International Journal of Advanced Technology in Engineering and Science
Vol. No.4, Issue No. 08, August 2016
www.ijates.com
ijates
ISSN 2348 - 7550

CONTENT BASED RECOMMENDATION USING MACHINE LEARNING

Vinay Chandragiri

Department of Computer Science

Indian Institute of Technology Guwahati, Guwahati, (India)

ABSTRACT

This paper discusses and presents a model about content-based recommendation systems. These are the systems which recommend an item to the user based upon a description of the item along with the profile of a user's interests or past activities. It also details about a Machine Learning model as an application over e-Commerce data. Machine Learning is the act of getting computers/machines to act or to learn themselves from known representations without being explicitly programmed to the task. In the past decade, Machine Learning has given us solutions to understanding of Natural Language, Image Recognition, Automated cars, effective search and many more. Content-based Recommendation systems can be used in various types of domains, from recommending music, news related articles, food places, e-Commerce products and items for sale. The inner details of systems differ, but content-based recommendation systems share a common means for item description which may be recommended, for managing user profile which is based on his/her's past activities, likes etc. The profile is often updated dynamically in response to feedback on the of items that which have been presented to the user.

Keywords - Content-Based, Recommendation, Machine Learning, E-Commerce

1. INTRODUCTION

Content based recommendation systems are those which are built using actual properties of item such as description, title, price, url, colour etc. A standard, use case scenario for latest recommendation systems is an application with a user interface that will help user to interact with. Typically, a system presents a list of items to a user, and that particular user selects one, from the items to get more information about an item by clicking it. For example, news websites showcase web pages with important news (Headlines), story summaries which allows the user to select one and read the whole story associated with it. E-commerce websites often showcase a webpage with a list of products and then permit the user to see more information about product which they select in order to purchase that particular product. The execution is done in this way, the web server sends HTML and the user sees a web page. Typically, the web server has a database of products/items and dynamically builds web pages with a list of them. There are often many things available in a database than the number of those that would comfortably fit themselves on a web page, it is essential to select a subset of these to showcase to the user or to determine a proper order in which to present the items.

Content-based recommendation systems analyze descriptions of a particular item in order to identify items that are of particular interest to the user. The details of recommendation systems differ based on the representation of items.

II. WHEN DO WE USE CONTENT BASED RECOMMENDER SYSTEM ?

There are two major types of recommendation systems namely Content based and Collaborative Filtering based. The idea behind which Collaborative Filtering works is pretty simple, the product which some user most likely to purchase is the same product that a group of users like you also bought. Pure Collaborative systems have no proper knowledge or informations about the products that they are recommending which is the major difference between this system and a content based recommender system.

When we don't know anything about the user so we can't build any database of his/her purchases to recommend using, Collaborative Filtering isn't a viable option. Here we can use content based recommender system to recommend products which is a start to the Collaborative Filtering systems have.

III. DATASET

The data which is used here to, develop this content based engine is provided by Kaggle as a part of its Open Data Platform. It consists of actual SKUs from an outdoor apparel brand's product catalog. This is a bit rare to get a product level data in a real-world format. This is also very useful for testing things like recommendation engines.

3.1 Representation of Data

Items in the dataset have an ID with a text - description about the product. The ID is the unique identifier and the database consists of 'rows' with products and their description in a table. Each item is described using a text. There is only one attribute 'description' for each item. This data is a structured dataset.

Many domains are represented in a best way by semi-structured data in which there are attributes with a set of confined values and also with complementary text fields. A typical way to deal with these free text fields is to convert that particular text to a well structured one. For example, each word can be seen as an attribute, with a flag indicating whether the word is present in the article or with its frequency pertaining to that article.

Example

ID	Description
10	Baby sun bucket hat - This hat goes on when the sun rises above the horizon….

IV. RECOMMENDATION ENGINE

In this model we use a traditional Information Retrieval technique called TF-IDF abbreviated as Term Frequency - Inverse Document Frequency in order to parse through the descriptions of products and also to identify phrases, so as to find similar items based on the description.

Before using the model, many personalized systems that deal with free text use a procedure to create a structured representation that starts with text based search systems. Rather than using words directly, the root forms of words are commonly used and this process is called stemming. The ultimate goal of this process is to create a word that reflects the typical meaning behind words such as “purple,” “purplish,” and “purply.” The value of a particular variable is that which is associated with a term is a real number which presents the relevance.

This value is called the tf*idf weight. The tf*idf weight is represented using “w(t,d)”. This values of a term t in a document d is a function of the frequency of t in the document (tf,d), the number of documents that contain the term (df) and the number of documents in the total collection (N). TF-IDF scans at one, two, and three-word phrases that appear multiple times in given description (the "term frequency") and divides them by the number of times those phrases are present in *all* other item descriptions in the database. Therefore, terms that are more specific to a particular product get a higher score, and terms that appear often, but also appear in other items get a lower score.

After obtaining, the TF-IDF terms and scores for each product, we will use a measurement to measure ‘similarity’ which helps us to identify which products/items are 'closer' to each other. This is similar to finding nearest neighbours. There are many different ways to measure similarity. I have considered using cosine-similarity in the model.

5. RESULTS and CONCLUSION

We train the engine on the given data by constructing a TF-IDF matrix of unigrams, bigrams and trigrams for each product. We also tell the TF-IDF module to ignore the common english words such as ‘the’ etc.
Then we compute the similarity between all products using cosine similarity. We then iterate through each item’s similar items to obtain the required number of most similar items from the data for that particular product. We then store the similarities and their scores for each item in a sorted set.
When the unique ID of a product is given, we then retrieve the similar items for that particular product and their scores from the sorted set.

A large variety of Machine learning algorithms have been developed and adapted to learn user behaviour over large group of websites say e-Commerce, Medicine etc, and the choice of the algorithm depends on the representation of type of content. So is the reason, content based recommendation systems play a major role in business at present. The following is the example from the above described model,
Example:

The 2 similar products to Baby sunshade top obtained using the model are as follows:
Sunshade hoody (score:0.21)
Baby baggies apron dress (score:0.109)

REFERENCES

[1] Billsus, D., Pazzani, M.: Learning Collaborative Information Filters. In: Proceedings of the International Conference on Machine Learning. Morgan Kaufmann Publishers. Madison, WI (1998) 46-54 7.

[2] Billsus, D., Pazzani, M., Chen, J.: A Learning Agent for Wireless News Access. In: Proceedings of the International Conference on Intelligent User Interfaces (2002) 33-36 8.

[3] Cohen, W.: Fast Effective Rule Induction. In: Proceedings of the Twelfth International Conference on Machine Learning, Tahoe City, CA. (1995) 115-123

[4] Kaggle - Open Data Platform https://www.kaggle.com/datasets