School of Computer Science and Engineering

Faculty of Engineering

The University of New South Wales

Managing Your Social Networking Profile

Enabling User-Tailored Views of Your Feed

by

Geoffrey Zhu and Richard Zhang

Thesis submitted as a requirement for the degree of

Bachelor of Engineering in Computer Engineering

Submitted:	26th May 2015
Supervisor:	Dr Hye-young Paik
Assessor:	Prof Fethi Rabhi
Student ID:	3415440 & 3416477
Topic ID:	3031

Abstract

This document describes the requirements to theses submitted for the Bachelor of Engineering in Computer Engineering degree at the School of Computer Science and Engineering. Requirements described are that of both of context and layout of the theses. The document is written using the LaTeX template provided by the school.

- General summary of what we have done
- Though what have we done?? ...

Acknowledgements

This work has been inspired by the labours of numerous academics in the Faculty of Engineering at UNSW who have endeavoured, over the years, to encourage students to present beautiful concepts using beautiful typography.

Further inspiration has come from Donald Knuth who designed TEX, for typesetting technical (and non-technical) material with elegance and clarity; and from Leslie Lamport who contributed LATEX, which makes TEX usable by mortal engineers.

John Zaitseff, an honours student in CSE at the time, created the first version of the UNSW Thesis LATEX class and the author of the current version is indebted to his work.

- Thank you to supervisor
- Thank you to assessor
- Thank you to everyone else . . .

Contents

Chapter 1

Introduction

Social Networking Services (SNS) are platforms where a diverse range of users are able share their interests, organise social activities and keep in touch with people. In almost all SNS, users are presented with a feed; this feed is a list of items generated via the user's connections that the user may be more or less interested in. The feed acts as a summary of activities that the user has subscribed to, a dashboard presented to them when they first log in. This feed contains a large amount of items that we would like to order, or rank in some way such that the items that the user finds more interesting have a higher precedence in the feed. As the use of SNS grows rapidly, so does the demand for such ranking algorithms.

While there exists many SNS currently being used, the scope of this thesis will be reduced to focus on only one of them; Facebook. This is mainly due to the time constraints involved, however the results and methodology used in both our research and implementation will be generalisable to all SNS. There are two main reasons why Facebook has been chosen as our SNS of focus. Firstly, Facebook is currently the most used SNS which allows us to more easily gather users for our purposes aswell as have more confidence that our results can be generalised to most SNS. Secondly, Facebook attracts many different types of users due to it's very flexible, generic nature (i.e. not a niche SNS). Each type of user will have different wants and needs and by having a large set of user types, we are able to more easily identify them.

With all this considered, it is clearly not possible to have one single ranking algorithm to accomodate for all users, and yet as of now, Facebook only offers one ranking, or view of a user's feed (besides chronological order). Our aim can be summarised as follows: Firstly we will set out to identify these different user types and their needs, then we aim to create a number of different ranking algorithms based on the discovered user types. Thus offering a more personalised ranking of a user's feed. It is important to note that we do not aim to create a *better* ranking algorithm than Facebook as an enormous amount of research and time has already been put into creating said algorithm, instead we aim to offer different, more personalised rankings.

Chapter 2

Background

2.1 User Modelling

As mentioned earlier, the first half of our approach is to identify different types of Facebook users and determine what preferences they have in terms of feed objects. We took a look to the literature to see if anyone had attempted user modelling for this purpose. The first question we set out to answer was quite intuitive, that being what are the reasons why people use Facebook? Or even SNS in general. Bon [BRR10] conducted a survey on university students to confirm the results of a research paper published in 2008 which found that the main reasons people used social networks were:

- Keeping in touch with friends
- Meeting new people
- Posting and looking at pictures

Factor analysis was performed on the survey answers and the results were tabulated into a list of loadings, representing how much certain features acted as the reason for the use of social networks. These results however, proved to be too general for our purposes as we are looking to define different, more concrete user types. On top of this, the scope of this Survey was restricted only to university students and contained a relatively small sample size of 200.

