International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 05 | May-2016 www.irjet.net p-ISSN: 2395-0072

Friend-book: - A Friend recommending system through Life-Style

Nitish Gaikwad[1], Ankit Bhanushali[2], Dheeraj Mishra[3], Rupali Nirgude[4]

[123]Student, B.E Computer Engineering, Dr .D. Y. Patil Institute of Engineering and Tech, Ambi, Maharashtra, India
[4]Professor, Dept. of Computer Engineering, Dr. D. Y. Patil Institute of Engineering and Tech, Ambi, Maharashtra, India

---

Abstract - *Existing social networking services recommend friends to users based on their social graphs, which may not be the most appropriate to reflect a user's preferences on friend selection in real life. In this project, we present friend-book, an efficient friend recommendation system which recommends friends to user on their life style, hobbies instead of social graphs. Friend-book discovers life style of user on their likes and hash-tags and measures the similarity of hash-tags between users, and recommends friends to users if it has complete similarity. And this is done by using Latent Dirichlet Allocation Algorithm. We further propose a similarity metric to measure hash-tags between users, and calculate user's impact in terms of hash-tags with a friend-matching graph. Upon receiving a request, friend-book recommends a list of people with high recommendation on hash-tags to the user query. Finally, friend-book integrates a feedback mechanism to further improve the recommendation accuracy.*

Key Words: Activity Recognition, Social Networks, Text Mining, Data Mining, Pattern Recognition.

1. INTRODUCTION

Existing social networking services recommend friends to users based on their social graphs, which may not be the most appropriate to reflect a user's preferences on friend selection in real life. In this project, we present friend-book, an efficient friend recommendation system which recommends friends to user on their life style, hobbies instead of social graphs. Friend-book discovers life style of user on their likes and hash-tags and measures the similarity of hash-tags between users, and recommends friends to users if it has complete similarity in hash-tags. In this project, we have proposed a user blog and hash-tag block for recommending friends. And this is done by using Latent Dirichlet Allocation Algorithm. In this project, user has to complete his user blog and hash-tag block and hash-tag block should be with the words present in the user blog. Friend recommendation is completely based on the hash-tag block with 100% matching of hash-tag block. We further propose a similarity metric to measure hash-tags between users, and calculate user's impact in terms of hash-tags with a friend-matching graph. Upon receiving a request, friend-book recommends a list of people with high recommendation on hash-tags to the user query. The different aspect of interests, life styles we have included are:-

1) Theatre
2) Music
3) Business
4) Programmer
5) Art
6) Sports
7) Food

Finally, friend-book integrates a feedback mechanism to further improve the recommendation accuracy.

2. LITERATURE SURVEY

Recommendation systems can be divided into two areas of focus: object recommendation and link recommendation. Companies such as Amazon and Netflix emphasize object recommendation where products are recommended to users based on past behavioral patterns. Social networking sites such as Face-book and LinkedIn focus on link recommendation where friend recommendations are presented to users.

The work we present in this project, we develop friend recommendations system within social networks. The recommendation algorithms employed by sites such as Face-book are proprietary. However, through observation, it is apparent that a friends-of-friends approach is being used. This approach is useful and efficient due to ease of implementation and the nature for humans to be drawn together through association. Similar network based approaches such as graph based induction and link mining have been considered but fall in comparison to the effectiveness and efficiency of a friend of friends approach.

Kuan et al.: Kuan proposes an algorithm to locate groups using a transitive extension based approach. This research proposed the use of a 1.5Cclique extension method to derive sub structures, or communities, within social networks. Results showed that this method was fairly effective in finding community of friends. However, this method does not provide insight into how these communities are formed. That is, it is significant to understand what common interests cause a formation in these communities. Recent

research has identified the potential effectiveness of combining complex network theory and genetic algorithms.

Silva et al.: Silva treated the recommendation problem as a filtering problem where a genetic algorithm was used to optimize three indices derived from structural properties of social networks. The result from this study was acknowledged as a baseline to initial work using a new methodology. Study of few recommendation pattern used by websites: Amazon recommendations change regularly based on a number of factors. These factors include time and day of purchase, rate or like a new item, as well as changes in the interests of other customers. Because your recommendations will fluctuate, Amazon suggests you add items that interest you to your Wish List or Shopping Cart. E-Bay recommends product on bases of features of items. You Tube recommends items based on like/dislikes concept. In.com recommends the songs that are popular, songs from the same movie, similar actor-actress, artist, director etc. RS is used to filter the item/product according to the user interest and looking at the like-minded-users.

