International Journal of Innovative Research in Engineering & Management (IJIREM)
ISSN: 2350-0557, Volume-10, Issue-1, February 2023
https://doi.org/10.55524/ijirem.2023.10.1.8
Article ID IJIR-1309, Pages 39-42
www.ijirem.org

Content Based Apparel Recommendation Engine

Mudasir Ahmad Hurrah[1], Dr Monika Mehra[2], and Ravinder Pal Singh[3]

[1]M. Tech Scholar, Department of Electronics and Communication Engineering, RIMT University, Mandi Gobindgarh, Punjab, India
[2]Professor, Department of Electronics and Communication Engineering, RIMT University, Mandi Gobindgarh, Punjab India
[3]Technical Head, Department of Research, Innovation and Incubation, RIMT University, Mandi Gobindgarh, Punjab India

ABSTRACT- With the rapid rising of living standard, people gradually developed higher shopping enthusiasm and increasing demand for garment. Nowadays, an increasing number of people pursue fashion. However, facing too many types of garment, consumers need to try them on repeatedly, which is somewhat time and energy-consuming. Besides, it is difficult for merchants to master the real- time demand of consumers. Proposed recommendation engine utilizes tops_fashion dataset which consists nearly 183k products acquired through Amazon API interface and processes this dataset using multiple techniques like *Bag_of_Words (BoW), tf, idf, Word2Vec model,VGG16 (CNN)* etc to recommend similar items to a given query item. Product Advertising API acts as a gateway to Amazon's databases so that we can take advantage of Amazon's sophisticated e-commerce data and functionality. In this project, we are using Python language for coding. The proposed system can recommend product based on various features of the product such as title, color, brand, price, image but we have only used title and image features of the products. Collaborative Filtering technique suffers from cold start problem- a situation where a recommender does not have adequate information about a user or an item in order to make relevant predictions. This is one of the major problems that reduce the performance of recommendation system.

KEYWORDS- Apparel, Recommendation, TF-IDF, Bag-of-Words, Word2Vec Model,VGG16 (CNN**),** Content-Based-Filtering.

I. INTRODUCTION

E-commerce sites contain information about millions of products in their databases. Most often the users interacting with such large catalog of items does not get what they are looking for and it is very challenging for these companies to recommend appropriate items to a customer leading to the evolution of recommender system technology. With the acceptance of web as a mode for e-transactions and the exponential increase in amount of data generated by automated systems, various ecommerce companies like Amazon, Myntra, eBay etc. have done a lot of research work in the field of recommender system technology. According to recent estimates Amazon generates 35% of revenue using product recommendations. These companies use various prediction mechanism based on preferences given by users in the past to recommend appropriate and attractive items to the users by incorporating recommender systems in their websites. With the advancement in the field of machine learning (ML) technology, recommendation engines are commonly used by the companies that deal with apparels. These systems recommend to the user certain items that they think the user will be interested in or that might have been difficult to find using conventional search techniques. The main aim of the researchers is to recommend appropriate items to specific users based on the rules that are mined from design knowledge and the experience of an expert with the recommendation system. Customized item suggestions are the elective method for exploring through the online shop. A recommender system is an information filtering system i.e., a system which removes redundant information from very large data corpus using algorithmic techniques to make information seeking fast and easy and to provide optimized items to the user. The effect of adopting a recommender system in a company‘s online presence can be classified into two different effects the direct effect and the indirect effect on revenue. The direct effect consists of all revenue from purchases made in recommended items. The indirect effect is considered as all the revenue that originates from repeat purchases of recommended items and the amount of purchases made in the same category as the recommended items. The products can be recommended based on the top overall sellers on a site, on the demographics of the consumer, or on an analysis of the past buying behavior of the consumer as a prediction for future. Four fundamental approaches to recommendation can be mentioned: demographic filtering, collaborative and content-based recommendation, and simplified statistical approaches. As in our project we are dealing with Amazon‘s data set, so we will focus on approaches used by Amazon for recommendation. Internally Amazon uses two sources of data to do product recommendation:

1. Content-based filtering
2. Collaborative filtering

Content-based filtering techniques utilize a set of discrete properties or description of an item in order to recommend items with similar characteristics. The description or properties of an item may include title, brand, color, image, price, review etc [1]. Content-based recommendation systems are used in a various domain like: webpages, news articles, hotels & restaurants, movies, apparels, people etc. Content-based filtering techniques create item profiles

which contain a set of features. It is convenient to think item profiles as vectors (Boolean or Real valued). A profile may contain features like:

- Movies: author, title, actor, director etc.
- Images, Videos: metadata, tags etc.
- People: Set of friends.
- Apparels: Brand, product type, color, price etc.