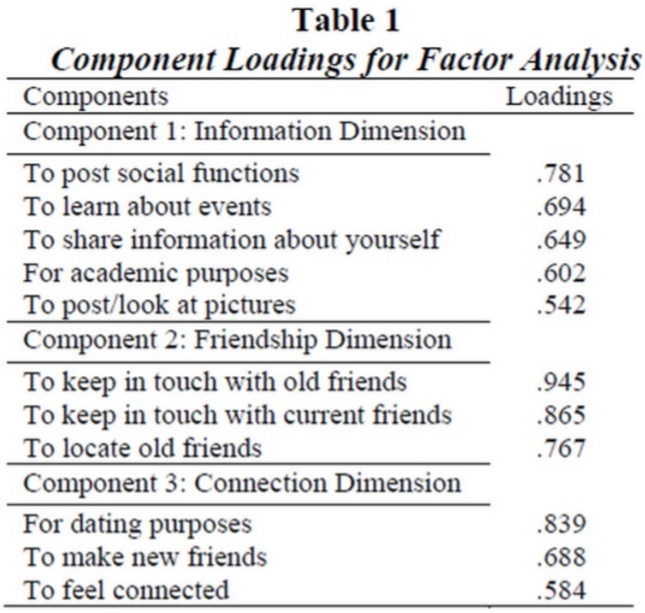

Table 1
Component Loadings for Factor Analysis

Components	Loadings
Component 1: Information Dimension	
To post social functions	.781
To learn about events	.694
To share information about yourself	.649
For academic purposes	.602
To post/look at pictures	.542
Component 2: Friendship Dimension	
To keep in touch with old friends	.945
To keep in touch with current friends	.865
To locate old friends	.767
Component 3: Connection Dimension	
For dating purposes	.839
To make new friends	.688
To feel connected	.584

Figure 2.1: Survey Results

Some interesting correlations were found with further statistical analysis, for example; men significantly used social networks more for the purpose of sharing personal information and dating compared to women. While being somewhat more specific, these correlations are still slightly too generalised. However, this being said, knowing some of the more common reasons that people use Facebook will give us direction in terms of what types of feed objects we should be looking for.

Nad [NH12] published a literature review which looked at user modelling Facebook users from a more psychological frame of view. They found that there were correlations between certain Facebook behavioural patterns and a uer's psychology. Some of the correlations that were found were:

- Users with higher levels of extraversion tended to use Facebook as more of a social tool rather than to replace social activities. They also generally had accumulated more social networking friends and had more addictive tendencies when it came to Facebook use

- Users with higher levels of Neuroticism Levels tended to prefer reading information in their feed whereas users with lower levels of neuroticism preferred seeing photos
- Users with significantly high or low levels of neuroticism tend to be more open to share personal information on their profiles

These findings are closer to what we are looking for, however, it is not easy to determine the psychology of a user for the application of ranking their Facebook feed. However, once again, these results allow us to get a better idea of the different types of feed objects that are to be considered, for example information vs photos.

It eventually became clear to us that even if we were to use these results for our user modelling, it is uncertain whether they are still relevant, as we saw, the survey conducted by Bon [BRR10] set out to confirm results from a study just 2 years previous to his research paper and already found that user's behaviour changed. So we began to look at how people perform user modelling and the practices involved with it. Cle [Cle04] conducted an interview with 4 different HCI proffessionals based in Sweden to gain insight into there methodologies and phylosophies. The following methods of performing user modelling that came from the interviews especially interested us and will most likely be used throughout our implementation:

- Usability Tests - Observing how a user interacts with a system and asking good questions to extract meaningful feedback from the user.
- Surveying - Simply asking existing users questions that will aid us in our user modelling - as seen in the survey conducted by Bon [BRR10]
- Conceptualise how users are represented in the system (as data) - By thinking about how a user's preferences will be represented in the system itself, we are able to design our algorithms accordingly