Collaborative filtering: There are many popular recommendation algorithms based on collaborative filtering. Collaborative Filtering creates a group of users with similar behavior, and finds the items preferred by this group. Ratings from user will be taken from user in two ways explicit rating and implicit rating. CF algorithms are divided into two types, memory-based algorithm and model based algorithm.

Memory-Based algorithm: Memory-Based algorithm simply stores all the user ratings into memory. There are two variants of memory-based recommendation and both are based on the k-Nearest Neighbor algorithm: user-based filtering and item-based filtering. In User - Based Filtering, Rating matrix is used to find neighboring users for the active user. This is done by using cosine or Pearson's correlation matrix. After knowing the neighboring user for active user, items preferred by neighboring users will be sorted on frequency and rating of items. Items that are not known to active user will be recommended. Item Based Filtering finds the most similar items. Items are considered to be similar when the same set of users has purchased them or rated them highly. For each item of an active user, the neighborhood of most similar items is identified. Collaborative filtering techniques can be expanded to other algorithms such as tag based and attribute aware and trust aware recommender systems. A diffusion based recommendation algorithm is proposed which consider the personal vocabulary. A hybrid user profiling strategy is proposed that take advantage of both content based profiles describing long-term information interests that a recommender system can acquired a long time and interests revealed through tagging activities, with the goal of enhancing the interaction of users with a collaborative tagging system. **Trip Tip system** is proposed to help negotiate traveler's way through the immense amount of information that is often available by recommending a set of choices. Trip Tip recommends to the users the next place, which they would most likely want to visit given their preference in previous choices. To generate this information, tags that are attached on a given place by users give the characteristics of a place and the reasons for visiting the place. Attribute-aware method pro-posed takes into account item attributes, which are defined by domain experts. In addition, content based algorithms can provide very accurate recommendations. Collaborative tagging systems (CTSes), allow users to freely assign tags to their collections, provide promising possibility to better address the above issues. A generic method was proposed that allows tags to be incorporated to the standard collaborative filtering, via reducing the ternary correlations to three binary correlations and then applying a fusion method to re-associate these correlations. Some diffusion-based algorithms are recently proposed for personalized recommendations. A spreading Action based collaborative filtering was proposed which is essentially an iterative diffusion process. A diffusion-based top-k collaborative filtering performs better than pure top-k CF and pure diffusion-based algorithm. Besides recommender systems, research on context aware computing seems promising. Context-awareness allows software applications to use information beyond those directly provided as input by users. More recently, there were attempts to define architectures for context-aware recommender. However, authors don't give details about the deployment of such architectures. An algorithm is proposed which adopt item-based algorithms in the early stage of the cold-start period and eventually switching to SVD based algorithms. A collaborative filtering recommendation algorithm based on the implicit information of the new users and multi-attribute rating matrix is proposed to solve the cold start problem.

3. Proposed System

Friend-book, an efficient friend recommendation system which recommends friends to user on their life style, hobbies instead of social graphs. Friend-book discovers life style of user on their likes and hash-tags and measures the similarity of hash-tags between users, and recommends friends to users if it has complete similarity in hash-tags. In this project, we have proposed a user blog and hash-tag block for recommending friends. And this is done by using Latent Dirichlet Allocation Algorithm. In this project, user has to complete his user blog and hash-tag block and hash-tag block should be with the words present in the user blog. Friend recommendation is completely based on the hash-tag block with 100% matching of hash-tag block. We further propose a similarity metric to measure hash-tags between users, and calculate user's impact in terms of hash-tags with

a friend-matching graph. The similarity threshold used for the friend-matching graph is fixed in our current prototype of Friend-book. Our proposed system addresses the problem of computing large amount of data and scalability. In our proposed method, we use incremental computation of Page Rank; it can be implemented incrementally (or) distributive for large scale evolving graphs. Additionally, we propose a novel algorithm, Weighted Page Rank algorithm which distributes rank score based on popularity of the pages and we set threshold for each edge & it can represent the similarity relationship of friend-matching graph. We also have implemented Hex Converter to efficiently secure the password of the users. Upon receiving a request, friend-book recommends a list of people with high recommendation on hash-tags to the user query. Finally, Friend book integrates a feedback mechanism to further improve the recommendation accuracy.

