**Cross-Domain User Modeling: Applying Graph Techniques for Reasoning on Personal Data from Social Networks**

**Research proposal for M.sc degree in the I.S department**

**Oz Shapira**

Contents

[1. Introduction 3](#_Toc342760027)

[2. Background and Related works 4](#_Toc342760028)

[2.1. Background 4](#_Toc342760029)

[2.1.1. Recommender systems 4](#_Toc342760030)

[2.1.2. Graphs as data structures 6](#_Toc342760031)

[2.1.2.1. Graph traversal 6](#_Toc342760032)

[2.1.3. Social networks (SN) 6](#_Toc342760033)

[2.2. Related work 8](#_Toc342760034)

[2.2.1. Generic Semantic-based Framework 8](#_Toc342760035)

[2.2.2. Taste Fabric of Social Networks 8](#_Toc342760036)

[2.2.3. Network Profiles as Taste Performances 8](#_Toc342760037)

[2.2.4. On the Social Web 8](#_Toc342760038)

[2.3. Summery 8](#_Toc342760039)

[3. Research Goals and Questions 9](#_Toc342760040)

[4. Tools and Methods 10](#_Toc342760041)

[4.1. Methods 10](#_Toc342760042)

[4.1.1. Data Source 10](#_Toc342760043)

[4.1.2. Crawling Pinterest 11](#_Toc342760044)

[4.1.3. Data collection and graph representation 12](#_Toc342760045)

[4.2. Tools 13](#_Toc342760046)

[4.3. Evaluation 14](#_Toc342760047)

[5. Timetable 15](#_Toc342760048)

[6. Initial Results 15](#_Toc342760049)

[7. Research Contributions 15](#_Toc342760050)

[8. References 15](#_Toc342760051)

# Introduction

What if we know that someone loves pizza, James bond movies and he also loves jogging on the beach and will ask for a recommended recipe of a cake he may love? How can we make use of the information we already know about the personal preferences of this user regarding fast food, entertainment and sport in order to suggest a preferred recipe for a cake? This research will try to address this challenge and suggest a recommender system able to answer on this kind of questions of how to use information from one domain for reasoning about preferences in another domain, using general graph based techniques.

“Recommender systems represent user preferences for the purpose of suggesting items to purchase or examine. They have become fundamental applications in electronic commerce and information access, providing suggestions that effectively prune large information spaces so that users are directed toward those items that best meet their needs and preferences“ [Burke 2002]. Recommender systems became an important research area since the appearance of the first papers on collaborative filtering since the mid-1990s [Adomavicius and Tuzhilin 2005]. There has been much work done both in the industry and academia on developing new approaches to recommender systems over the last decade. Examples of such applications include recommending books, CDs and other products at Amazon.com, movies by IMDB , and news at VERSIFI Technologies (formerly AdaptiveInfo.com). [Adomavicius and Tuzhilin 2005]. Most recommender systems nowadays are focused on providing a personalized service in a specific domain, as does Pandora – a music recommender system or IMDB movies recommender system (see figure 1 as an example).



Figure 1: from IMDB website as movie recommender system

In order to provide a personalized service to their users, recommender systems need to have relevant personal information about their users, or a “model” of them - a “User Model”. When this information is available, then the task of recommendation may be straight forward – provide a service based on the relevant information. In our example, if the system knows the user’s preferences about cakes, then finding a similar recipe becomes and easy task. However, in many cases, like in our example, the system does not have this information. This may be the case of a first time visitor to a cultural heritage site/a new city/ restaurant etc. The lack of sufficient user modeling data at the onset of a service is among the classical and well known problems of user modeling and recommender systems – the “cold start” problem [Guo 1997].

Nowadays, as we surf and visit websites we leave identifiable digital “fingerprints” not to mention explicit definition of interests and preferences, that are used or may be used for personalization purposes. As a result, many Web sites contain partial and specific user models which reflect user characters that are relevant for personalizing the services they provide. In recent years, social networks that contain freely available and diverse information about users became a major source for personal information [Boyd 2007]. The freely available personal information, scattered over various online sources (including social networks) may be a valuable source of information for building an initial user model for recommendation. However, even though these social networks may be rich in personal information, they may lack specific personal information that is required for a specific personal service requested.