- Creating personas and use cases/scenario - Designing a fictional character who would benefit from our system and creating a use case around it; this allows us to consider all the different types of users that can be modelled

With this information in mind, we have chosen to conduct another survey similar to the ones performed in the literature; to help deermine whether there are certain patterns in feed preferences for users and to group those patterns into user types. For example, if we find that there is a pattern of users that prefer to see friend's social updates, photo's and event information, we can classify those set of preferences as a user type. The survey questions will be formulated with ideas from the past surveys that have been conducted and will serve as the basis of our user modelling.

2.2 Ranking Algorithms

A ranking algorithm will give each item a score and order them with the item with the highest score at the top and the item with the lowest score at the bottom. The score of an item will depend on a set of criteria that the ranking algorithm uses. Social networking services will use these ranking algorithms in order to provide the user with items that will interest them. Since we are focusing on Facebook, the users will be provided with a feed that contains a lot of posts that they will receive. In Li [LTL+10] paper, they discovered that there are three major factors that could affect how interesting a user may find a particular item. They are:

- Topical Preference
- Topological Locality
- Social Influence

Topical reference is the idea that most users are interested in a limited range of topics. Topological locality refers to the fact that users are interested in the topics that their friends like and Social influence basically says that users are interested in famous people such as singers or actors. These factors do provide some insight on what a user likes but could be further generalised to topics that a user likes. The Topological locality does raise an interesting notion, that is, users are more likely to like a post that a friend likes. We can summarize these two into a more general categorization of Topic classification and connections. Topic classification will be classifying the topics that a user may like while connections will be a measure of how 'close' the user is with their friend based on their interactivity.

Topic classification is quite difficult we have to generalise a topic that they may like based on the posts that they receive. In a paper by Bur [BMAC13], they analysed twitter tweets and tried to generalise a topic based on the tweets each user received. They have analysed two types of methods. They are:

- BestOverlap
- UserInfoBigram

The BestOverlap method attempts to gather a huge amount of tweets and look at the common words in those tweets. A topic can be generated by the word is overlapped the most across all the tweets. In regards to our algorithm where we have to look at Facebook posts, the likelihood of word overlaps across a large amount of posts is quite low. This method would not be appropriate for our purposes.

The UserInfoBigram analyses the optional text that is provided in every tweet and generalises a topic from those words. In Facebook, almost no one uses the optional text so this method will also not work.

In order to do topic classificication we looked to a paper by Szo [SAC$^+$08] who proposed a system for defining topics of interest with the aid of Wikipedia. He collated a list of tags using a user's "tagging activity" which, for our purposes could mean commonly used words in past posts. These tags are then verified using Wikipedia to check that there indeed does exist something along the lines of the tag. Wikipedia is chosen over more formal dictionaries due the the nature of social media tags not being real words, thus many of the tags will not be able to be verified by formal dictionaries such as WordNet. Wikipedia is then traversed (from the tag's wiki page) to find a super category that the tag belongs to. For example if a user generates the tag "C++", in Wikipedia, the C++ page is a subpage of the "Programming Language Families" page.

Since a single verified tag can have many categories and the final list ends up being dominated by the broader categories, only categories that meet either of these criteria are taken:

- The tag has only 1 category
- The category matches the tag name exactly
- The category is a plural of the tag

This approach nets us a list of the users general interests, which can then be used to determine how much a certain post is related to the user's interests by observing the contents of the post and determining if they belong to any of these topics, again using Wikipedia.

Aga [ACG+14] discusses activity ranking for LinkedIn which is also a Social Networking Service. They discover two more factors that have a huge impact on whether the activity is deemed interesting or not. They used the measurement of CTR or click-through-rate which is the probability of a user clicking on the link to measure the appeal of an activity. An activity that was old had a low CTR compared to an activity that was new. It seems that the freshness of an activity or the time that the activity was made had to be taken into account in the ranking algorithm. In the graph below, we see a drop in the CTR as time progresses.

