International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-11 Issue-4, November 2022

Apparel Recommendation System using Content-Based Filtering

Sagar Yeruva, Addi Sathvika, Damera Sruthi, Duggasani Yaswanth Reddy, Gogineni Gopi Krishna

Abstract: ***Nowadays, people are constantly moving towards various fashion products as a result the e-commerce market for garments is growing rapidly. Online stores must update their features according to user requirements and preferences. However, there are too many options for users to select from these online stores which may leave them in a dilemma to identify the correct outfit, save the user time, and increase sales, efficient recommendation systems are becoming a necessity for online retailers. In this paper, we proposed an Apparel Recommendation System that generates recommendations for users based on their input. We used a real-world data set taken from the online market giant Amazon using Amazon's Product Advertising API. We aim to use keywords like brand, color, size, etc., to recommend. Data exploration to get detailed information about our dataset, Data Cleaning(pre-processing) to remove invalid sections, Model selection (We have compared different feature extraction techniques like bag of words, TF-IDF, and word2vec model) to find out efficient techniques and Deployment of the model that could facilitate recommendation system to simplify the task of apparel recommendation system. The accuracy of the model is identified using the response time and content matching.***

Keywords: Apparel, Recommendation, TF-IDF, Bag-of-Words, Content-Based-Filtering

I. INTRODUCTION

In the present world, a user usually adopts a system that provides services to make the user's job easier. A recommendation system is one such feature that makes the user's experience comfortable and easy. A user exploring all the products on the website for shopping for a specific product is not an easy and possible task. Here Recommendation Systems play a very important role which helps the user find what they are looking for. If a user is searching for a shirt according to his preferences and finds a particular shirt that matches his choice, then if the shopping website recommends shirts that are alike to the chosen shirt it enhances the user's shopping experience. This will lead to happy customers which will also increase sales and customer engagement with the company. In this way, the recommender system benefits users in finding items of their interest. Recommender systems guide users to the items which users are most likely to purchase. Recommender systems are like a well-known service person who knows the user's history and makes his decision-making process easier. The user may feel known and start exploring more and more products which ultimately leads to more purchases. Companies focus on personalized product recommendations to increase their sales. Personalized product recommendations help companies to engage more with users and provide a delightful shopping experience. Personalized product recommendations help in finding items that are most relevant to the users. When a user can relate to the recommendations perfectly with his choices then he is bound to visit the website again and again and chances are high that he may also become a loyal customer. A recommendation system is an added benefit to the companies which they can utilize to move ahead of their competitors in this competitive world.

Recommendation systems are of common use in e-commerce websites and online streaming companies. These recommendation systems allow customers of a company to see relevant products. The ease of use of the products and being given more customer-related choices enhance the user experience. The recommender systems help in enhancing customer engagement and brand loyalty. They reduce the transaction costs of finding and selecting items in an online shopping environment. Recommender systems create a delightful user experience while driving incremental revenue for websites/companies.

Our work has a user web interface from which the user can provide a string containing the details of the product like color, product type, product brand, etc. as these are one of the main parameters based on which a user intends to get similar results of the product and also as the input user can provide several similar products, he wants to retrieve so that he may choose a wide range of products according to his available time and resources. The dataset is processed by the importance of features. ASIN number in the dataset plays a key role in uniquely identifying a product. The text-based recommendation of the products makes the title a suitable feature to be selected for the data analysis. So, TF-IDF based prediction is the main model used for text-based prediction.

Manuscript received on 25 October 2022 | Revised Manuscript received on 31 October 2022 | Manuscript Accepted on 15 November 2022 | Manuscript published on 30 November 2022.

* Correspondence Author (s)

Sagar Yeruva*, Department of Computer Science and Engineering, (AIML and IoT), VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad (Telangana), India. Email: sagar_y@vnrvjiet.in

Addi Sathvika, Department of Computer Science and Engineering, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad (Telangana), India.

Damera Sruthi, Department of Computer Science and Engineering, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad (Telangana), India.

