	OPPO A5s (Blue, 32	0.999348	OPPO A5s (Blue	0.999294	OPPO A5s (Re	0.99922	OPPO A5s (Gr	0.99773854	OPPO A5s (Gol	0.997162	OPPO A7 (Glaze	0.909142	OPPO A7 (G	0.907767	OPPO A1K
OPPO R17 (Ambi	OPPO R17 (Neon P	0.9585908	Samsung Galaxy M3	0.690574	Samsung Galaxy	0.672268	Huawei Y9 (Sa	0.650127	Huawei Y9 (B	0.6077547	Samsung Galax	0.587142	Samsung Galaxy	0.566542	Vivo U20 (B	0.536896	Samsung G
OPPO R17 (Neor	OPPO R17 (Ambien	0.9585908	Honor View 20 (sap	0.67726	Samsung Galaxy	0.643873	Samsung Gala	0.619999	Huawei Y9 (S	0.60010815	Samsung Galax	0.553696	LG Q60 (New Mc	0.551761	Samsung Ga	0.510756	OPPO K3 (
OPPO Reno2 (Oc	OPPO Reno2 (Ocea	1	OPPO Reno2 Z (Sky	0.900154	OPPO F15 (Blazi	0.848071	OPPO F15 (Lig	0.839641	Realme X2 (P	0.79591024	OPPO A52 (Twil	0.778786	Realme 5 Pro (Sp	0.776446	Realme 5 Pr	0.775315	Realme XT
OPPO K1 (Piano	OPPO K1 (Piano Bla	1	OPPO K1 (Astral Blu	0.898959	OPPO A1K (Red,	0.767842	OPPO F11 (Ma	0.72053	OPPO F11 (Je	0.71967214	OPPO F11 Pro (	0.709833	OPPO A5s (Gold,	0.704337	OPPO A5s (	0.704301	OPPO A5s
OPPO F11 (Marb	OPPO F11 (Marble	1	OPPO F11 (Marble G	0.968495	OPPO F11 (Fluo	0.955917	OPPO F11 Pro	0.923688	OPPO F11 Pro	0.90648144	OPPO A1K (Red	0.897017	OPPO F9 Pro (Su	0.77572	OPPO A5s (	0.765102	OPPO A5s
OPPO F11 Pro (A	OPPO F11 Pro (Aur	1	OPPO F11 Pro (Wat	0.971895	OPPO F11 (Marl	0.950457	OPPO F11 (Ma	0.923688	OPPO F11 (Je	0.92232174	OPPO A1K (Blac	0.87172	OPPO F9 Pro (Su	0.871	OPPO F15 (	0.779923	OPPO R17
OPPO A7 (Glarin	OPPO A7 (Glaring G	0.99999994	OPPO A7 (Glaze Blu	0.998804	OPPO A7 (Glarir	0.988488	OPPO A7 (Gla	0.987396	OPPO A5s (Bl	0.9077667	OPPO A5s (Blue	0.906894	OPPO A5s (Black,	0.906089	OPPO A5s (	0.906055	OPPO A5s
OPPO A7 (Glarin	OPPO A7 (Glaring G	1	OPPO A7 (Glaze Blu	0.998811	OPPO A7 (Glarir	0.988488	OPPO A7 (Gla	0.987203	OPPO A5s (Bl	0.8854592	OPPO A5s (Blue	0.884524	OPPO A5s (Black,	0.883472	OPPO A5s (	0.883181	OPPO A5s
OPPO A7 (Glaze	OPPO A7 (Glaze Blu	1	OPPO A7 (Glaring Gc	0.998811	OPPO A7 (Glaze	0.988494	OPPO A7 (Gla	0.987396	OPPO A5s (Bl	0.8871393	OPPO A5s (Blac	0.887013	OPPO A5s (Blue,	0.88513	OPPO A5s (	0.884974	OPPO A5s
OPPO A7 (Glaze	OPPO A7 (Glaze Blu	0.9999999	OPPO A7 (Glaring Gc	0.998804	OPPO A7 (Glaze	0.988494	OPPO A7 (Gla	0.987203	OPPO A5s (Bl	0.9093351	OPPO A5s (Blac	0.909142	OPPO A5s (Blue,	0.907612	OPPO A5s (	0.907412	OPPO A5s

Fig. 3. Recommendation for additional set of user inputs.