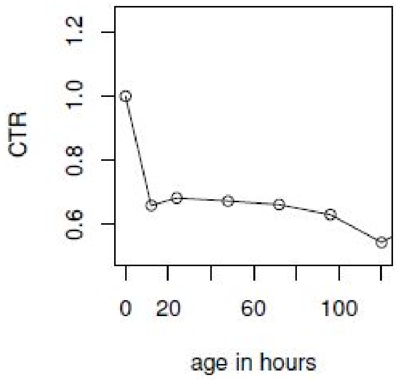

Figure 2.2: CTR vs Time Graph

Another factor was diversity. A huge drop in CTR was found when they gave users a repeated type of activity in their feed. We can see this effect in the graph below. Actor repetitions refer to activites that are posted by the same user appearing in our feed.

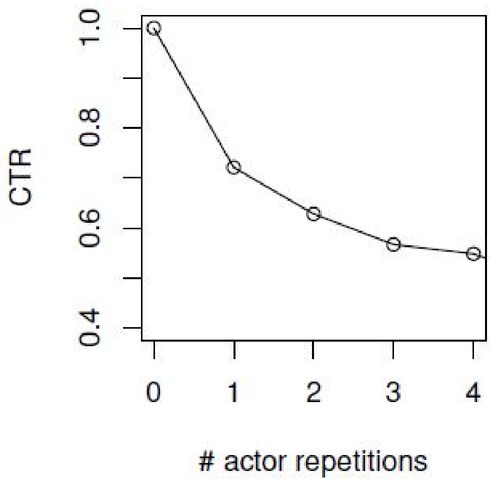

Figure 2.3: CTR vs Actor Repetitions

We can surmise that freshness and diversity are key factors that must be considered in our ranking algorithm. The method that they have used to deal with these two issues involved re-ranking the feed with a decay factor to account for time and adding a negative score to activities of the same type. For our algorithm, we plan to utilise the same methods proposed as they have been successful.

Aga [ACG+14] also reinforces the idea that people are interested in what their friends like when they analysed the activities of co-workers and colleagues. Like Li [LTL+10], they found that there was an increase in the click-through-rate of activities they were made by co-workers in the same organisation and colleagues. This emphasizes the importance of the factor of connections. We can see a huge increase in CTR in the graph below if the viewer has some relationship with the actor.

Viewer-Actor Relationship	CTR Lift
Same Company	66%
Same Job Function	25%
Same Industry	33%
Same Geo Region	24%

Figure 2.4: Table of Relationships with CTR

Chapter 3

Design

3.1 User Modelling

To determine the different user types that users of our system will be able to select from, we will perform a survey spanning a variety of different topographies as mentioned in the Background Research section. If the survey results do not yield any significant preference patterns for users, we can fallback to asking users to complete a short version of our survey upon login to our system. This way, we can still generate a list of preferences for each individual user - the only tradeoff is less convenience for the user.

- Discuss the survey and analyze results? ...