In order to address this issue, of using information available about a user in one domain for recommendation in another domain, “cross domain” recommendation/personalization was defined – how can we use personal information available about the user in one domain for providing service in another domain [Berkovsky et al. 2007]. Still, “user models “Mediation”, the solution suggested by [Berkovsky et al. 2007], requires some semantic knowledge and specific mediation mechanism. Other interoperability approaches surveyed by Carmagnola et al. [2007] requires user tagging activity

The proposed research is intended to address the issue of cross domain recommendation by integrating personal information that is freely available in social networks with a simple yet powerful graph based representation. User characteristics will be represented by nodes and relations between them will be represented by edges. Traversing the graph will enable to find out relations and links between characteristics that were not explicitly defined in the original information sources. We plan to explore, demonstrate and evaluate the ability to use graph based representation of user modeling for representing and reasoning on data elicited from social network in order to help solving the cross-domain user modeling challenge.

# Background and Related works

## Background

### Recommender systems

Recommender systems are now an integral part of some e-commerce sites such as Amazon.com and CDNow [Schaferet al. 1999]. Recommender system applies knowledge discovery techniques to the problem of making product recommendations during a live customer interaction. These systems are achieving widespread success in E-commerce nowadays, especially with the advent of the Internet [Sarwar et al.2000] .there is many type of recommendation systems and in each one of them have different approach for recommendation (vs. Pandora[[1]](#footnote-1) , Google[[2]](#footnote-2) search ,YouTube[[3]](#footnote-3) ,amazon[[4]](#footnote-4) etc.) There are several types of recommendation techniques when the major fundamental techniques are:

#### Content-Based,

Recommendations are based on matching semantic properties (preferences) of items similar to those that user liked in the past. A content-based recommender learns a profile of the user’s interests based on the features present in objects the user has rated. The user model depends on the learning method employed. There are many methods for establishing content based recommendations, including decision trees, neural nets, and vector-based etc. [ref]

Adomavicius et el. [2005] conclude this technique in the following algorithm: if utility u(c, s) (when u is the method used for recommendation) of item s for user c is estimated based on the utilities u(c , si) assigned by user c to items si ∈ S that are “similar” to item s. for example let’s assume we have music recommender system M in order to recommend song s to user c, the system will try understand (using utility u) the commonalities among music user c has rated highly in the past – them only songs that have the highest similarity degree to users preference will recommended.

**Content-based disadvantage** is that recommendation is based on past user preferences and will recommend only “more of the same” – items that are similar to those the user liked [ref].

#### Collaborative-Based

This technique implemented in variety of commercial systems, where the recommendations are based on “mutual taste” as represented by previous ratings of users to items, with the assumption that users who agreed in the past on item ratings are likely to agree again in the future, in some case the ratings is binary (like/dislike) like Pandora or real-valued indicating degree of preference like movie rating in IMDB [ref]. Some of the most important systems using this technique are GroupLens/NetPerceptions [Resnick et al. 1994], Ringo/Firefly [Shardanand & Maes 1995], Tapestry [Goldberg et al. 1992] and Recommender [Hill et al. 1995]. These systems can be memory-based, some use users to compare against each other use direct approach or other measure. Other system use model-based which a model is derived from the historical rating data to make predictions [Breese et al. 1998]. Model-based recommenders have used a variety of learning techniques including neural networks [Jennings & Higuchi, 1993], latent semantic indexing [Foltz, 1990], and Bayesian networks [Condliff, et al. 1999].

Collaborative techniques is are completely independent of any machine-readable representation of the objects being recommended, and work well for complex objects such as music and movies where variations in taste are responsible for much of the variation in preferences. Schafer et al. [1999] call this “people-to-people correlation.

**Collaborative-based disadvantages** – This approach suffers from the cold start problem- when recommendation is needed to new user when we have very few data on him or about a new item.

#### Hybrid systems

”Hybrid recommender systems combine two or more recommendation techniques to gain better performance with fewer of the drawbacks of any individual one” [burke 2002]. Since hybrid systems are combination of several techniques they have the abilities to overcome on each techniques weakness.

Burke [2002] surveys additional common technique like Demographic, Utility-based and Knowledge-based. Recommender systems are commonly based on some personalize estimating rating technique which saved on internal data information while recommendation algorithm is gain throw users rating [Adomavicius et al. 2002] .

