<table>
<thead>
<tr>
<th><strong>Title</strong></th>
<th>Recommending user connections by utilising the real-time Web</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Authors(s)</strong></td>
<td>Hannon, John</td>
</tr>
<tr>
<td><strong>Publication date</strong></td>
<td>2014</td>
</tr>
<tr>
<td><strong>Publisher</strong></td>
<td>University College Dublin. School of Computer Science and Informatics</td>
</tr>
<tr>
<td><strong>Link to online version</strong></td>
<td><a href="http://dissertations.umi.com/ucd:10028">http://dissertations.umi.com/ucd:10028</a></td>
</tr>
<tr>
<td><strong>Item record/more information</strong></td>
<td><a href="http://hdl.handle.net/10197/6789">http://hdl.handle.net/10197/6789</a></td>
</tr>
</tbody>
</table>
UNIVERSITY COLLEGE DUBLIN

Recommending User Connections by Utilising the Real-Time Web

by

John H. Hannon B.Sc.

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the
College of Science
School Of Computer Science & Informatics

Head Of School: Mr. John Dunnion
Principal Supervisor: Prof. Barry Smyth
Mentor: Dr. Kevin McCarthy

July 2014
Declaration of Authorship

I, John Hannon, declare that this thesis titled, ‘Recommending User Connections by Utilising the Real-Time Web’ and the work presented in it are my own. I confirm that:

■ This work was done wholly or mainly while in candidature for a research degree at this University.

■ Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.

■ Where I have consulted the published work of others, this is always clearly attributed.

■ Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.

■ Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: 

Date: 

i
“People think that computer science is the art of geniuses but the actual reality is the opposite, just many people doing things that build on each other, like a wall of mini stones.”

-Donald Knuth
Social media services, such as Facebook and Twitter, thrive on user engagement around the active sharing and passive consumption of content. Many of these services have become an important way to discover relevant and interesting information in a timely manner. But to make the most of this aspect of these services it is important that users can locate and follow the most useful producers of relevant content. As these services have continued to grow rapidly this has become more and more of a challenge, especially for new users. This problem can be solved in principle by constructing a recommendation system based on a model of users' preferences and interests to recommend new users worth following.

In this thesis we propose a recommendation framework for friend finding. It is capable of integrating different sources of user preference information that is available through services such as Twitter and related services. It is also designed to provide a natural partitioning of user interests based on those topics that are core to the user versus those that are more peripheral and the social connections linked with the user. This provides access to a range of different types of recommendation strategies that may be more helpful in focusing the search for relevant users according to different types of user interests. We demonstrate the effectiveness of our approach by evaluating recommendation quality across large sets of real-world users.
Acknowledgements

I would like to firstly thank Prof. Barry Smyth whose patient guidance and encouragement has helped me develop from a student to a researcher. I would also like to thank Dr. Kevin McCarthy who throughout my PhD has been there to aid me and help me with any research questions I’ve had. To both of you I am grateful and without your helpful steering much of this work would not have been possible. I would like to thank Science Foundation Ireland via grant 07/CE/I1147 and Amdocs Inc., who have funded my PhD and made it possible for me to carry out my research. I would also like to thank my parents Harry & Sylvia, who have always encouraged me to continue with education and to fully reach my potential, for both of you I am truly thankful. I would also like to especially thank my aunt Patricia, whose spell checking skills have been called upon numerous times throughout this PhD. I promise to never send on anything to be proof read again. To my uncle Noel and the rest of my family you have given me a strong foundation from which I could build upon and for that I will always appreciate you. To my colleagues in the Clarity Centre and specifically the recommender systems research group, we’ve had some trip together! One could not look for a better set of colleagues, whom now I class as my closest friends. Most importantly, I must thank my fiancée Ashlyn who has restrained herself from killing me as I’ve neared the final stages of this PhD. You are truly one in a billion and I’m grateful for your love and support.
# Contents