Content-based recommendations often require text processing algorithms, since the content of the items first has to be analyzed and transformed in such a way that it is possible to find similarities between items. Content-based recommendation is basically saying, if you are looking at a white dotted shirt, there is a very high chance that you will also like another shirt which is also like white dotted shirt but properly different shirt may be of different brand. So it is using the TEXT and IMAGE description of product to recommend. Collaborative filtering approaches build a model from a user's past behavior as well as similar decisions made by other users. This model is then used to predict items that the user may have an interest in. for example A consumer C1 who has seen items in order I1, I2 and I3 and second user C2 checked out items I1, I3 and I4. If a new consumer C3 who is checking out I1, since I3 was checked out by both users C1, C2 after I1. So there is a very high chance that C3 will be interested in I3. Therefore, I3 can be recommended because other users who came before C3 typically went and spent some time on product I3. Collaborative filtering methods are based on collecting and analyzing a large amount of information on users behaviors, activities or preferences and predicting what users will like based on their similarity to other users. Collaborative filtering is based on the assumption that people who agreed in the past will agree in the future, and they will like similar kind of items as they preferred in the past. But unfortunately we don't have this data, it is highly guarded by Amazon. In our project we will use Content-based recommendation technique.

II. LITERATURE REVIEW

Bobadilla J et al(2013)[3]:Content-based technique is a domain-dependent algorithm and it emphasizes more on the analysis of the attributes of items in order to generate predictions. In content-based filtering technique, recommendation is made based on the user profiles using features extracted from the content of the items the user has evaluated in the past.

Friedman N et al(2014)[4]: In Content-based technique items that are mostly related to the positively rated items are recommended to the user. CBF uses different types of models to find similarity between documents in order to generate meaningful recommendations. It could use Vector Space Model such as Term Frequency Inverse Document Frequency (TF/IDF) or Probabilistic models such as Naïve Bayes Classifier.

Burke R et al(2007)[5]:Collaborative Filtering technique suffers from cold start problem. This refers to a situation where a recommender does not have adequate information about a user or an item in order to make relevant predictions .

This is one of the major problems that reduce the performance of recommendation system. The profile of such new user or item will be empty since he has not rated any item; hence, his taste is not known to the system.

Park DH et al(2012)[6]: Collaborative Filtering technique suffers from Data sparsity problem. This is the problem that occurs as a result of lack of enough information, that is, when only a few of the total number of items available in a database are rated by users . This always leads to a sparse user item matrix, inability to locate successful neighbors and finally, the generation of weak recommendations. Also, data sparsity always leads to coverage problems, which is the percentage of items in the system that recommendations can be made.

Park DH et al(2012)[6]: Collaborative Filtering technique suffers from Scalability. This is another problem associated with recommendation algorithms because computation normally grows linearly with the number of users and items. A recommendation technique that is efficient when the number of dataset is limited may be unable to generate satisfactory number of recommendations when the volume of dataset is increased. Thus, it is crucial to apply recommendation techniques which are capable of scaling up in a successful manner as the number of dataset in a database increases. Methods used for solving scalability problem and speeding up recommendation generation are based on Dimensionality reduction techniques,such as Singular Value Decomposition (SVD) method, which has the ability to produce reliable and efficient recommendations.

Su X et al(2009)[7]: Collaborative Filtering technique suffers from Synonymy .Synonymy is the tendency of very similar items to have different names or entries. Most recommender systems find it difficult to make distinction between closely related items such as the difference between e.g. baby wear and baby cloth. Collaborative Filtering systems usually find no match between the two terms to be able to compute their similarity. Different methods, such as automatic term expansion, the construction of a thesaurus, and Singular Value Decomposition (SVD), especially Latent Semantic Indexing are capable of solving the synonymy problem. The shortcoming of these methods is that some added terms may have different meanings from what is intended, which sometimes leads to rapid degradation of recommendation performance.

III. OBJECTIVE

The objective of the project is as follows:

- It Provides User Independence. It exploits solely ratings provided by the active user to build his/her own profile.
- It Provides Transparency. Explanations on how the recommender system works can be provided by explicit listing content feature or descriptions that caused an item to occur in the list of recommendations.
- It Solves Cold Start Problem. It recommends items not yet rated by any user.