### Graphs as data structures

Graph is a representation of a set of objects where some pairs of the objects are connected by links The interconnected objects are represented by mathematical abstractions called vertices, and the links that connect some pairs of vertices are called edges (Wikipedia 2012 ). Graphs are widely used for modeling complicated data, including chemical compounds, protein interactions, XML documents, and multimedia [jiang 2007], main advantage of deploying graph in computerized systems is the abilities to automatically traverse and reason on it. Used graph is to reduce analysis of source-target implantation based on web linked hierarchy (MapReduce problem) [Dean & Ghemawat 2004] .

### Graph traversal

Graph traversal (the search problem) is the problem of visiting all the nodes in a graph in a particular manner, updating and/or checking their values along the way. Verity algorithms exist for reducing graph for like BFS[[5]](#footnote-5),DFS[[6]](#footnote-6) ,Dijkstra[[7]](#footnote-7), a different way is to change graph structure for reduce searching time ,other common approach is to represent both graphs and queries on graphs by sequences, thus converting graph search to subsequence matching[jiang 2007].

### Social networks (SN)

Social networking service or at the they short name Social networks are an online services, platform, or sites that focuses on facilitating the building of social networks or social relations among people who, for example, share interests, activities, backgrounds, or real-life connections. A social network service consists of a representation of each user (often a profile), his/her social links, and a variety of additional service.[Wikipedia[[8]](#footnote-8) 2012] They have been with us since 1997 (the first one was sixDegrees.com), social networks site (SNS) have successfully changed worldwide communication. They gave personal users the ability to reach any user in the world. They attracted millions of users, many of whom have integrated these sites into their daily practices. As of this writing, there are hundreds of SNSs, with various technological affordances, supporting a wide range of interests and practices (for example Facebook ,Google+ ,twitter ,Linkedin etc) those abilities allow SNS the to connect between separate type of population using SNS users, boyd [2007] rise the fact the SNS can provide rich sources of personalize data. Profile and linkage data from SNSs can be gathered either through the use of automated collection techniques or through datasets provided directly by the company, enabling network analysis researchers to explore large-scale patterns of friending, usage and other visible indicators [Hogan, B 2007], continuing an analysis trend that started with examinations of blogs and other websites.

SNS basically contain social circles when each one of those circles can relate to different aspect. For example a regular user in LinkedIn (LinkedIn a professional SNS specialize on work relation between work colleagues) will have a work circle but s/he also can be at a different circle like friend from school or military service. A similar concept can append in Facebook: A user may have friends from different circles: school, university, work place, neighborhood, preferred music, food etc. in Google+ they even coded this feature as you can create or join to “circle”. Important circle is shared interests and preference circle, is this circle users like to connect to each other through shared subjects for example fans group of rock band, movie fans, members at sushi restaurant etc. this values can establish large data collection of user’s preference and interests, this effort of collecting data have been mention before by Rhodes, Bowie and Hergenrather[2003] in their research they concluded that using the web as empiric tools for behavioral science research will increase the tested population from local to global distribution.

#### Social Network (sn) as a source

Social networks (SN) can be effective sources for establish database, the main key in social networks is to shard the individual to the common population. Each social network are depend user data, in general users are uploading their personal data to the SN, they are owned the decision to share information to common population.

Abdesslem, Parris, and Henderson [2011] concluded the use of SN for collecting data, they speared the collecting to two sections – collect user social behavior in SN and collect user characteristics. From their aspect when we used SN as our source not only is preference and characteristics we can actually build social profile from the user data, Fehmi [2012] used Facebook, for creating random sampling, they create recursive process that extract new users form for users friends – with this approach they achieve random sampling and successfully manage to recursive process but keep their subjects normally distributed throw all many different users.

## Related work

### Generic Semantic-based Framework

Fernández-Tobías et al. [2011] try to create an automated system that will recommend to user preference by to different domains in their approach the used graph for mapping the connection between the domain and analyzing the nodes relation in graph, this approach adopt the *Content-based recommendations* mention on paper by Adomavicius1 and Tuzhilin [2005] used past user data for create the recommendation, in their system they used DBpedia as the database source. DBpedia is a graph based database that contains values from Wikipedia. The main problem with that experience is that DBpedia is not updated daily. Since they can be depend on that database they created description framework built upon semantic networks, the problem with attitude it’s for obtaining data you to maintain and upgrade your frameworks.