3.2 System Architecture

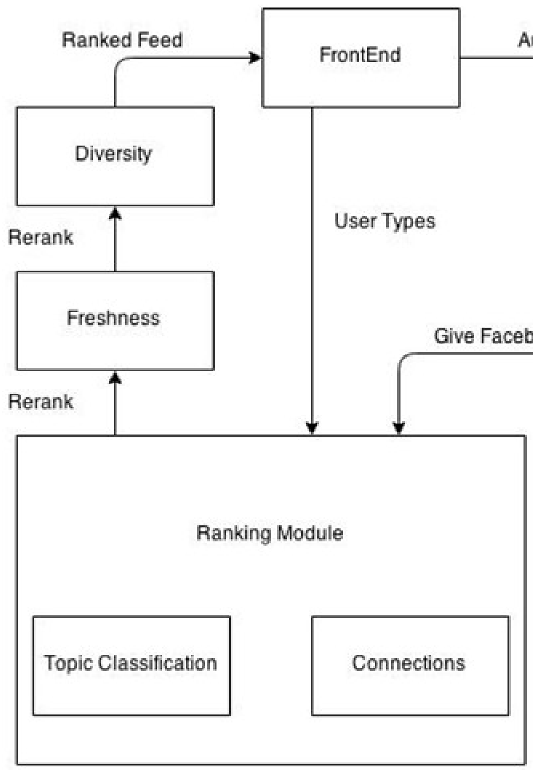

Figure 3.1: Block Diagram

Figure 3.1 provides a general overview of our implementation plan. It is a block diagram of our system.

In our design, we will have a front end module which will be the website that is seen be the user. The website will have a login screen for user authentication which will allow us to pull the data from their Facebook feed. This is the job of the puller module. The user will be given a couple of options of which user type that they think they are. The associated weights from the user types will be brought into the ranking module which contains two algorithms, one for topic classification and one for connections. The puller module will pull the feed in data into this ranking module which will rank the feed using the total scores from the topic classification, connection module and the assigned weights of the user types that came from the website or front end. To do topic classification we will look at the user's posts and try to generalise the topic based on what they have written. We will use the method proposed by Szo [SAC+08] to do topic classification. The connections module will simply look at how often the user has interacted with the person who posted that item and give a score based on that.

After the scores have been assigned to each post, the feed will be ranked on the score and passed over to the freshness module. We will use Aga's [ACG+14] method and assign a decay factor to feeds that are not as recent. This feed will be reranked based on the new scores and passed over to the diversity module. In this module, we will rerank the feed and add a negative score to consecutive posts of the same type. Lastly, the newly ranked feed will be displayed in the frontend in front of the user.

We plan to use nodejs to do the whole project due to its flexibility and adaptability. We will use some written nodejs API's in order to interact with the Facebook API.

Chapter 4

Implementation

4.1 Approach

Our aim was to create a more personalized view of the Facebook feed. The top down approach was used to break the problem into smaller modules and to create a solution for each module. The problem was broken down into the following parts:

- Pulling data from the Facebook feed
- Ranking Algorithm
- User Interface

To implement this solution we have used nodejs due to it's flexibility and diversity. Nodejs also has many modules that allows simpler and easy to understand interactions with the Facebook API.

4.2 Facebook API and App

Facebook requires the creation of an app in order to allow a website to pull any data from Facebook. In order to create the app, we must first sign up as Facebook developers. The app that is created will have a unique App ID and secret. This is used to let Facebook know that our app is being used when we make calls to Facebook's API.

To grab the feed data from Facebook we utilized the nodejs module written by Thuzi. This module communicates with the Graph API provided by Facebook that is used to gather information from the feed. Another module that was used was the passport-facebook module written by Jared Hanson. This module was mainly used to allow ease of authentication for Facebook. This means that we do not directly control the authentication and we are not capable of storing usernames or passwords. Facebook sends the data in JSON format which is convenient for us since nodejs can easily handle data in this format.

Before we can begin pulling data from the feed we first need to get permissions from the user about which data that we need. The permissions that we desired were:

- read stream
- user likes
- read mailbox

The reason for getting each of these permissions will be explained in the ranking algorithm.

4.3 Ranking Algorithm

The algorithm for ranking the feed from Facebook comes from the combination of many other algorithms that have been researched. The algorithm contains the following key components:

- Topic Classification
- Connections
- Freshness
- Diversity
- User Modelling

Each set component will provide either a positive or negative score to each feed post that we have received from the feed.

The Topic classification module followed an idea from Szo [SAC+08] where a search of Wikipedia can be used on tags in order to generalise the topic. We decided to go through the user's likes and to do a wikipedia search in order to determine the topic of interest. To get this data we require the read likes permission from Facebook. The Wikipedia search used a nodejs module called wikipedia-js written by kenshiro.