Duggasani Yaswanth Reddy, Department of Computer Science and Engineering, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad (Telangana), India.

Gogineni Gopi Krishna, Department of Computer Science and Engineering, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad (Telangana), India.

The output of the project is the listing of products whose listing is according to the given number of similar products and also their brief details. Recommendation systems are useful for customers and companies. They help in reducing transaction costs of finding and choosing items in an online shopping environment. Recommendation systems play a key role in improving revenues, and they are the major reason for selling more products [1]. The recommender system helps in increasing the average order values. This is because the website displays more personalized choices since the customers like to have the products that they strongly desire. Recommender systems also help in the increasing number of items per order. When the applications display the products which meet the user's interest, then the user is more likely to add those items to the bag [2]. The most successful company amazon reported that there is a 29% increase in sales because of recommender systems [3]. And it also specified that 35% of its revenue is generated by recommendation systems[4]. Netflix which has 182.8 million subscribers specified that 80% of the whole streaming time is achieved only by their recommender systems [5]. Flipkart stated that due to the recommender system there was a 10% increment in the CTR [6].

II. EXISTING-STATE-OF-THE -ART

A. Artificial Intelligence in Recommender System [7]

Zhang, Qian, Jie Lu, and Yaochu Jin provided insights into the main models and methods of recommendation systems and also the usage of computer vision techniques in recommendation systems. A detailed role of Neural networks in recommender systems is mentioned in the paper.

B. Modular Architecture for Recommender System [8]

This research paper discusses the various architectures for recommender systems and proposed a modular and extensible recommender system architecture the architecture proposed is implemented on a Brazilian e-commerce website.

C. Recommendation system development for fashion retail e-commerce [9]

Hwangbo, Hyunwoo, Yang Sok Kim, and Kyung Jin Cha talk about developing a real-time recommender for a large Korean Fashion Company. They extended existing item-based collaborative filtering to create a new system called K-RecSys which combined online product click data and offline product sales. They tested the system using A|B testing and got better results than the existing system.

D. Garment Recommendation with Memory Augmented Neural Networks [10]

In this paper, the author used Memory Augmented Neural Network which is used for one-shot learning tasks and they stored a non-redundant subset of samples. They used matrix factorization to include user preferences.

E. Collaborative approach for research paper recommender system [11]

In the following paper, a collaborative approach for the research paper recommender system is presented to help researchers on finding works of their interest over the Information provided on the internet.

F. Literature survey on recommendation system based on sentimental analysis [12]

Jain, Achin, Vanita Jain, and Nidhi Kapoor presented an analysis of different approaches to the recommender system based on sentiment analysis. This case study gives statistics and approximate trends in recommender structures research on the recommender system using sentimental analysis.

G. Clothes Matching and Recommendation Systems based on user Attributes [13]

This paper gives a clear analysis of various systems, their methodologies, and approaches, algorithms that have been used. It also proposes a system that blends multifarious approaches and provides the features of apparel recommendation.

H. Clothing Recommendation System Based on Advanced User-Based Collaborative Filtering Algorithm [14]

This research paper assesses the famous UCF algorithm, and it also discusses its application in apparel recommendation systems. It designs the advanced UCF algorithm which helps in improving the efficiency of calculation of the similarity among customers.

III. DATASET

The dataset that we are using in the project is very exciting. It is extracted by web scraping the Amazon apparel product data through Amazon product advertising API. From most of the brands in the dataset, we are inferring that the dataset is taken from amazon.com which is a USA website of amazon (www.amazon.com).