Need to add : fabric taste ->extract 3-4 related works

### Taste Fabric of Social Networks

Liu Maes and Davenport [2006] mined 100,000 social network profiles , by using machine learning technique they segmented them into categories interest like of music, books, films, foods, etc. and infer semantic fabric of taste. They examined ways in which the performance of tastes constitutes an alternate network structure which they call a ‘‘taste fabric.” this effort had help the creation of semantically flexible user representations, cross-domain taste-based recommendation, and the computation of taste-similarity between people.

### Network Profiles as Taste Performances

Another research perform by Liu[at el 2007] was to better understand user taste performances, using semiotic framework interest tokens are been analyze when socioeconomic and aesthetic influences on taste are considered, he based a theory to sort taste statements to 4 types: prestige, differentiation, authenticity, and theatrical persona. by analysis of 127,477 profiles collected from the MySpace SN. He founded statistical evidence for prestige and differentiation that are unique for MySpace community.

### On the Social Web

Abel and herder [2011] developed and evaluated the performance of several cross-system user modeling strategies in the context of recommender systems .they analyzed large dataset of more 25000 user profiles from Facebook ,LinkedIn, Twitter, Flickr and delicious and aggregated their data and created SN aggregated tool for improving recommendation results The evaluation results show that the proposed methods solve the cold-start problem and improve recommendation quality significantly, even beyond the cold-start.

## Summery

Recommendation systems use common people knowledge about items in an information domain to help people choose other items. Most recommendation systems determine the “decision” by internal data which gain by collaborative way, contend base or the hybrid of attitudes. The Major barriers of recommender systems is most of them with different recommending technique, different domains in different user context [kuflik 2012], when users spread their modules across separate domains for example user have is music profile in Pandora while in IMDB he have movie preferences , in Amazon he have his favorite fashionable shirts etc. Common user at the World Wide Web have different context throw several domain which create user duplicated information, data that was exist in one domain will not reflect on other domains. Berkovsky [2008] understand this problem and propose general framework for enhancing the accuracy of user modeling in recommender systems, he suggest user models mediation process that will be cross-user, cross-item, cross-context and cross-representation, in his research he develop a generic mediation mechanism for integrating user modeling data in a distributed environment. He contribute to creation of more accurate user modeling that will assist to hybrid recommendation technique.

We want to take this step forward and used Berkovsky conclusion to create cross-domain recommendation system with more accurate user model by collection user’s interest and preference from social networks and find the their common taste as same Liu also collect user preference throw social networks and base his recommendation from machine learning , but machine learning require process that not contribute to cold start problem for overcoming this problem contend base recommender system that will used graph for mapping user interest relation . Fernández-Tobías mapped in graph relation between music and location interest we want to create generic process with abilities to map any type of interest inside large scale mathematic graph. Recommendation process can be Usage throw graph traverse algorithms for finding optimal recommendation algorithm and create cross-domain recommendation when our base data was ported throw user’s personalization interest exist inside Social networks into graph database will be attribution the cold start problem.

# Research Goals and Questions

The “Cold Start” problem is a well-known problem in user modeling and recommender systems – how to bootstrap a user model in order to provide the user with a specific personalized service. Given the fact that a lot of personal information may be available in various sources, and the fact that this information may not exactly represent the user interests/needs/preferences in the target domain, a question is how can we use information available about a user in one domain for modeling the user in another domain, or “cross domain” modeling or recommendation.

As social networks are known to be rich source for freely available diverse personal information, we plan to explore the use of such source for cross domain recommendation. The goal of the proposed research is to explore the possibility to use freely available information on a social network for cross-domain recommendation using a graph representation of a user model.

It is assumed that the wealth and variety of information that is available in social networks can be used for cross-domain user modeling when represented in a simple graph data structure and by applying generic graph search techniques.

The research will answer the following question:

*How we can use the social network of curators for mining the links between topics of interest for cross domain recommendation?*

# Tools and Methods

We will construct a research tool called **TraitsFinder** that will allow us to collect user’s information and extract the data [[9]](#footnote-9)from a specific social network – Pinterest and build a graph that will represent the connection between traits, the research will work by two steps:

* **TraitsFinder**  will crawl pinterest social networks and collect user’s information in our servers, the outcome of this step is users folder with user information save in XML files.
* **TraitsFinder** will create graph base users crawled data – this step can run in offline mode or online (user information are update immediately after saved) the outcome of this step update neo4j graph.