Declaration of Authorship ........................................... i
Abstract ........................................................................ iii
Acknowledgements ....................................................... iv
List of Figures ............................................................... ix
List of Tables ................................................................... xiii
List of Publications ......................................................... xiv

1 Introduction .................................................................. 1
   1.1 The Real-Time Web ................................................. 4
   1.2 Motivating Examples ............................................... 11
   1.3 Core Contributions .................................................. 12
   1.4 Thesis Overview .................................................... 14

2 Background .................................................................. 15
   2.1 Introduction .......................................................... 15
   2.2 Recommendation Systems ....................................... 16
      2.2.1 Content-based vs. Collaborative-filtering ............ 18
      2.2.2 Items as Ratings and Features ......................... 19
      2.2.3 A Generic Recommendation Architecture ............ 20
   2.3 Content-based ....................................................... 21
      2.3.1 Structured Content ........................................... 22
      2.3.2 Unstructured Content ........................................ 26
   2.4 Collaborative-filtering ............................................ 32
2.4.1 Prediction vs. Recommendation ................. 33
2.4.2 User-Based ........................................ 33
2.4.3 Item-Based ......................................... 38
2.5 Pros & Cons ............................................ 40
  2.5.1 Content-based ................................... 40
  2.5.2 Collaborative-filtering ......................... 41
  2.5.3 Towards Hybrid Recommendation Systems .... 44
2.6 Evaluating Recommendation Systems ............... 46
  2.6.1 Validation ........................................ 47
  2.6.2 Prediction Metrics ............................. 48
  2.6.3 Recommendation Metrics ...................... 49
  2.6.4 Beyond Accuracy ................................ 50
2.7 Friend Recommendation Systems ................. 54
2.8 Summary .............................................. 56

3 Tweet-based Profiling & Recommendation ........... 58
  3.1 Introduction ....................................... 58
  3.2 Profiling User Interests ......................... 60
    3.2.1 From Tweets to Profiles .................... 60
    3.2.2 Harnessing The Social Graph ................. 62
  3.3 Recommending Social Connections ................ 64
  3.4 Evaluation ......................................... 67
    3.4.1 Dataset ....................................... 67
    3.4.2 Methodology .................................. 69
    3.4.3 Detailed Analysis ............................ 71
    3.4.4 Beyond Precision and Coverage ............. 74
    3.4.5 Lessons Learned .............................. 80
  3.5 Conclusions ........................................ 82

4 From Tweets to Tags .................................. 84
  4.1 Introduction ....................................... 84
  4.2 From Tweets to Tags .............................. 88
  4.3 Tag-based Interests ................................ 88
    4.3.1 User-tags ..................................... 89
    4.3.2 List-tags ..................................... 90
  4.4 Evaluation ......................................... 93
    4.4.1 Dataset ....................................... 94
    4.4.2 Methodology .................................. 97
## Contents

6.7.1 Transitioning To Other Contexts .............................................................. 153  
6.7.2 Evaluating Cold-start ................................................................................. 153  
6.7.3 Towards Improving Precision Metrics ....................................................... 154  
6.7.4 Utilising Alternative Approaches ............................................................... 154  
6.8 Closing Comments ....................................................................................... 155

A Evaluating New User Cold-start ................................................................. 157  
A.1 Introduction ................................................................................................. 157  
A.2 Evaluation .................................................................................................... 158  
A.2.1 Search Relevance: MAP ........................................................................... 158  
A.2.2 Relevance Positions .................................................................................. 160  
A.3 Discussion & Conclusions ........................................................................... 161

B Evaluating Friend Recommendation Metrics .............................................. 162  
B.1 Introduction ................................................................................................. 162  
B.2 Friend-of-a-friend Precision ....................................................................... 163  
B.3 Discussion & Conclusions ........................................................................... 167

Bibliography ...................................................................................................... 168
List of Figures