The comparison of *LSI* and *cosine similarity* technique was carried out for the mobile models 'OnePlus 7T Froster Silver, 256GB' and 'Realme XT Pearl Blue, 64GB'. The methods are compared based on the similarity score. The results obtained for 'OnePlus 7T Froster Silver, 256GB' and 'Realme XT Pearl Blue, 64GB' is presented in Figure 4 and Figure 5, respectively. The Figures display the mobile phones recommended by the two techniques for a given mobile phone in *x* axis. The similarity score is used as reference for *y*-axis. It was found from the results that the similarity scores of many of the recommended mobiles phones are similar by the cosine similarity technique. However, the similarity scores are varying for each of the mobile phone recommended by the *LSI* technique.

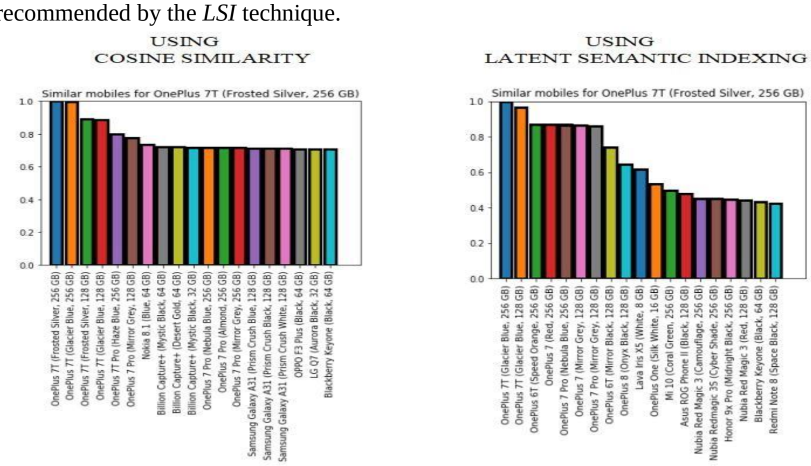

Fig. 4. Comparison of LSI and Cosine similarity techniques for 'OnePlus 7T Froster Silver, 256GB'

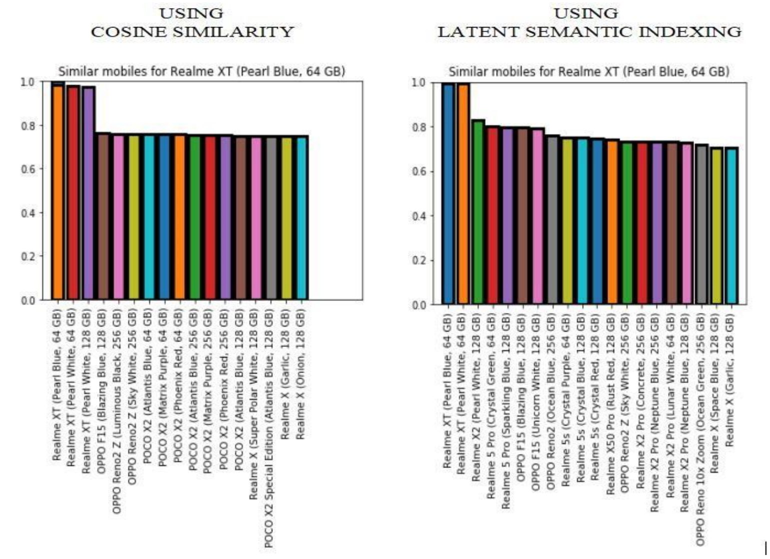

Fig. 5. Comparison of LSI and Cosine similarity techniques for 'Realme XT Pearl Blue, 64GB'.