## Methods

The research is a design research [Hevner et al. 2004]. As such, an experimental tool will be built. It will be used for representing user models over a graph and graph based techniques will be used for cross domain recommendation generation. Personal information will be collected from web Social network called Pinterset By Our research tool Called **TraitsFinder** The information will then be uploaded to the graph using also **TraitsFinder.**

### Data Source

Our data source is users personal preferences that exits at a SN called pinterest, inside Pinterest we can extract users albums that represent user interests like hobbies food , music etc. Since we want to rely on social networks (SN) as source, the logical conclusion was to use Facebook, since (at least for now) Facebook is the biggest SN exist, it’s have more them one billion members, it’s update very frequently – today most of their users user updates data in the mobile devices, since it’s popularity it’s cover almost any type of population at any age. However, after massive investigation with Facebook API we discovered that Facebook does not allow developers to collect data from Facebook users (if this ability was exist and publish most likely Facebook have been busy with defending itself agents legal claims) instead, a regular user needs to access to some kind of application (it’s can be game, puzzle, quiz interview or any application we want) and once the user accessed the application then through the application researchers/developers can access the user’s profile using Facebook query language (FQL). Hence we decided to abandon Facebook for several problems we encountered:

* **Sampling problem** - in Facebook we can’t samples random users’ instants and we need to create some kind of “bait” for calling our users ,with this action our effort will be leaning for particular population (population that was interest in our application ).
* **Circle problem -** since we are not sampling random users we can’t get access to users’ friends. We can only publish in our participant users wall – this action will not help to expand our sampling, intend it’s will create a circle of users that will use the application.
* **Semantic problem –** in Facebook users upload pictures, update status, join groups, check in places etc. from all these actions it is hard to understand user characteristics and there is a need for a sematic parser for analysis of user preferences and traits. For example if user upload birth cake with no explanation what we can understand form this picture, that he have birthday party, he love to bake cake or he just love cake.
* **Legal issues** – if we use Facebook we will need to ask or mention to the user this is academic experiment - this can also harm our user sampling.

Unlike Facebook we can use Pinterset SN for collecting relevant users’ information. Pinterest is a curator photograph sharing social network. Pinterest is not only simple and have specific attribution we need – in Pinteres user interested **catalogued** to subjects, we also get the connection between users – when user upload photo and catalogue it, any other user that will pin this picture we can understand and analyze is connection to that picture, we have also very big advantage in Pinteres the subjects are basically our characteristic that we seek. In addition we don’t have to become entangled with random sampling issue – when can just sample all the users .

### Crawling Pinterest

Unfortunately pinterest doesn’t have API, so in order to crowl the website we connect to web application that parse HTML web page and extract the data from web site: The data in Pinterest exists in chronologic way at the main page there pictures with comment when you press at one picture you access to user that upload the picture under is relevant subject. The hierarchy in pinterest is simple and constant is work as follow way:

Figure 3 : Pinterest hierarchy

Since the hierarchy is simple and fixed we can use the web crawler for establish our data, as follows:

*Go to pictures group G*

*Crawl(****G****)*

*{*

*If* ***G*** *is empty exit*

*Else*

*{*

*Find pictures* ***P*** *from* ***G***

*Save Comment* ***C*** *from* ***P*** *under* ***P***

*Foreach user* ***X*** *in* ***C***

*{*

*Add user* ***X*** *to group* ***U***

*Foreach subject* ***S*** *in* ***X***

***{***

*Save subject* ***S****i under user* ***X***

*Save all items (it) under* ***S****i*

***}***

*}*

*Crawl(X)*

*Crawl(U)*

*}*

*}*

### Data collection and graph representation

The data we are going to extract will be saved in files for each user, subject, picture and comment. The crawling process will convert the HTML pages to standard XML files under folder hierarchy (see figure 5) at same way we save the subjects and pictures page files.

The crawling process will create hierarchy of folders and xml files. From those files we can parse the content and retrieve information a graph. with the ability to represent the native ontology of user curator subjects.

#### Main general graph

The main graph will represent all type of connections inside Pinterest website, each object will be document in our main graph.

