Hybrid of Topic Modelling & Collaborative Filtering in Recommender Systems

Neelima Potharaj
Information of Technology
University of Washington -Tacoma
Tacoma, USA

***Abstract*—This paper focuses on testing the use of tf-idf model on LDA model based collaborative filtering with varying values for nearest neighbor's computation. The impact of number of topics on recommendation is also studied.**

Keywords—recommender systems; collaborative filtering; topic modelling; Bag-of-Words; LDA; TF-IDF; Cosine Similarity; Pearson Correlation;

I. INTRODUCTION

Social Consumer Network based websites allow their users to present their personalities, ideologies, or opinions online through the form of likes, photos, ratings, reviews/comments and tips, creating a unique online presence for themselves. The rise of recommendation systems was in tune with this rise in availability of data on users' preferences.

Recommendation systems for customer driven businesses such as Yelp, Amazon, and Netflix, cut the computational costs of users searching through catalogs and simultaneously drive revenue by recommending products based on user's preferences [2]. These systems rely heavily on explicit data, such as visiting the product or venue's page, purchase of the product (such as Amazon), and ratings of their products or venues. Implicit content from data such as user reviews/comments/tips are usually used in combination with one or more of the explicit data when building a recommender system.

Collaborative filtering (CF) is a widely-used technique in creation of recommender systems. CF is based on the assumption that users who have agreed in the past will agree in the future [3]. It involves using available ratings to predict missing ratings to products based on the similarity between users or similarity between products users previously rated.

However, as ratings are not good enough representation of a user's preferences or in some cases when ratings are not available, studies started to use textual data from user provided reviews to adjust the previous ratings or predict new ratings. Previously studied recommendation systems which use reviews datasets focus on using sentiment analysis of text to predict rating values [4]. Moreover, rather than using basic statistical analysis most research studies have used advanced machine learning techniques on top of statistical analysis of text [3]. Machine learning algorithms are computationally expensive, thereby in this study the aim is to keep the computational expenses low and use simple techniques to build a recommendation system as efficient.

The focus of this study is to evaluate the use of only implicit data and basic statistical analysis to build a recommender system for recommending attractions in a city. The implicit data in this case is the tips dataset from FourSquare.com [5, 6, 7].

In this paper, I propose the use of topic modelling of tips dataset for recommending attractions to users. A topic classification for each user-attraction document (tip) would be used to calculate similarity between users. KNN algorithm will be used to reduce the set of similar users and thereby improve the recommendation accuracy. To improve the accuracy of the topic modelling, TF-IDF based corpus is used as input to the topic modelling algorithm in place of "bag-of-words" based corpus.

II. BACKGROUND

A. TF-IDF

Term Frequency – Inverse Document Frequency (TFIDF) is a most popular term weighting scheme [8]. The term frequency is simply the frequency of a term in a document. The IDF part of the scheme is the inverse document frequency, it is frequency of this term found in the document in comparison with its occurrences in the whole document collections (or corpus) [8]. Figure 1 presents the formula for this scheme.

$$tf_{i,j} = f_{i,j}$$

$$IDF = log_2\left(\frac{number\ of\ documents\ in\ the\ corpus}{number\ of\ documentss\ that\ contain\ the\ term\ i}\right)$$

Fig 1. TF-IDF Formula [8]

This scheme is used to measure the importance of term in terms of the corpus. This method is often used by search engines in measuring the relevance of a document given the query terms.

B. Bag of Words

Bag of Words model is a way to represent a document/collection of documents in a numerical vector format. Each document in a corpus would be represent with a vector with each feature (or term, in case of 1-gram) and the

frequency of this feature occurring in the given document. Features can be a single word, lemmatized word, or more than 1 word tokens [9]. In topic modelling, documents are transformed to either tf-idf or bag-of-words representation before being inputted

to a topic model. The main difference between tf-idf and bag of words models is that one presents the relevance of each term in terms of the corpus represented as weights, whereas the other considers the occurrence of each term represented as frequencies, respectively.