IV. METHODOLOGY

Proposed system implements Content-based filtering technique which suggests top similar products based on the description of the query product using multiple techniques like Bag_of_Words (BoW), tf_idf, Word2Vec model etc. Proposed recommendation engine utilizes _tops_fashion'

dataset which consists nearly 183k products acquired through Amazon API interface and processes this dataset using techniques mentioned above to recommend similar items to a given query item. Product Advertising API acts as a gateway to Amazon's databases so that we can take advantage of Amazon's sophisticated e-commerce data and functionality. The proposed system can recommend product based on various features of the product such as title, color, brand, price, image but we have only used title and image features of the products.The recommender system general model have been shown in Figure 1.

Strengths of the proposed system are:

1. Unlike Collaborative Filtering, if the items have sufficient descriptions, we avoid the new item problem.
2. Content representations are varied and they open up the options to use different approaches like: text processing techniques, the use of semantic information, inferences, etc.
3. It is easy to make a more transparent system: we use the same content to explain the recommendations.

Algorithms to be used

1. Bag Of Words
2. TF-IDF
3. Word2Vec Model
4. IDF

Proposed evaluation scheme

1. Bag Of Words
2. TF-IDF
3. Word2Vec Model
4. IDF

Term Frequency (TF) = (Number of times word w appears in a Document) ÷ (Total number of words in the document.)

IDF = $\log_e$ (Total number of documents in the corpus) ÷ (Number of documents containing word 'w')

TFIDF = TF (W, Ti) * IDF (W, D)

$$\text{Precision} = \frac{\text{Correctly Recommended Items}}{\text{Total Recommended Items}}$$

$$\text{Recall} = \frac{\text{Correctly Recommended Items.}}{\text{Total useful Recommended Items.}}$$

Hardware & Software requirements:

➢ ***Hardware***

- Processor: Core I5
- RAM: 4GB
- HDD: 250GB

➢ ***Software***

- Operating system: Ubuntu/windows
- Python 3.6
- Various Packages Such as Anaconda, Jupitor and pip etc.

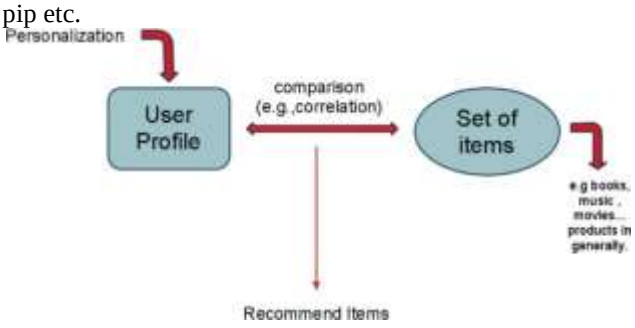

Figure 1 : Recommender system general model

V. SYSTEM IMPLEMENTATION AND AQUISITION

A. *Python and relevant Packages Install*

In our project, we are using Python language for coding. So, first of all we have to download and install Python (3.6) on our system. To check whether Python installation is successful or not open terminal / command prompt and type the following command python or python version. Next, we need to download a tool called Anaconda. Anaconda is a free and open source distribution of the Python and R programming languages. The anaconda window is shown below in Figure 2.

For data science and machine learning related applications (large-scale data processing, predictive analytics, scientific computing), that aims to simplify package management and deployment. One thing that Anaconda does is, when you install Anaconda, it automatically install a package called IPYTHON NOTEBOOK – Jupyter Notebook. Jupyter Notebook is a GUI on which we write code. It is one of the most popular user interfaces for Data sciences, Machine Learning. The IPython Notebook is now known as the Jupyter Notebook. It is an interactive computational environment, in which you can combine code execution, rich text, mathematics, plots and rich media.