The main graph is bend to Pinterest hierarchy will marked as graph G when each node (V) will represent object in Pinterest hierarchy (users, subjects , items ) therefor V={users ,subjects ,items} , the edges (E) are the connection is represent belonging at the hierarchy when EE ⊆ {u,v∈V } for example : let’s presume user\_x ∈ V when he owned subjects like pizza ,animals (animals ∈ V, pizza ∈ V) and we also have user\_y that also love animals and pizza we will create edge between user and is subjects (user\_x,pizza)∈E and (user\_x,animals)∈E and relation between user\_y ((user\_x,pizza)∈E & (user\_x,animals)∈E)

User\_z love cars(cars ∈ V) and pizza also ,therefore (user\_z,pizza) ∈E & (user\_z,cars) ∈E.

pizza subject owned items likes tuna\_pizza ,olives\_pizza and mushrooms\_pizza (tuna\_pizza ∈ V ,olives\_pizza ∈ V and mushrooms\_pizza ∈ V) will create edges between subject to item (tuna\_pizza ,pizza) ∈ E , (olives\_pizza, pizza) ∈ E and (mushrooms\_pizza ,pizza ) ∈ E .

With this graph we can analysis the connection between each object in pinterest site



Figure 4: graphical view main graph

#### Interests connection graph

To reduce searching time and decrease the graph traversal we decide to create reduce graph that we represent the relation between interest ,our interests graph is undirected graph G that will extract from the main graph ,each node character will mark as (V,E) when V is our node group (interest) and E represent the edge when EE ⊆{u,v∈V} the edge E is represent connection between the interests, the connection is establish when user X have both interests , the weight of the edge is the number of users obtain the same two interests. In our example the interest graph have only the interest nodes: {animals, pizza, cars} = V the edges are represent the native of the interests connection when (animals, pizza)∈E & (pizza ,cars)∈E when ι(animals,pizza)=2 and ι (Cars,Pizza)=1 the weight animals-pizza edge is 2 since user\_x and user\_y are both like animals and pizza.



Figure 5: graphical view Interests connection graph

Eventually we will have weighted undirected graph that will represent the our graph database when the number of nodes (vertex) is equal to the number of characters |V|=numof(characters) , the |E| represent the number of connections between characters

## Tools

For the purpose of the planned research to use the following tools:

* **TraitsFinder** - we will construct a multithread application cross OS research tool called TraitsFinder, implemented in java that will collect data from Pinterest website. TraitsFinder will Crawl Pinterest and will save data as local xml files. It will have also the ability to upload the information to graph database.
* **Neo4J[[10]](#footnote-10)** – is a high-performance, NOSQL graph database with all the features of a mature and robust database. **TraitFinder** will upload user’s interests to Neo4j graph.
* **Gephi[[11]](#footnote-11)** – Gephi is an interactive visualization and exploration platform for all kinds of networks and complex systems, dynamic and hierarchical graphs. Gephi will be used to survey and analysis neo4j graph .

## Evaluation

The proposed research will be evaluated by collecting publicly available data from social networks regarding users’ preferable items and using it to train a graph based recommendation engine. Approximately 100,000 profiles are to be crawled; initial test crawls show an average of X subjects of interests per user, with Y items in average listed under each interest album.

The cross domain recommendations will be evaluated by using a 10 fold validation as described in Kohavi [1995]. Of the collected data, 90% will be used to train the recommendation engine and 10% for its evaluations. The 10% selected for evaluation will be changed during the evaluation process to include different features of the data. Once the evaluation iterations will result with a minimal error (matching recommendations considered as ‘hits’) single test iteration will provide a final result.

As part of the evaluation it is also intended to analyze how the size of the dataset/size affects recommendations results. This will be done by taking different subsets of the available data and measuring the changes in recommendations quality.

Since our system is cross-domain recommendation system the variety of interest is basically infinite Kohavi [1995] have been investigating using cross-validation and bootstrap for analyzing bias learning. He concluded the more the K-fold is bigger it reduces the variance while increase the bias, since our graph will be high interests variance we will need to equalize graph to the K value in the K-fold cross validation – for better evaluation we find the K value by measuring the all graph interest nodes.