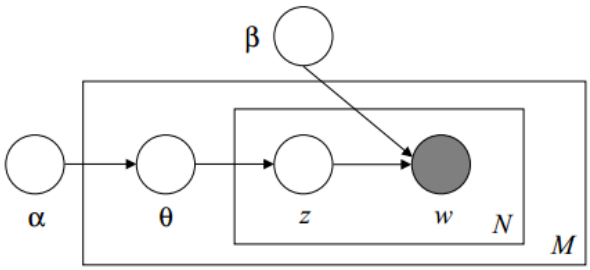

Fig 2. Graphical Representation of the LDA model [10]

C. LDA

Latent Dirichlet Allocation (LDA) is a generative probabilistic model representing each document as a mixture of topics where each topic has mixture words their probabilities associated with it.

The LDA representation has 3 levels as depicted in Figure 2. The parameters α and β are the corpus level parameters, the θ_d are the document level variables, the variables z_N and w_N are word-level variables [10]. α and β are sampled once to generate the corpus, θ_d are sampled once per document and, z_N and w_N are sampled once for each word in each document.

LDA model for text mining usually is used with a bag-of-words representation as an input. In this research, LDA model's efficiency is measured with the use of tf-idf model as an input in comparison with bag-of-words as input

III. Related Work

The research by Shengli Xie and Yifan Feng about scenic spot recommendations using LDA and collaborative filtering [11]. This article supports the idea that collaborative filtering alone is not the best recommendation system as it only considers behavior. Collaborative filtering model with the use of topic distribution results from reviews by users for different venues provides better results [11]. The research by Xie and Yifan is an inspiration of this new algorithm presented in this paper. This paper further evaluates the usage of TF-IDF corpus data set in place of bag of words data set as input for the LDA model, in terms of how it affects the recommendation outcomes.

The research by Xie and Yifan is an inspiration of this new algorithm presented in this paper. This paper further evaluates the usage of TF-IDF corpus data set in place of bag of words data set as input for the LDA model, in terms of how it affects the recommendation outcomes

IV. Methodology

A. Dataset

Dataset used is the NYC Restaurant Rich Dataset which contains checkins, tips and tags data; only the tips data is used for this research [12]. This data is collected from FourSquare.com from 24th October 2011 to 20 February 2012, which contains 3112 users and 3298 venues with 10377 tips [12].

B. Data Preprocessing

The data is stored by rows in csv file, where each row contains user id, venue id and a comment. Two dictionaries were created from this data: one for users and one for venues, where a key in each corresponding dictionary is the respective user id or venue id. A single list of all comments irrespective of the venue or user id information is created.

For evaluation purposes, the dataset is split into training and testing set [13]. All users that have commented only once are not used for the testing set. For the rest of the users, half of the comments and their respective venue ids are stored in a dictionary, with user ids as the keys and the value being another dictionary containing the venue ids and comments.

From the given list of comments a dictionary is created, where a word is the key and its frequency of occurrence in the whole comments dataset is the value. Prior to the creation of this dictionary, all comments are tokenized into their respective vectors and *stopwords* are removed from these vectors. Next, the dictionary and word vectors are used to create a corpus where each comment is represented with a vector of vectors where each unique word from the comment is given an id and frequency of occurrence in the given comment.

C. Topic Modelling and Matrix Factorization

After the processing the data, the corpus is transformed onto a TF-IDF space using the genism package's TF-IDF model. The frequency value present in the previous representation of corpus would be replaced by weighted frequency under the TF-IDF model. The LDA model used for this research is also from the genism package. Either the TF-IDF corpus or the original corpus along with the dictionary of words is loaded into the LDA modelling package. The topic distribution for a single user is acquired by transforming its individual comments vectors using LDA. The topic with the highest probability of occurrence in the user's comments would be selected. Using this topic id, a sparse user-venue matrix (P) is created where each comment is represented with the corresponding distribution value for the selected topic. SVD is performed over the P matrix to fill in the missing gaps for better similarity calculations.