3.1 Advantages

- Recommends potential friends to users if they share similar life styles.
- The feedback mechanism allows us to measure the satisfaction of users, by providing a user interface that allows the user to rate the friend list.

3.2 Architecture

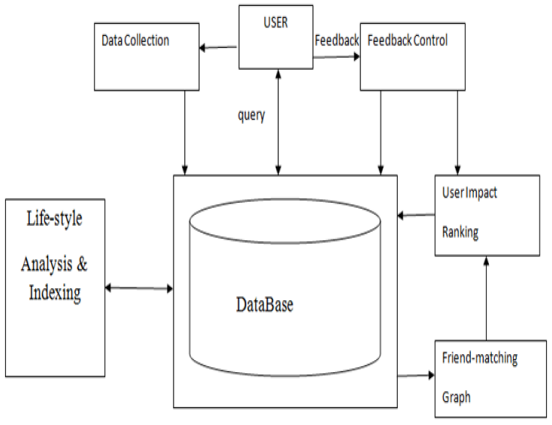

Fig-1: System Architecture

3.3 LDA Algorithm

Example of LDA Algorithm:

Suppose you have the following set of sentences:

- I **eat fish** and **vegetables**.
- Fish are pets.
- My kitten **eats fish**.

LDA is a technique that automatically discovers topics that these documents contain.

Given the above sentences, LDA might classify the **bold** words under the **Topic F**, which we might label as "**food**". Similarly, words in italics might be classified under a separate Topic P, which we might label as "pets".

LDA defines each topic as a bag of words, and you have to label the topics as you deem fit.

There are 2 benefits from LDA defining topics on a word-level:

1) We can infer the content spread of each sentence by a word count:

- **Sentence 1**: 100% Topic F
- **Sentence 2**: 100% Topic P
- **Sentence 3**: 33% Topic P and 67% Topic F

2) We can derive the proportions that each word constitutes in given topics. For example, Topic F might comprise words in the following proportions: 40% eat, 40% fish, 20% vegetables. We can implement LDA to achieve the above results in 3 steps.

Step 1: First we give input that how may topics are there. You can either use an informed estimate e.g. results from a previous analysis, or simply trial-and-error. In trying different estimates, you may pick the one that generates topics to your desired level of interpretability, or the one yielding the highest statistical certainty i.e. log likelihood.

Step 2: The algorithm will assign every word to a temporary topic. Topic assignments are temporary as they will be updated in Step 3. Temporary topics are assigned to each word in a semi-random manner i.e. according to a Dirichlet distribution, to be exact. This also means that if a word appears twice, each word may be assigned to different topics. Note that in analyzing actual documents, function words (e.g. "the", "and", "my") are removed and not assigned to any topics.

Step 3: This is iterative step. The algorithm will check and update topic assignments, looping through each word in every document. For each word, its topic assignment is updated based on two criteria:

i. **How meaningful is that word across topics?**
ii. **How meaningful are topics in the document?**

All "fish" words across both documents comprise nearly half of remaining Topic F words in Document are assigned to Topic F and Topic P in a 50-50 ratio. To understand how these two criteria work, imagine that we are now checking the topic assignment for the word "fish" in Document.

- **How prevalent is that word across topics?** Since "fish" words across both documents comprise nearly half of remaining Topic F words but 0% of remaining Topic P words, a "fish" word picked at random would more likely be about Topic F.

- **How prevalent are topics in the document?** Since the words in Doc Y are assigned to Topic F and Topic P in a 50-50 ratio, the remaining "fish" word seems equally likely to be about either topic.

Weighing conclusions from the two criteria, we would assign the "fish" word of Doc Y to Topic F. Doc Y might then be a document on what to feed kittens. The process of checking topic assignment is repeated for each word in every document, cycling through the entire collection of documents multiple times. This iterative updating is the key feature of LDA that generates a final solution with coherent topics.