For evaluate this graph we be cross validation technique, we will run cross validation runs as described in the following table (in this example k-fold test when k=1000):

|  |  |  |
| --- | --- | --- |
| Train size (creating graph based on X users) | Number of folds tested user check | Accuracy needed |
| 1000 | 1 |  |
| 10000 | 10 |  |
| 100000 | 100 |  |

The tested fold user will checked by checking recommitting hits, for each tested user we scan is interests and valid our recommitting algorithm on is only 30-40 % of is interests , the recommitting that TraitFiner will return will cross examined with is actually traits. Using cross validation is common technique especially for learning system, since our recommendation system is learning with graph base learning algorithm , we will applied this technique on our system.

# Timetable

Each of the phases discusses the primary focus of each time period:

Phase I - Literature Survey and Focus.

Phase II - establish first stage of TraitsFinder – the web crawler

Phase III - collecting users traits (exit criteria: at least 100,000 users)

Phase IV - broadening TraitsFinder - add graph builder.

Phase V - improve TraitFinder: add automatic graph analysis GUI and RI

Phase VI - Graph analysis, evaluation and algorithm establish

Phase VII - Writing Thesis.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Phase | Spring 2012 | Summer  2012 | Fall  2013 | Spring  2013 | Summer  2013 |
| I |  |  |  |  |  |
| II |  |  |  |  |  |
| III |  |  |  |  |  |
| IV |  |  |  |  |  |
| V |  |  |  |  |  |
| VI |  |  |  |  |  |
| VII |  |  |  |  |  |

# Initial Results

# Research Contributions

The proposed research main contribution to the field of user modeling will be a cross domain recommender algorithm. The algorithm will apply graph analysis methods for the purpose of recommendation generation and will be based on data (interests) extracted publicly available data.

# References

1. Abdesslem, F. B., Parris, I., & Henderson, T. (2012). Reliable online social network data collection. *Computational Social Networks*, 183-210.
2. Abel, F., Herder, E., Houben, G. J., Henze, N., & Krause, D. (2011). Cross-system user modeling and personalization on the social web. User Modeling and User-Adapted Interaction (UMUAI), Special Issue on Personalization in Social Web Systems, 22(3), 1-42.
3. Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. Knowledge and Data Engineering, IEEE Transactions on, 17(6), 734-749.
4. Al-Khalifa, H. S., & Davis, H. C. (2006, November). Folksannotation: A semantic metadata tool for annotating learning resources using folksonomies and domain ontologies. In Innovations in Information Technology, 2006 (pp. 1-5). IEEE.
5. Berkovsky, S., Kuflik, T., & Ricci, F. (2008). Mediation of user models for enhanced personalization in recommender systems. User Modeling and User-Adapted Interaction, 18(3), 245-286.
6. Boim, R., & Milo, T. (2011, April). Methods for boosting recommender systems. In Data Engineering Workshops (ICDEW), 2011 IEEE 27th International Conference on (pp. 288-291). IEEE.
7. Breese, J. S., Heckerman, D., & Kadie, C. (1998, July). Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence (pp. 43-52). Morgan Kaufmann Publishers Inc.
8. Burke, R. (2002). Hybrid recommender systems: Survey and experiments. User modeling and user-adapted interaction, 12(4), 331-370.
9. Carmagnola, F., Cena, F., Cortassa, O., Gena, C., & Torre, I. (2007). Towards a tag-based user model: how can user model benefit from tags? User Modeling 2007, 445-449.
10. Condliff, M. K., Lewis, D. D., Madigan, D., & Posse, C. (1999, August). Bayesian mixed-effects models for recommender systems. In Proc. ACM SIGIR (Vol. 99).
11. Ellison, N. B. (2007). Social network sites: Definition, history, and scholarship. Journal of Computer‐Mediated Communication, 13(1), 210-230.
12. Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P.,and Uthurusamy, R., Eds. (1996) - Advances in Knowledge Discovery and Data Mining
13. Fernández-Tobías, I., Cantador, I., Kaminskas, M., & Ricci, F. (2011, October). A generic semantic-based framework for cross-domain recommendation. In Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems (pp. 25-32). ACM.
14. Foltz, P. W. (1990, March). Using latent semantic indexing for information filtering. In ACM SIGOIS Bulletin (Vol. 11, No. 2-3, pp. 40-47). ACM.
15. Guo, H. (1997). Soap: Live recommendations through social agents. In Fifth DELOS Workshop on Filtering and Collaborative Filtering, Budapest.
16. Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. MIS quarterly, 28(1), 75-105.
17. Hill, W., Stead, L., Rosenstein, M., & Furnas, G. (1995, May). Recommending and evaluating choices in a virtual community of use. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 194-201). ACM Press/Addison-Wesley Publishing Co.
18. Hogan, B. (2008). Analyzing social networks via the Internet. Sage Handbook of Online Research Methods. Thousand Oaks, CA: Sage.
19. Hogan, B. (2008). Analyzing social networks via the Internet. Sage Handbook of Online Research Methods. Thousand Oaks, CA: Sage.
20. iang, H., Wang, H., Yu, P. S., & Zhou, S. (2007, April). “Gstring: A novel approach for efficient search in graph databases”. In Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on (pp. 566-575). IEEE.
21. Jeffrey Dean and Sanjay Ghemawat “MapReduce: Supplied Data Processing on Large Clusters” *Google, Inc (2004)*.
22. Jennings, A., & Higuchi, H. (1993). A user model neural network for a personal news service. User Modeling and User-Adapted Interaction, 3(1), 1-25.
23. Liu, H. (2007). Social network profiles as taste performances. Journal of Computer‐Mediated Communication, 13(1), 252-275.
24. Liu, H., Maes, P., & Davenport, G. (2006). Unraveling the taste fabric of social networks. International Journal on Semantic Web and Information Systems (IJSWIS), 2(1), 42-71.
25. Resnick, P., N. Iakovou, M. Sushak, P. Bergstrom, and J. Riedl. GroupLens 1994: “An open architecture for collaborative filtering of netnews”. In Proceedings of the 1994 Computer Supported Cooperative Work Conference.
26. Rhodes, S. D., Bowie, D. A., & Hergenrather, K. C. (2003). Collecting behavioural data using the world wide web: considerations for researchers. Journal of Epidemiology and Community Health, 57(1), 68-73.
27. Ron Kohavi (1995) A study of Cross-validation and bootstrap” for accuracy Estimation and model selection” ,Stanford university.
28. Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2000). Application of dimensionality reduction in recommender system-a case study (No. TR-00-043). MINNESOTA UNIV MINNEAPOLIS DEPT OF COMPUTER SCIENCE.
29. Schafer, J. B., Konstan, J., & Riedi, J. (1999, November). Recommender systems in e-commerce. In Proceedings of the 1st ACM conference on Electronic commerce (pp. 158-166). ACM.
30. Shardanand, U., & Maes, P. (1995, May). Social information filtering: algorithms for automating “word of mouth”. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 210-217). ACM Press/Addison-Wesley Publishing Co.”.
31. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rives Introduction to Algorithms chapter VI Graph Algorithms (1990)
32. Tsvi Kuflik, Judy Kay and Bob Kummerfeld (2006) “Challenges and Solutions of Ubiquitous User Modeling”