The Connections module was based on the idea from Li [LTL+10] where a user may prefer a feed item that their friend likes or posts. Our implementation of this module involved looking at the friends that the user has recently messaged. This required the read mailbox permission from Facebook. We do not directly look into the message but do consider the friend that the user has recently talked to.

The Freshness module implementation followed an idea from Aga [ACG+14] where we assigned an intial score to each post and decreased the score depending on the post's creation time. Aga [ACG+14] also provided an implementation for the Diversity module where we apply a negative score to consecutive posts that are similar.

User modelling came from Bon [BRR10] and Nad [NH12] who concluded users have different reasons for using Facebook. This module hopes to find a generalized user and adapt the feed to their particular needs. Some types of users we generalised were:

- A **Socialite** is a user who mainly uses Facebook to see posts from friends or families.

- A **News Reader** is a user who mainly uses Facebook to keep up-to-date with the news.

- A **Follower** is a user who uses Facebook to see posts from organisations.

These may not be the only types of users so we had conducted a survey and used a survey distributor called Survey Monkey to distribute the survey. We received over 100 responses and have analysed these responses for any notable patterns. In this analysis we have concluded that the three types we have are suitable. There is a chart below showing the different types.

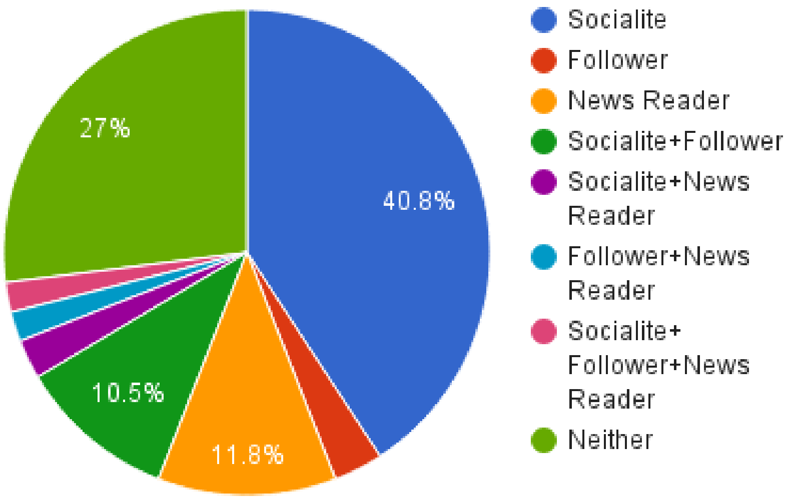

Figure 4.1: Welcome screen

4.4 User Interface

A typical user that uses our app will first be welcomed with a welcome screen as shown in the figure below.

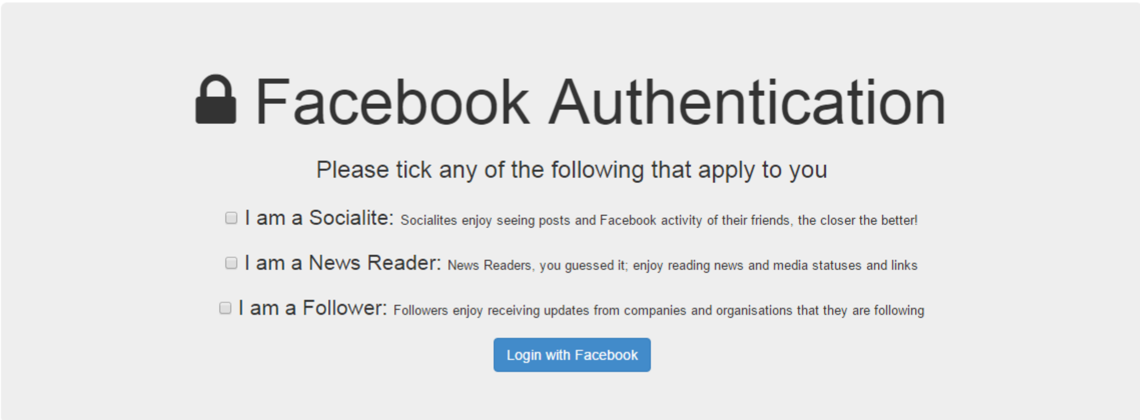

Figure 4.2: Welcome screen

The user can choose one or more of the options before they click log in. When the log in is clicked, the user is taken to a typical log in screen from Facebook.