D. Similarity and KNN

As the P matrix is sparse, use of corrected cosine similarity or Pearson correlation algorithm is appropriate as it will

consider the missing values and normalize the results accordingly. A "users x selected user" similarity vector (S) is created using the P matrix. K-nearest algorithm and is run on the S vector to get top K similar users to the selected user. An aggregated list of all venues from the top K users (other than the ones the selected user commented for) is made.

This list of venues is compared with the hidden venues from the testing set for the selected user. The number of venues in the list and the matches made between this list and the hidden venues from the testing set is recorded.

The same steps of modelling and recommending is performed for rest of the users in the testing set. The values from performance on each user are aggregated to be used for precision and recall calculations.

V. Evaluation

For each user id in the testing set, k nearest neighbors (or k users most like the given user id) are found and a set of all the venues that these neighbors visited is made. Venues which are contained in the training set of the user id are removed from the recommended venues list.

Once the recommended venues set is formed it compared with the venues present under the user id in the testing set, and number of venues common in both sets is recorded along with the number of recommended venues and number of venues present under the user id in the testing set. For each user id in the testing set, the precision and recall values are calculated after the recommended venues comparison. At the end of all iterations on testing set the averages of precision and recall are calculated. This testing is done on different k values, to study its impact on efficiency of the recommendation system.

For 100 topics, both precision and recall with tf-idf modelling and with tf-idf modelling followed a similar trend (Figure 3) (Figure 4). One common trend among both precision and recall plots for all 3 topics size variations is increase in recall with decrease in precision, however around K=5 and below there is increasing trend in both precision and recall (Appendix Figure 7-10). From these plots, it is evident that tf-idf alone doesn't have much impact on the recommendations.

Topwords selection's impact on the algorithm is also tested. From the dictionary, words with at least 20 occurrences and words occurring in lower than 90% of the whole corpus were filtered. The resulting values from the calculation on the transformed dataset (without tf-idf modelling) with 100 topics are presented in Figure 9 and Figure 10. According to the plots, dictionary filtering for topwords lowered the precision and recall for varying K values. This shows data preprocessing step can have significant positive or negative impact on the recommendations.

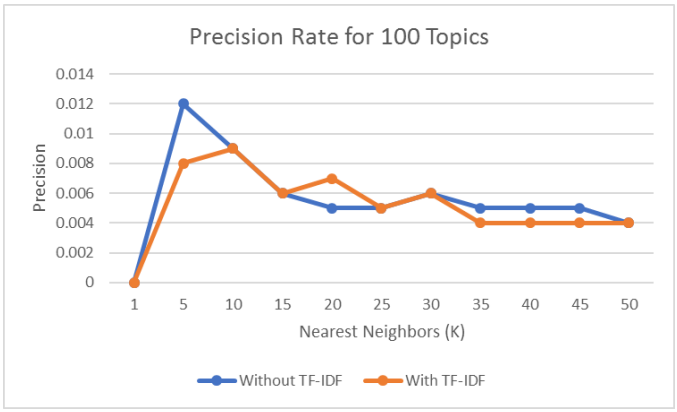

Fig. 3 Precision results when using 100 topics and using tf-idf modelling with varying K values.

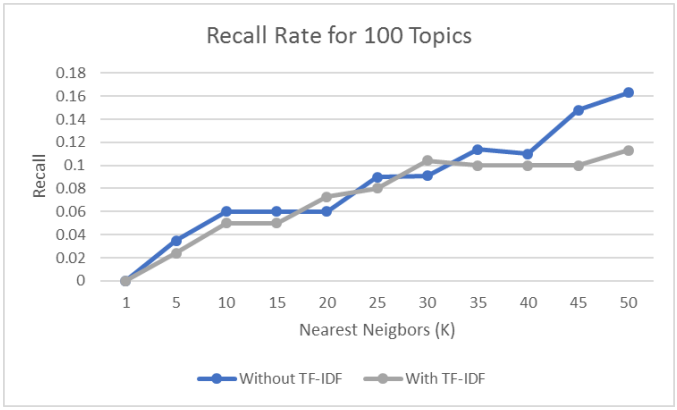

Fig. 4 Recall values when using 100 topics for LDA modelling and using tf-idf modelling with varying K values.