1.1 Graphic of Twitter’s User Interface for User @johnhenryhan-non Displaying his Timeline of Tweets. .......................... 3
1.2 Proliferation of Activities on the RTW (Infographic by Shang- hai Web Designers 2011 (http://www.go-globe.com)). ............ 5
1.3 Geo-tagged Tweets since 2009 from Users in Europe, By Miguel Rios (@miguelrios). ................................. 7
1.5 How Users Connect with Each Other Across Various RTW Platforms. ...................................................... 12

2.1 Amazon Product Recommendations Page Showing Recommendations Based on a User’s Profile of Prior Purchases. 17
2.2 Generic Architecture for Recommendation Systems. ...... 20
2.3 Example of Structured Content Source with Features such as Actors, Studio, Release Date. .......................... 23
2.4 Architecture for Case-based Recommendation System. ... 24
2.5 Calculating Similarity Metrics: (a) Represents a Standard Similarity Metric; (b) Represents a Metric that gives Preference to Feature Values that are Lower than the Target Value. [Smyth, 2007] ................................. 25
2.6 Example of Unstructured Content where Features must be Elicited from the Text. ................................. 27
2.7 Architecture for Unstructured CB System. .................... 29
2.8 System Flow for Prediction vs. Top N Recommendation Tasks. 34
2.9 User-Item Matrix Representation of Movie Ratings. ....... 35
2.10 Item Matrix of Movie Ratings with User Similarity Computed on the Rows and Item Similarity Computed on the Columns. 36
2.11 Summary of how the User-based and Item-based Prediction Task Works. ........................................... 39
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>User Tweet from Twitter Profile of @johnhenryhannon.</td>
<td>61</td>
</tr>
<tr>
<td>3.2</td>
<td>Examples of Profiling Information Sources.</td>
<td>63</td>
</tr>
<tr>
<td>3.3</td>
<td>Architecture for our CB Friend Recommendation System.</td>
<td>65</td>
</tr>
<tr>
<td>3.4</td>
<td>Dataset Collection Strategy Started with Initial Seed Set and Expanded based on their Social Connections.</td>
<td>68</td>
</tr>
<tr>
<td>3.5</td>
<td>Average Precision vs. Recommendation List Size for the 5 Different Recommendation Strategies using the 1000-user Test Set.</td>
<td>72</td>
</tr>
<tr>
<td>3.6</td>
<td>Average Relevant Position vs. Recommendation List Size for the 5 Different Strategies using the 1000-user Test Sets.</td>
<td>74</td>
</tr>
<tr>
<td>3.7</td>
<td>Relevant Recommendations for Different Recommendation List Position Ranges.</td>
<td>77</td>
</tr>
<tr>
<td>3.8</td>
<td>An Example of the User Trial Interface using Search Mode for the Query “social search”.</td>
<td>78</td>
</tr>
<tr>
<td>3.9</td>
<td>A Wordle Tag Cloud of the Search Query Terms Submitted during the User Search Trial.</td>
<td>79</td>
</tr>
<tr>
<td>3.10</td>
<td>Relevant Search Results for Different Recommendation List Position Ranges.</td>
<td>80</td>
</tr>
<tr>
<td>3.11</td>
<td>Summary of (a) Average Precision and (b) Position Results across the 4 Content (S1-S4) and 1 Hybrid (S5) Strategies.</td>
<td>81</td>
</tr>
<tr>
<td>4.1</td>
<td>RTW Content Space Representing the Level of Association the Content (tweets/tags) has to a User.</td>
<td>87</td>
</tr>
<tr>
<td>4.2</td>
<td>How we Apply Tags associated with Multiple Lists to Individual Users.</td>
<td>92</td>
</tr>
<tr>
<td>4.3</td>
<td>Three Content Sources from which Recommendation Knowledge is Collected for each User.</td>