After logging in, the user will be able to see a Facebook feed.

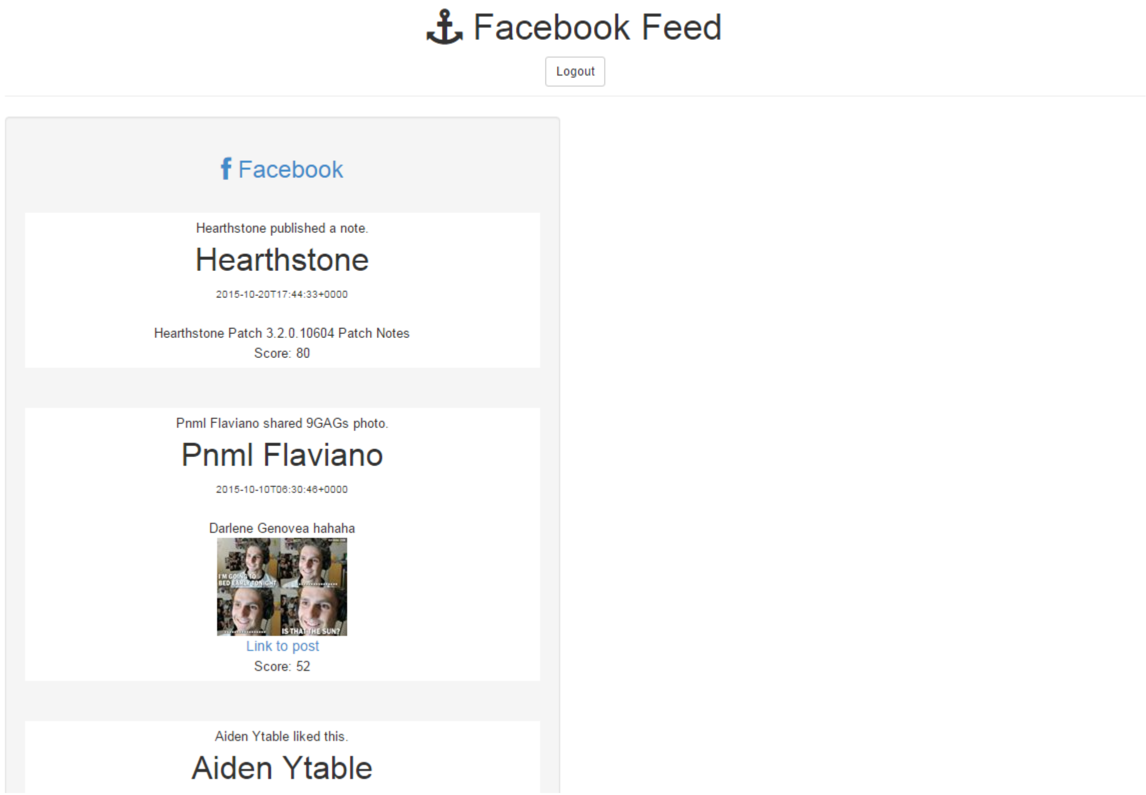

Figure 4.3: Welcome screen

The user interface is not the main focus of this thesis so it definitely could be improved. (Should i say this?)