A. Features of dataset

The dataset consists of women’s clothing products. It has 1,83,138 data points and 19 features. In this project, we limited ourselves to only 7 features based on certain criteria. The 7 attributes of the dataset that we are using in our project are:

- **ASIN**- Amazon Standard Identification Number. It is a unique identifier of 10 letters/ numbers for a product that’s assigned by Amazon. It is primarily used for product identification and also solves the problem of duplicate products by merging them. Amazon follows the rule that while a product exists in the amazon catalog the new seller needs to use an existing ASIN number which is the same for all the sellers who sell the same product and this is usually the case for resellers, products with wide distributions. A product that is not in the Amazon catalog will be assigned a unique ASIN number [15]. In this way the ASIN number is unique for every unique product, so we considered ASIN in our data analysis as it is unique for every product and it allows easy identification of product.
- **Brand**- The brands that are on sale for Amazon are unique and there will not be any infringement or inaccurate brand information [16].

Retrieval Number: 100.1/ijrte.D73311111422
DOI: 10.35940/ijrte.D7331.1111422
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

Amazon does not accept listings that do not have proper permissions for selling that product which comes under brand infringement and it also does not accept products in the branch field has generic naming like (Shirt) and unapproved brand [17]. So, the brands of apparel in amazon listings are accurate and they help in recommending similar brands' apparel which will help users to choose the products.

- **Colour**- While listing a product on the amazon website color of the product also plays a key role because two products that are the same but differ in color of the product can be under the "Variation Parent" property [18]. A customer may be interested in similar color products or different color products from the same brand. The Colour of the product is a kind of more complex data because we can have a color composition that is too diverse, especially for apparel and color is too important for apparel listing because the amazon website enables its users the featured shop by colour [19]. So, the color of a product also helps in the recommendation.
- **Medium image URL-** Amazon imposes certain criteria on the images that have to be followed by sellers while listing a product. Images almost show the story as much as the reading of the description or details of the product. The medium-sized URL feature of data helps in retrieving the image and using it for interactive display and the user easily assesses the products.
- **Product type name-** Amazon's classification system uses values supplied by sellers to determine where a product appears in the catalog and the product type of that particular product [20]. The product type name helps in grouping related. The user can recommend products according to his/her relevant products from the group of products.
- **Title-**The title of a product is almost like a short description of the product. It conveys a lot of information about a product. The title of a product is considered as clean data about a product as Amazon imposes strict rules like the title should be of fixed length and there should not be any promotional offers or subjective commentary such as “hot item”, “fast selling” etc.. and should not contain non-language ASCII characters [21]. So, with all of these rules in the title section, it is easy to analyze and generate insights.
- **Formatted price**- The price details of the product too help in recommending similar items. The prices of products that are featured on the amazon website are more dynamic because they use this feature of dynamic pricing as the main idea to adjust their prices at a rapid pace to meet the market demand than any other e-commerce website [22]. The too-complex pricing model makes over 250 million price updates. An average product price will change once every 10 minutes [23]. The formatted price has too many null rows/data points because Walmart has scraped the amazon data and got the amazon pricing for products and relevantly adjusted the product prices on the Walmart website. So, to remain in the competitive business amazon pricing data may not be revealed in web scraping.

B. Unused Features

The other attributes of the dataset which are not made into the model for a recommendation of products are:

- **SKU**- SKU which stands for stock keeping unit is a number that can be created by a seller (according to to sell.amazon.in). The seller may mark a batch of products with the same SKU number which may lead to confusion also in the dataset we can observe that most of the SKU numbers are not provided. On the other hand, the ASIN number stands as a great alternative for all situations where SKU cannot be relied on.
- **Availability and availability type**- Availability and availability of a product do not give any necessary details for the apparel recommendation project because we are addressing only the recommendation part of apparel and not the selling or business aspects of the online apparel system.
- **Reviews, editorial review and editorial review**- Reviews do not give much information to process in the apparel recommendation system as the approach for the recommendation system remains content-based filtering, not collaborative filtering. So, we are not going to consider the approach of a user towards a product in the recommender systems. And we are not going to use the sentiment analysis part in this model.
- **Author**- The author feature in the dataset has no values at all.
- **Publisher and Manufacturer-** While a user searching for a product he may consider products from the same brand but surely not just by considering only the manufacturer. A manufacturer of apparel can be a third party who manufactures a product on behalf of a company but not manufactures only a specific brand of apparel and the publisher stands as the third-party person who may sell different brands of apparel who sells products on behalf of the company
- **Large image URL and small image URL-** As we have already considered medium image URLs for the visual representation of the recommended results, we did not consider large image URLs and small image URLs.
- **Model-** The model represents the person who is featured in photos with the model wearing apparel and the model information does not convey much useful information for the recommender systems so it is not useful in the recommendation of apparel.