Graph-

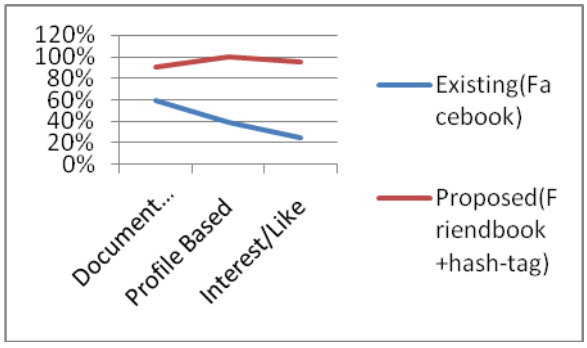

Chart -1: Difference between Existing and Proposed System on Performance

4. CONCLUSIONS

We proposed an efficient recommendation system for social networks i.e. Friend-book. In Friend-book, recommendation is done through life styles of the user. The likes and hash-tag of the users are major part in recommending the friends to the user. Friend-book is quiet efficient than the existing system in terms of performance. It gives fast and better results as compared to existing system.

REFERENCES

[1] Zhibo Wang. Qing Ca, and Zhi Wang, "Friend-book: A semantic-based Friend Recommendation system for Social Networks", IEEE TRANSACTIONS ON MOBILE COMPUTING.
[2] Amazon. http://www.amazon.com/.
[3] J. Kwon and S. Kim," Friend recommendation method using physical and social context," Int. J. Computer Sci. Newt. Security Vol. 10, no, 11, pp, 116-12, 2010.
[4] Netflix. https://signup.netflix.com/.
[5] Rotten tomatoes. (2014). [Online]. Available: http://www.rottentomatoes.com/.
[6] G.R. Arce, Nonlinear Signal Processing: A Statistical Approach, New York, NY, USA: Wiley, 2005.
[7] B. Bahmani, A. Chowdhury, and A. Goel, "Fast incremental and personalized page rank," Proc. VLDB Endowment, vol. 4, pp. 173-184, 2010.
[8] J. Biagioni, T. Gerlich, T. Merrifield, and J. Eriksson, "Easy Tracker: Automatic transit tracking, mapping, and arrival time prediction using Smart-phones," in Proc. 9th ACM Conf. Embedded Netw. Sensor Syst., 2011, pp. 68-81.
[9] L. Bian and H. Holtzman, "Online friend recommendation through personality matching and collaborative filtering," in Proc. 5th Int. Conf. Mobile Ubiquitous Computer, Syst., Services Technol., 2011, pp. 230-235.
[10] C. M. Bishop, Pattern Recognition and Machine Learning. New York, NY, USA: Springer, 2006.
[11] D.M. Blei, A. Y. Ng. and M. I. Jordan, "Latent Dirichlet Allocation," J. Mach, Learn. Res., vol. 3, pp. 993-1022, 2003.
[12] P. Desikan, N. Pathak, and J. Srivastava, V. Kumar," Incremental page rank computation on evolving graphs," in Proc. Special Interest Tracks Posters 14th Int. Conf. World Wide Web, 2005, pp. 1094-1095.
[13] N. Eagle and A. S. Pentland," Reality mining: Sensing complex social systems," Pers. Ubiquitous Computer, vol.10, no. 4, pp. 255-268, Mar. 2006.
[14] K. Farrahi and D. Gatica-Perez," Probabilistic mining of socio geographic routines from mobile phone data," IEEE J. Select. Topics Signal Process, vol. 4, no. 4, pp. 746-755, Aug. 2010.
[15] K. Farrahi and D. Gatica Perez," Discovering routines from large scale human locations using probabilistic topic models," ACM Trans. Intel. Syst. Technol., vol. 2, no. 1, pp. 3:1-3:27, 2011.
[16] B. A. Frigyik, A. Kapila, and M. R. Gupta," Introduction to the Dirichlet distribution and related process," Dept. Elect.

Eng., Univ. Washington, Seattle, WA, USA, UWEETR-2010-0006, 2010.
[17] A. Giddens, Modernity and Self-Identity: Self and society Univ. Press, 1991.
[18] L. Gou, F. You, J. Guo, L. Wu, and X. L. Zhang," Sfviz: Interest based friend's exploration and recommendation in social networks," in Proc. Visual Inform. Commun. - Int. Symp. 2011, p.15.
[19] W. H. Hsu, A. King, M. Paradesi, T. Pydimarri, and T. Weninger," Collaborative and structural recommendation of friends using weblog-based social network analysis," in Proc. AAAI Spring Symp. Ser., 2006, pp. 55-60.
[20] T. Huynh, M. Fritz, and B. Schiel," Discovery of activity patterns using topic models," in Proc. 10th Int. Conf. Ubiquitous Computer, 2008, pp. 10-19.