6 Conclusion

A system for mobile phone recommendation was proposed and its architecture, implementation details and the results were discussed in this paper. The experimental analysis of the proposed system was done based on the *LSI* and *cosine similarity* methods. The similarity indices were taken as reference for the recommendation. The proposed system could map each of the mobile phone to its similar mobile phones in the dataset based on the filtering done by keeping a threshold for relevance. Proper and nearly accurate prediction of similar mobile phones based on all the input features and descriptions was achieved.

The work can be further extended to be applied in real life where it can be implemented in the form of an application which does comparisons between multiple websites selling the same product and gives an in depth analysis of all the similar mobile phone to the input/searched mobile phone.

Also the work can be further extended to be applied to give an audio based output/recommendations and later when the final target is chosen from the similar choices, the text from the audio can be extracted and in depth analysis of the final choice can be provided.

References

1. Sanjeevan Sivapalan, Alireza Sadeghian, Hossein Rahnama and Asad M. Madni, "Recommender Systems in E-Commerce", *In Proceedings of 2014 World Automation Congress (WAC)*, DOI. 10.13140/2.1.3235.5847, 2014.
2. Francesco Ricci, "Mobile Recommender Systems", *Jounral of Information Technology and Tourism*, Vol. 12., No. 2, pp. 205-231, 2011.
3. Yan Sun, Woon Chong, Han, Y.-S, Seungmin Rho and Ka Lok Man, "Key Factors Affecting User Experience of Mobile Recommendation Systems", *Lecture Notes in Engineering and Computer Science*, Vol. 2., pp. 724-728, 2015.
4. Folasade Isinkaye, Yetunde Folajimi and Bolanle Ojokoh, "Recommendation systems: Principles, methods and evaluation", *Egyptian Informatics Journal*, Vol. 16, 2015.
5. Nikolaos Polatidis and Christos K. Georgiadis, "Mobile recommender systems: An overview of technologies and challenges", *In Proceedings of 2nd International Conference on Informatics and Applications*, 2013.
6. Marwa Hussien Mohamed, Mohamed Helmy Khafagy and Mohamed Hasan Ibrahim, "Recommender Systems Challenges and Solutions Survey", *In Proceeding of International Conference on Innovative Trends in Computer Engineering (ITCE)*, 2019.
7. Neelam V Phadnis and Jayant Gadge, "Framework for Document Retrieval using Latent Semantic Indexing", *International Journal of Computer Applications*, Vol. 94., pp. 37-41., 2014.
8. Sivaramakrishnan A, Krishnamachari M, and Vidhya Balasubramanian, "Recommending Customizable Products: A Multiple Choice Knapsack Solution", *In Proceedings of the 5th International Conference on Web Intelligence, Mining and Semantics*, New York, NY, USA, 2015.
9. Jia-Ming Low, Ian KT Tan and Choo-Yee Ting, "Recent Developments in Recommender Systems", *In Book Multi-disciplinary Trends in Artificial Intelligence, 13th International Conference MIWAI 2019*, pp. 38-51, 2019.
10. Yedugiri K. B, Chandni S, Sini Raj P and Souparnika, S, "Recommender systems - A deeper insight", *International Journal of Applied Engineering Research*, Vol. 9., pp. 28521-28531., 2014.
11. Angelica Aquino and Enrico Chavez, "Analysis on the use of Latent Semantic Indexing (LSI) for document classification and retrieval system of PNP files". *In proceedings of 2nd International Conference on Material Engineering and Advanced Manufacturing Technology (MATEC)*, Vol. 189., No 4. 2018.
12. Zouaoui Samia and Rezeg Khaled, "Ontology-based Approach for Semantic Indexing of Arabic Documents", *In Proceedings of 1stInternational Conference on Artificial Intelligence and its Applications* (*AIAP' 2018*), 2018.
13. Sudarsun Santhiappan and Venkatesh Gopalan, "Finding the Optimal Rank for LSI Models", In Proceedings of *ICAET*, 2010.