C. Data Pre-processing

The dataset that we have taken is of size ~183k records. And by eliminating unwanted data items and by performing text pre-processing, and deduping we reduced it to ~16k. Initially, we eliminated the data items that have a price as a null value because the price is the main option that is considered by any average consumer while selecting or searching for a product. On eliminating those data items which do not have price values we are left with 28,395 records as shown in figure-1. Then we eliminated the data items that are having color values as null in other words which are not having any price details.

On eliminating those data items which do not have color values we are left with 28,385 records. The below snippet of code describes the process that is done for eliminating null price and null color items.

Data Cleaning

```
In [4]: #removing the rows which are having null values for price
data = data.loc[~data['formatted_price'].isnull()]
print('Number of data points after eliminating price=NULL :', data.shape[0])

Number of data points after eliminating price=NULL : 28395

In [5]: #removing the rows which are having nulls values for color
data = data.loc[~data['color'].isnull()]
print('Number of data points after eliminating color=NULL :', data.shape[0])

Number of data points after eliminating color=NULL : 28385
```

Fig. 1: Removal of null-valued rows

After removing the data items that have price and color as null, there are some products whose title length is less than 5 words. Those titles do not convey any powerful meaning for the analysis because those types of titles will generally be abstract like T-shirt boys tops, black shirts women wear, etc. These types of titles enter into the data even though amazon policies were made to control the title for the product. So, those data points are removed from the dataset, and then we are left with 27255 products/data points as shown in figure-2. Titles are sorted alphabetically to be used in the next processing. The code snippet below depicts the implementation.

```
In [14]: data=pd.read_pickle('pickles/28k_apparel_data')

In [15]: #removing the products whose titles are too short that is the titles whose number of words are less than 5
data_sorted = data[data['title'].apply(lambda x: len(x.split())>5)]
print("Number of data points after removal of products with short description/title:", data_sorted.shape[0])

Number of data points after removal of products with short description/title: 27255

In [16]: data_sorted.sort_values('title',inplace=True, ascending=False)
data_sorted

Out[16]:
```

	asin	brand	color	medium_image_url	product_type_name	title	formatted_price
61973	B06Y1KZ2WB	Éclair	Black/Pink	https://images-na.ssl-images-amazon.com/images...	SHIRT	Éclair Women's Printed Thin Strap Blouse Black...	$24.99
133820	B010RV33VE	xiaoming	Pink	https://images-na.ssl-images-amazon.com/images...	SHIRT	xiaoming Womens Sleeveless Loose Long T-shirts...	$18.19
81461	B01DDSDLNS	xiaoming	White	https://images-na.ssl-images-amazon.com/images...	SHIRT	xiaoming Women's White Long Sleeve Single Brea...	$21.58
75995	B00X5LYO9Y	xiaoming	Red Anchors	https://images-na.ssl-images-amazon.com/images...	SHIRT	xiaoming Stripes Tank Patch/Bear Sleeve Anchor...	$15.91
151570	B00WPJG35K	xiaoming	White	https://images-na.ssl-images-amazon.com/images...	SHIRT	xiaoming Sleeve Sheer Loose Tassel Kimono Woma...	$14.32
...	...	...	...	...	...	...	...
12050	B06WRW8RQ1	AJ	Black	https://images-na.ssl-images-amazon.com/images...	SHIRT	"Life is a Journey" Self-Help DIY T-Shirt (Wom...	$9.38
40451	B073SKNQHD	The Workout Princess	Premium Heather Gray	https://images-na.ssl-images-amazon.com/images...	SHIRT	"I Workout To Burn off The Crazy", Tri Blend T...	$24.99
109599	B00KI3VDXM	Crazy4Bling	Purple	https://images-na.ssl-images-amazon.com/images...	SHIRT	"I Wanna Be Adored" Long Sleeve Top with Shred...	$39.99

Fig. 2: Removal of short title rows

And then in our data, we have around 1981 titles which are having duplicate titles i.e., the same titles, and also some of the titles are of the format

- Woman's place in the house and the senate shirts for Women's XXL white.
- Woman's place in the house and the senate shirts for Women's M Grey.