1. <http://www.pandora.com> [↑](#footnote-ref-1)
2. <http://www.google.com> [↑](#footnote-ref-2)
3. <http://www.youtube.com> [↑](#footnote-ref-3)
4. <http://www.amazon.com> [↑](#footnote-ref-4)
5. breadth-first search (**BFS**) is a [strategy for searching in a graph](http://en.wikipedia.org/wiki/Graph_search_algorithm) when search is limited to essentially two operations: (a) visit and inspect a node of a graph; (b) gain access to visit the nodes that neighbor the currently visited node. The BFS begins at a root node and inspects all the neighboring nodes [↑](#footnote-ref-5)
6. Depth-first search (**DFS**) is an algorithm for traversing or searching a tree, tree structure, or graph. One starts at the root (selecting some node as the root in the graph case) and explores as far as ossible along each branch before backtracking [↑](#footnote-ref-6)
7. **Dijkstra's algorithm**, conceived by Dutch computer scientist Edsger Dijkstra in 1956 and published in 1959,[1][2] is a graph search algorithm that solves the single-source shortest path problem for a graph with nonnegative edge path costs, producing a shortest path tree. This algorithm is often used in routing and as a subroutine in other graph algorithms. [↑](#footnote-ref-7)
8. <http://en.wikipedia.org/wiki/Social_networking_service> [↑](#footnote-ref-8)
9. www.Pinterest.com [↑](#footnote-ref-9)
10. http://neo4j.org/ [↑](#footnote-ref-10)
11. https://gephi.org/ [↑](#footnote-ref-11)