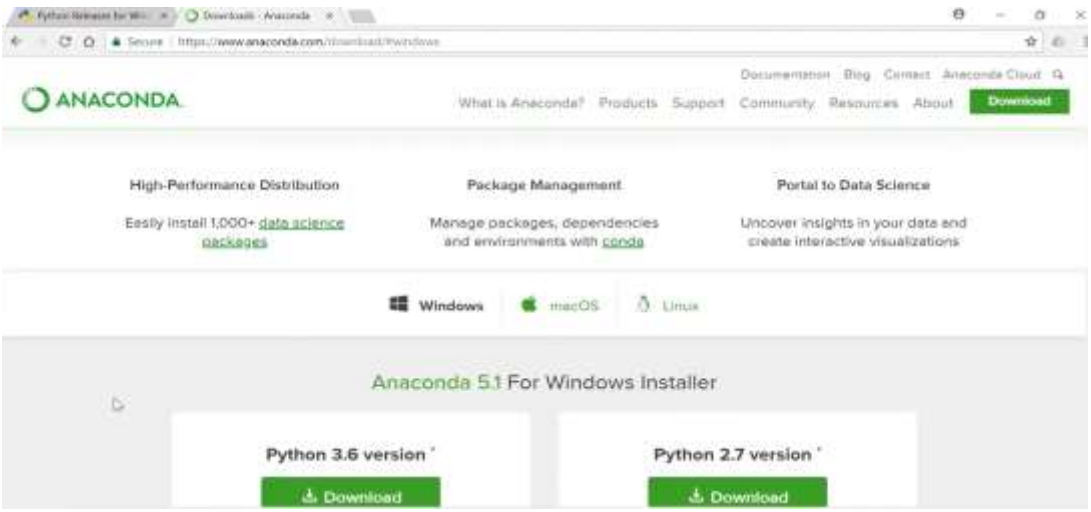

Figure 2 : Anaconda Tool Window

VI. DATA AQUISITION

We got data from Amazon itself in a policy compliant manner. We actually used Amazon‘s Product Advertising API to get data. Amazon Product Advertising API is basically an API offered by Amazon for accessing data on every product that are being displayed at any Amazon site. Product Advertising API operations open the doors to Amazon's databases so that you can take advantage of Amazon's sophisticated e-commerce data and functionality [2]. We get a data set of about 183000 products which is in JSON format and for each product we obtain all the interesting features / properties like image u r l , product description etc. JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is derived from the JavaScript language. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language.

Just like most of the API that are offered by sites like Facebook, Twitter, we need to create an access key to be used by our application. The access key that was going to create can be used to authenticate requests to any Amazon Web Service API and that includes the Product Advertising API.

VII. CONCLUSION

Recommender systems are a powerful new technology for extracting additional value for a business from its customer databases. These systems help customers to find products they want to buy from a business. Recommender systems benefit customers by enabling them to find products they like. Conversely, they help the business by generating more sales. Unlike collaborative filtering, content-based recommending holds the promise of being able to effectively recommend unrated items and to provide equality recommendations to users with unique, individual tastes. Collaborative methods are best at recommending reasonably well known items to users in communities of similar tastes when sufficient user data is available but effective content information is not. Content-based methods are best at recommending unpopular items to users with unique tastes when sufficient other user data is unavailable but effective content information is easy to obtain. In this project we will presented and experimentally evaluate various algorithmic choices for Content-based recommender systems.

VIII. FUTURE WORK

Since no system in the world is complete and only time can prove its incompleteness, same is the case with this system. Since it is an academic project, there is a lot of scope for this project in the future. Some of the future enhancements include:

- Integrate the recommendation engine trained on our own dataset on ecommerce website.
- We want to recommend those products/ apparels to the consumers that are common in the output of implemented algorithms (BOW, TF-IDF, WORD2VEC and VGG16).
- Train one of the implemented models (VGG16) on our own data.
- Recommend apparels based on color and price features.
- We want to combine Content-based technique with Collaborative-based technique in order to improve performance of our system.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] Charu C. Aggarwal, "Recommender Systems The Text Book," in Recommender Systems.: Springer, 2016, ch. 1, p. 498.
[2] Amazon. (2013, Aug.) Amazon. [Online]. http://amazon.com
[3] Burke R. Hybrid recommender systems: survey and experiments.User Model User-adapted Interact 2002;12(4):331–70.
[4] Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers. Mach Learn 2014;29(2–3):131–63.
[5] Burke R. Web recommender systems. In: Brusilovsky P, Kobsa A, Nejdl W, editors. The Adaptive Web, LNCS 4321. Berlin Heidelberg (Germany): Springer; 2007. p. 377–408. http://dx.doi.org/10.1007/978-3-540-72079-9_12.
[6] Park DH, Kim HK, Choi IY, Kim JK. A literature review and classification of recommender systems research. Expert Syst Appl 2012;39(11):10059–72.
[7] Su X, Khoshgoftaar TM. A survey of collaborative filtering techniques. Adv Artif Intell 2009;4:19.