VI. Discussion

LDA is a probabilistic model and it does not consider syntax when modelling the topics and topic/word distribution. Moreover, it is computationally expensive to update a LDA model and inefficiencies arise with the model at adjusting topics with each update [14]. However, these flaws do not have as big of an impact on this algorithm to a certain extent as SVD of the user-venue fills in the gaps of the sparse user-venue matrix and helps in predicting the possible topic distribution offline. The problem occurs with the fact that changing user behavior is not considered with this algorithm, the same set of recommendations would be repeated.

Use of a better LDA model, improving the data preprocessing step could significantly improve this algorithm. Collaborative filtering based on rating values could be used to make better judgement about the efficiency of this algorithm. Use of different computational packages with data mining step like for functions such as similarity computation, SVD computation could also improve the efficiency of the recommendation system.

APPENDIX

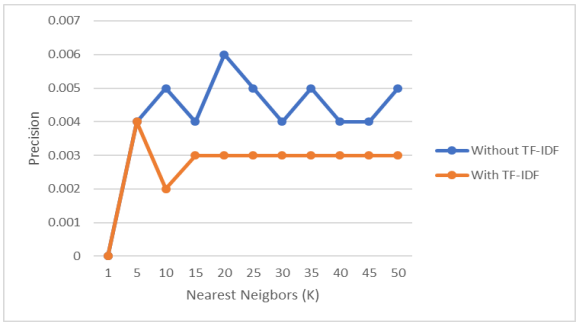

Fig. 5 Precision Rate for 10 Topics

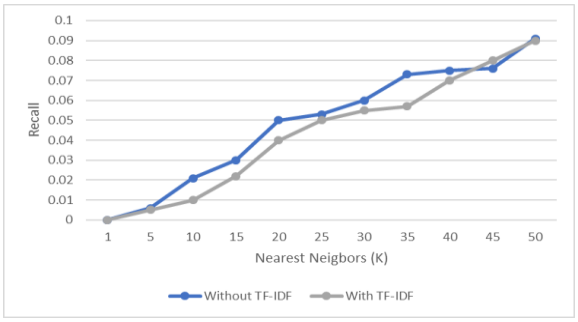

Fig. 6 Recall Rate for 10 Topics

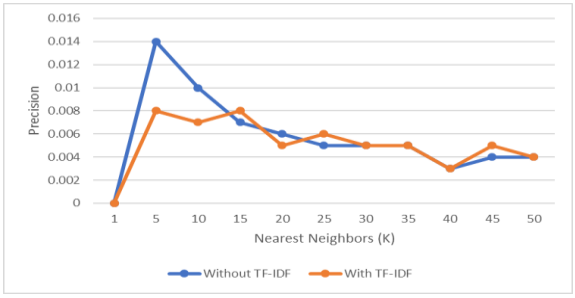

Fig. 7 Precision Rate for 500 topics

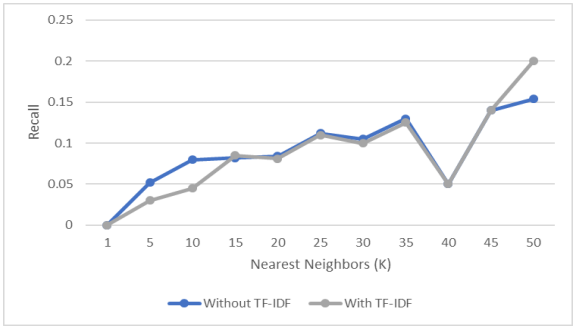

Fig. 8 Recall Rate for 500 topics

Fig. 9 Precision Rate Comparison on Topwords Selection

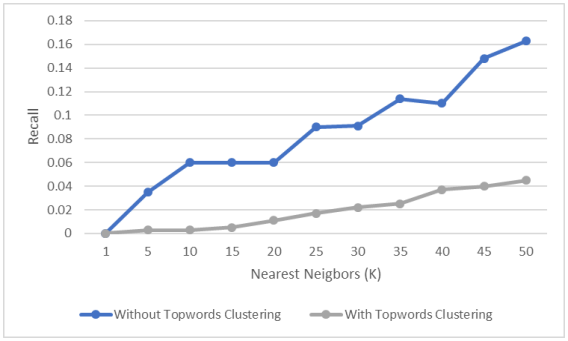

Fig. 10 Recall Rate Comparison on Topwords Selection

REFERENCES

[1] S. Xie and Y. Feng, "A Recommendation System Combining LDA and Collaborative Filtering Method for Scenic Spot," *2015 2nd International Conference on Information Science and Control Engineering*, Shanghai, 2015, pp. 67-71.

[2] F.O. Isinkaye, Y.O. Folajimi, B.A. Ojokoh, Recommendation systems: Principles, methods and evaluation, Egyptian Informatics Journal, Volume 16, Issue 3, November 2015, Pages 261-273, ISSN 1110-8665.

[3] Sindhu Raghavan, Suriya Gunasekar, and Joydeep Ghosh. 2012. Review quality aware collaborative filtering. In *Proceedings of the sixth ACM conference on Recommender systems* (RecSys '12). ACM, New York, NY, USA, 123-130.