These are the titles that only differ in the last two words. Such titles are eliminated because they almost all represent the same title.

Number of exactly same titles

```
In [17]: print('Number of products having duplicated titles are:',sum(data_sorted.duplicated('title')))

Number of products having duplicated titles are: 1981
```

Removing titles which only differs in the last 2 words

```
In [40]: indices = []
for i,row in data_sorted.iterrows():
    indices.append(i)
```

Fig. 3: Duplicate title rows removal

So, in the below code snippet, both the duplicate and near duplicate data items are removed from the dataset as shown in figure-3, which makes the data more dependable. So, after the process, we are left with 17286 data items.

```
In [41]: import itertools
stage1_dedupe_asins = []
i = 0
j = 0
num_data_points = data_sorted.shape[0]
while i < num_data_points and j < num_data_points:
    previous_i = i
    a = data['title'].loc[indices[i]].split()
    j = i+1
    while j < num_data_points:
        b = data['title'].loc[indices[j]].split()
        length = max(len(a), len(b))
        count  = 0
        for k in itertools.zip_longest(a,b):
            if (k[0] == k[1]):
                count += 1
        # if the number of words in which both strings differ are < 2 , we are considering those two apperals as almost same, her
        if (length - count) > 2:
            stage1_dedupe_asins.append(data_sorted['asin'].loc[indices[i]])
            i = j
            break
        else:
            j += 1
    if previous_i == i:
        break

In [42]: data = data.loc[data['asin'].isin(stage1_dedupe_asins)]

In [43]: print('Number of data points : ', data.shape[0])

Number of data points :  17286
```

Fig. 4: Removal of titles that differ in less than 2 words

In the previous stage, we cleaned the data by removing the duplicates and also the titles which are different in the last two words of the title. So, in this stage, we clean the data by removing near duplicate elements which may not need to be different in the last few words. This stage of cleaning the data ends with 16176 data points as shown in figure-4.

IV. METHODOLOGY

The entire work of this paper is represented with the following architectural model diagram as shown in figure-5. This model takes user preferences as input in the web interface. This input will further be used to process and produce the best possible apparel recommendations. The user can give a string as the input to the model which may denote color, product type, or brand of product, as these can be the parameters on which a user intends to get similar recommendations and also can give several recommendations the user wants to retrieve.

Retrieval Number: 100.1/ijrte.D73311111422
DOI: 10.35940/ijrte.D7331.1111422
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

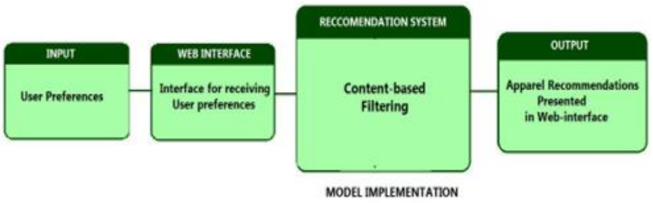

Fig. 5: System Architecture

The text-based recommendation of the products makes the title a suitable feature to be selected for the data analysis. So, the bag of words model, TF-IDF based prediction, IDF-based prediction, and Word2vec models are the main models used for text-based prediction. The output of the project is the listing of products whose listing is according to the given number of similar products and also their brief details. Finally, the Output of the project is apparel recommendations which will be presented in a web interface recommended by the model considering the preferences given by the user.