4.5 Problems Encountered

- Facebook API Limitations
- Paging
- Asynchronous vs Synchronous

A typical user receives an enourmous amount of items to their feed daily. This means that we would be required to pull a large amount of feed items in order to decide which items that the user would prefer to see. Unfortunately Facebook has a limitation on how many API calls can be made in a period of time. Only 600 API calls can be made every 10 minutes. This limitation has prevented us from getting more data about the user such as searching through the feed to see if a friend has liked a particular post. Another limitation of the Facebook API involved the friends list. The Facebook API only allows us to get friends who have also installed the app. For our particular time frame, this is very impractical. Facebook API also does not show us how close the friends are to the user.

Chapter 5

Evaluation

This chapter is mainly provided for the purpose of showing a typical thesis structure. There are no more thesis requirements described.

5.1 Evaluation Method

We considered two different ways of evaluating our system.

The first method was to simulate real users by creating Facebook accounts and attempting to mimic behaviour of each user type. We would then create a ground truth regarding how that type of user would like their feed ranked and compare the output of our ranking algorithm to the ground truth. We found quite a few flaws in this method, the major one being how difficult it would be to simulate a real user. Creating social interactions and simulating connections between users would prove very difficult. On top of this, the ground truths that we would be creating could be affected by confirmation bias. This left us with a very questionable evaluation method, so we arrived at our second one.

The second, and chosen evaluation method takes the form of gathering real users and performing a form of usability test. In this test, we will ask participants to order their feeds how they would like it to be seen, this forms an unbiased ground truth. We then run our ranking algorithm on their feeds and compare the ground truth they gave us earlier to the output. In addition to this ground truth comparison, we will ask the user to compare our ranking algorithm with the one provided by Facebook, without telling them which is which. This will give us some subjective results as to whether our ranking algorithm has succeeded in personalising the user's feed.

5.2 Results

The result of this work is the present document, being both a LaTeX template and a thesis requirement specification.

- The results we got from our test subjects ...

5.3 Discussion

The Dual function of this document somewhat de-emphasises the primary purpose of the document, namely the thesis requirements. It would be better, if these could be stated on a few concise pages (cf Appendix 1, p**??**).

- Discuss stuff!! ...

Chapter 6

Conclusion

A thesis requirements/template document has been created. This serves the dual purposes of giving students specific requirements to their theses — both style and content related — while providing a typical thesis structure in a LaTeX template.

- Do we need a conclusion for thesis A ??
- Summarize what we have so far?? ...

6.1 Future Work

Extract the requirements from the template in order to have very concise requirements.

- Possible stuff to extend our thesis ...

Bibliography

[ACG+14] Deepak Agarwal, Bee-Chung Chen, Rupesh Gupta, Joshua Hartman, Qi He, Anand Iyer, Sumanth Kolar, Yiming Ma, Pannagadatta Shivaswamy, Ajit Singh, et al. Activity ranking in linkedin feed. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1603–1612. ACM, 2014.

[BMAC13] Matthew Burgess, Alessandra Mazzia, Eytan Adar, and Michael J Cafarella. Leveraging noisy lists for social feed ranking. In *ICWSM*, 2013.

[BRR10] Jennifer Bonds-Raacke and John Raacke. Myspace and facebook: Identifying dimensions of uses and gratifications for friend networking sites. *Individual Differences Research*, 8(1):27–33, 2010.

[Cle04] Torkil Clemmensen. Four approaches to user modelling-a qualitative research interview study of hci professionals' practice. *Interacting with computers*, 16(4):799–829, 2004.

[LTL+10] Huajing Li, Yuan Tian, Wang-Chien Lee, C Lee Giles, and Meng-Chang Chen. Personalized feed recommendation service for social networks. In *Social Computing (SocialCom), 2010 IEEE Second International Conference on*, pages 96–103. IEEE, 2010.

[NH12] Ashwini Nadkarni and Stefan G Hofmann. Why do people use facebook? *Personality and individual differences*, 52(3):243–249, 2012.

[SAC+08] Martin Szomszor, Harith Alani, Ivan Cantador, Kieron OHara, and Nigel Shadbolt. *Semantic modelling of user interests based on cross-folksonomy analysis.* Springer, 2008.