[4] Guang Ling, Michael R. Lyu, and Irwin King. 2014. Ratings meet reviews, a combined approach to recommend. In *Proceedings of the 8th ACM Conference on Recommender systems* (RecSys '14). ACM, New York, NY, USA, 105-112.

[5] Dingqi Yang, Daqing Zhang, Zhiyong Yu and Zhiwen Yu, Fine-Grained Preference-Aware Location Search Leveraging Crowdsourced Digital Footprints from LBSNs. In Proceeding of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2013), September 8-12, 2013, in Zurich, Switzerland.

[6] Dingqi Yang, Daqing Zhang, Zhiyong Yu and Zhu Wang, A Sentiment-enhanced Personalized Location Recommendation System. In Proceeding of the 24th ACM Conference on Hypertext and Social Media (HT 2013), 1-3 May, 2013, Paris, France.

[7] Dingqi Yang, Daqing Zhang, Zhiyong Yu, Zhiwen Yu, Djamal Zeghlache. SESAME: Mining User Digital Footprints for Fine-Grained Preference-Aware Social Media Search. ACM Trans. on Internet Technology, (TOIT), 14(4), 28, 2014.

[8] D. Kraft and E. Colvin, Fuzzy information retrieval, 1st ed. San Rafael, California: Morgan & Claypool Publishers, 2017, pp. 52-58.

[9] "4.2. Feature extraction — scikit-learn 0.18.1 documentation", Scikitlearn.org, 2017. [Online]. Available: http://scikitlearn.org/stable/modules/feature_extraction.html#the-bag-of-wordsrepresentation. [Accessed: 23- May- 2017].

[10] D. Blei, A. Ng and M. Jordan, “Latent Dirichlet Allocaiton”, Journal of Machine Learning Research 3, pp. 993-1022, 2003

[11] S. Xie and Y. Feng, "A Recommendation System Combining LDA and Collaborative Filtering Method for Scenic Spot", 2015 2nd International Conference on Information Science and Control Engineering, no. 2015, pp. pp.67-71, 2015.

[12] D. Yang, D. Zhang, Z. Yu, and Z. Yu, “Fine-grained preference-aware location search leveraging crowdsourced digital footprints from LBSNs,” Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing - UbiComp 13, 2013.

[13] G. Yao, "User-Based and Item-Based Collaborative Filtering Recommendation Algorithms Design", 2017.

[14] "gensim: topic modelling for humans", *Radimrehurek.com*, 2017. [Online]. Available: https://radimrehurek.com/gensim/wiki.html#latent-dirichlet-allocation. [Accessed: 07- Jun- 2017].