A. Bag of Words

The bag of Words model is used to convert text into some numerical format. It stores the frequency of the most frequent words. In the Bag of Words implementation of our model, we started with the preprocessed data where the titles don't have any stop words. We extracted a bag of words by taking all the product titles. For each title, we get a d-dimensional vector in which we have a count of occurrence of each word in that particular title. This is obtained with the help of the CountVectorizer function which is imported from the sklearn library. In this way, we get vectors for all 16042 titles. By running the fit transform function we get a matrix that has titles as rows and each of its words as columns. In the bag of words function, we are going to calculate the pairwise distances which is nothing but the Euclidean distance of title T with all of our points. And then we sort those distances accordingly, from that sorted array we are going to retrieve the required no of titles which has the smallest pairwise distance.

B. TF-IDF

TF-IDF model is an analytical measure that evaluates how relevant a word is to a document in a collection of documents [24]. TF-IDF is a combination of two extraction methods. They are TF - Term Frequency and IDF - Inverse Document Frequency.

$$\text{TF-IDF}(x) = \text{TF}(x) * \text{IDF}(x) \quad (1)$$

TF-Term Frequency
IDF- Inverse Document Frequency
Term frequency calculates how frequently a word occurs in a document.
TF(x)=count of the x in the document / total no of words in the document
Inverse Document Frequency calculates how common or rare a word is in the entire document set by which the importance of a word is known.

$$\text{IDF}(x) = \log(N / df + 1) \quad (2)$$

N-Total number of documents
df-number of documents with term x.
In our project, we used the pre-built function Tfidf Vectorizer () offered by python's library sklearn. feature_extraction. text. Here the function Tfidf Vectorizer (min_df = 0) converts our raw data into a matrix of TF-IDF features. Min_df is used to remove the terms that appear too infrequently. By default, it is set to 1 which ignores the terms that appear in less than one document. By setting it to 0 we didn't remove any data because the data was already cleaned and pre-processed. Next, we used the same fit transform function used in Bag of Words to get a sparse matrix that has titles as rows and each of its words as columns. Now that we had required a sparse matrix we implemented our tfidf_model () function. In the tfidf_model(), we used pairwise distances() from python's sklearn. metrics library which calculates the Euclidean distances of the passed doc_id's title and all the other titles in our dataset.

V. RESULT

The input given by the user is an animal pattern shirt. As mentioned before, the model is trained to retrieve the best-fit recommendations for the given input. In this instance, the model must return shirts of different kinds of animal patterns or a shirt that consists of animal pictures on it. We can observe that the major part of the shirts retrieved is zebra, tiger, and lion patterned or shirts with pictures of animals as shown in Fig. 6, 7, and 8.

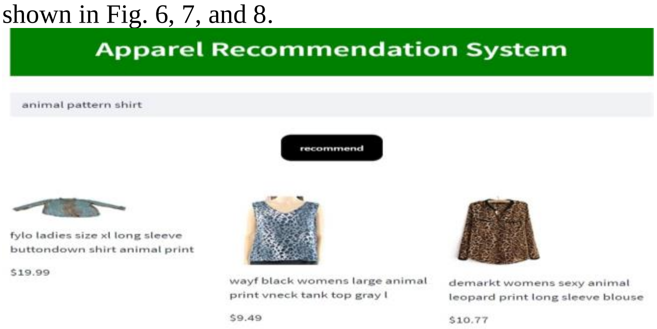

Fig. 6: Apparel Recommender Result Part-1

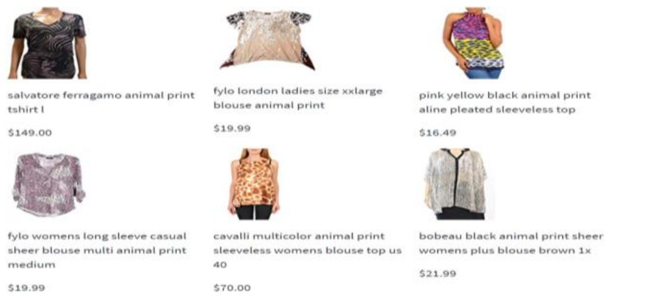

Fig. 7: Apparel Recommender Result Part-2

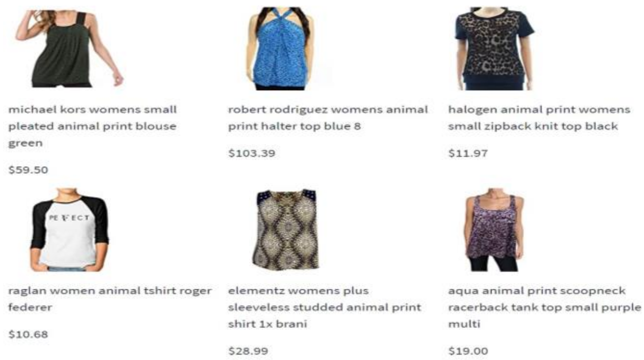

Fig. 8: Apparel Recommender Result Part-3

VI. DISCUSSION

The input from the web interface is forwarded to the tfidf_model() function which returns a list that contains all the relevant outputs for the given input string. The testing for the relevance of the output is based on how close the presented results are to the input string. For example when we give an 'animal patterned shirt' we get different shirts with either animal patterns on them or animal pictures on them but not some trousers which are completely irrelevant to the input string. In the web interface, part response time is in the range of around 1.2 to 2.7 seconds. So, the input string that is passed to these many functions retrieves all the relevant products in roughly around 1.5 seconds.

VII. CONCLUSION

In this paper, we were able to learn the usage of Bag of words and TF-IDF algorithms on the Amazon Apparel data. Firstly, we implemented the model using a bag of words algorithm. Based on the results, it was clear that the recommendations obtained are not desirable for the given input as words like shirt and zebra-pattern are given equal importance. Hence, we moved on to the TF-IDF algorithm which overcame the drawbacks of a bag of words. TF-IDF gives weightage to the words according to their uniqueness and importance of the word in each sentence which eventually resulted in desirable recommendations. We learned how to overcome the drawbacks of Bag of Words and make the recommendations more relevant.

REFERENCES

1. Isinkaye, Folasade Olubusola, Y. O. Folajimi, and Bolande Adefowoke Ojokoh. "Recommendation systems: Principles, methods, and evaluation." *Egyptian Informatics Journal* 16.3 (2015): 261-273. [CrossRef]
2. Röllecke, Felix Johannes, Arnd Huchzermeier, and David Schröder. "Returning customers: The hidden strategic opportunity of returns management." *California Management Review* 60.2 (2018): 176-203. [CrossRef]
3. Amazon's Recommendation Engine: The Secret To Selling More Online Available Online http://rejoiner.com/resources/amazon-recommendations-secret-selling-online (accessed on June 2021)
4. How Amazon Has Reorganised Around Artificial Intelligence and Machine learning Available Online: https://www.forbes.com/sites/blakemorgan/2018/07/16/how-amazon-has-re-organized-around-artificial-intelligence-and-machine-learning/?sh=250a76e77361 (accessed on June 2021)
5. Chong, D. "Deep dive into Netflix's recommender system." (2020).
6. Eisner, Alan, et al. "FLIPKART: WINNING IN INDIA?" *GLOBAL JOURNAL OF BUSINESS PEDAGOGY* 4.1 (2020): 79.
7. Zhang, Qian, Jie Lu, and Yaochu Jin. "Artificial intelligence in recommender systems." Complex & Intelligent Systems 7.1 (2021): 439-457. [CrossRef]
8. Prando, Allan Vidotti, and Solange N. Alves de Souza. "Modular Architecture for Recommender Systems Applied in a Brazilian e-Commerce." *JSW* 11.9 (2016): 912-923. [CrossRef]
9. Hwangbo, H., Kim, Y.S. and Cha, K.J., 2018. Recommendation system development for fashion retail e-commerce. *Electronic Commerce Research and Applications*, *28*, pp.94-101. [CrossRef]
10. De Divitiis, L., Becattini, F., Baecchi, C. and Del Bimbo, A., 2020. Garment Recommendation with Memory Augmented Neural Networks. arXiv preprint arXiv:2012.06200. [CrossRef]
11. Haruna, K., Akmar Ismail, M., Damiasih, D., Sutopo, J. and Herawan, T., 2017. A collaborative approach for research paper recommender system. *PloS one*, *12*(10), p.e0184516. [CrossRef]
12. Jain, Achin, Vanita Jain, and Nidhi Kapoor. "A literature survey on recommendation system based on sentimental analysis." *An International Journal (ACII)* 3.1 (2016). [CrossRef]
13. Pandit A, Goel K, Jain M, Katre N. A Review of Clothes Matching and Recommendation Systems based on User Attributes.
14. Liu, Yu, et al. "Clothing recommendation system based on advanced user-based collaborative filtering algorithm." *International Conference on Signal and Information Processing, Networking and Computers*. Springer, Singapore, 2017. [CrossRef]
15. Read More What is Amazon ASIN number & how to get it? Available online: https://www.datafeedwatch.com/blog/amazon-asin-number-what-is-it-and-how-do-you-get-it (accessed on June 2021)
16. Amazon Brand Registry: Help Protect Your Brand on Amazon Available online: https://brandservices.amazon.com (accessed on June 2021)
17. Amazon Brand Name Policy Available online: https://sellercentral.amazon.in/gp/help/external/G2N3GKE5SGSHWYRZ (accessed on June 2021)
18. Creating color variation on the existing product - Listing Management & Reports - Amazon Seller Forums Available online: https://sellercentral.amazon.com/forums/t/creating-color-variation-on-existing-product/150470 (accessed on June 2021)
19. Navigation by size or color Available online: https://sellercentral.amazon.com/gp/help/external/G53001 (accessed on June 2021)
20. Classify your products with Product Classifier Available online: https://sellercentral.amazon.in/gp/help/external/G200956770 (accessed on June 2021)
21. Product title requirements Available online: https://sellercentral.amazon.in/gp/help/external/YTR6SYGFA5E3EQC (accessed on June 2021)
22. The Amazon Effect: Dynamic Pricing Done Right Available online: https://www.pragmaticinstitute.com/resources/articles/product/the-amazon-effect-dynamic-pricing-done-right/ (accessed on June 2021)
23. An Introduction to Amazon Pricing in 2021 Available online: https://www.omniaretail.com/blog/an-introduction-to-amazon-pricing (accessed on June 2021)
24. TF IDF from Monkeylearn https://monkeylearn.com/blog/what-is-tf-idf/ (accessed on June 2021)

AUTHORS PROFILE

Sagar Yeruva, Associate Professor in the Department of CSE (AIML and IoT) in VNRVJIET, Hyderabad, India. He has done a Ph.D. in CSE from JNTU Hyderabad and has 19 Years of teaching experience in the field of engineering. He has authored 30 research publications in various journals/conferences. His areas of research include Data Bases, Data Mining, Machine Learning, and Distributed Computing

Addi Sathvika, Pursuing B. Tech in CSE from VNR VJIET, Hyderabad, India. She is currently working as an intern at Experian. She is an active member of the Computer Society of India.

Damera Sruthi, Pursuing B.Tech in CSE from VNR VJIET, Hyderabad, India. She is currently working as an intern at UnPlatforms. She holds a position on the core committee of the Computer Society of India, VNRVJIET chapter.

Duggasani Yaswanth Reddy, Pursuing B.Tech in CSE from VNRVJIET, Hyderabad, India. He is an active member of the Computer Society of India.

Gogineni Gopi Krishna, Pursuing B.Tech in CSE from VNRVJIET, Hyderabad, India.

Retrieval Number: 100.1/ijrte.D73311111422
DOI: 10.35940/ijrte.D7331.1111422
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)