<td>95</td>
</tr>
<tr>
<td>4.4</td>
<td>Key Datasets Collected of Core and Expanded Twitter Users.</td>
<td>96</td>
</tr>
<tr>
<td>4.5</td>
<td>Summary Across Sources for all Strategies Combined, Averaged over $k=10$ Splits using Random Subsampling Validation on Samples of 1000 Users.</td>
<td>99</td>
</tr>
<tr>
<td>4.6</td>
<td>Precision (bar) and Coverage (line) Results for Twitter user-tweets based Strategies Averaged over $k=10$ Splits Using Random Subsampling Validation on Samples of 1000 Users.</td>
<td>100</td>
</tr>
<tr>
<td>4.7</td>
<td>Precision (bar) and Coverage (line) Results for Listorious List-tag Strategies Averaged over $k=10$ Splits Using Random Subsampling Validation on Samples of 1000 Users.</td>
<td>101</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>----------------------------------------------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>4.8</td>
<td>Precision (bar) and Coverage (line) Results for WeFollow User-tag Strategies Averaged over k=10 Splits Using Random Subsampling Validation on Samples of 1000 Users.</td>
<td>102</td>
</tr>
<tr>
<td>5.1</td>
<td>(a) Multifaceted User Profile, (b) Multifaceted User Profile of Twitter User @BarackObama.</td>
<td>110</td>
</tr>
<tr>
<td>5.2</td>
<td>Forming a Multifaceted Profile for a User using their Own, their Friends, and their Followers Tags.</td>
<td>111</td>
</tr>
<tr>
<td>5.3</td>
<td>Division of Facets into Regions of Core, Secondary and Tertiary Interests.</td>
<td>114</td>
</tr>
<tr>
<td>5.4</td>
<td>How we Form our Faceted Queries and Profile Indexes from our Multifaceted User Profile.</td>
<td>115</td>
</tr>
<tr>
<td>5.5</td>
<td>Updated System Architecture from Figure 3.3 to Reflect new Queries and Indexes used in the Multifaceted Approach.</td>
<td>116</td>
</tr>
<tr>
<td>5.6</td>
<td>Summary Precision (bar) and Coverage (line) Results Averaged across Recommendation Indexes, Profile Facets, and Interest Sources.</td>
<td>119</td>
</tr>
<tr>
<td>5.7</td>
<td>Precision (bar) and Coverage (line) Results for Query Facets Evaluated on Users’ Indexed on their (a) Core Interests, (b) Core + Secondary Interests Regions.</td>
<td>129</td>
</tr>
<tr>
<td>5.8</td>
<td>Precision (bar) and Coverage (line) Results for Query Facets Evaluated on Users’ Indexed on their (c) Tertiary Interests, (d) Everything/All Interests Regions.</td>
<td>130</td>
</tr>
<tr>
<td>5.9</td>
<td>Summary Analysis of Target User Similarity (TUS) Averaged across Recommendation Indexes, Profile Facets, and Interest Sources.</td>
<td>134</td>
</tr>
<tr>
<td>5.10</td>
<td>Summary Analysis of Inter-recommendation Similarity (IRS) Averaged across Recommendation indexes, Profile Facets, and Interest Sources.</td>
<td>136</td>
</tr>
<tr>
<td>A.1</td>
<td>Mean Average Precision of User Searches Indicating Total Relevant Results from Relevant and Would Follow and Relevant and Wouldn’t Follow.</td>
<td>159</td>
</tr>
<tr>
<td>A.2</td>
<td>Position of User Searches Indicating Total Relevant Results from Relevant and Would Follow and Relevant and Wouldn’t Follow.</td>
<td>160</td>
</tr>
<tr>
<td>B.1</td>
<td>How User’s Friends of Friends are Selected.</td>
<td>163</td>
</tr>
</tbody>
</table>
B.2 Friend of a Friend Precision Results for Query Facets Evaluated on Users’ Indexed on their (a) Core Interests, (b) Core + Secondary Interests Regions. ......................... 165
B.3 Friend of a Friend Precision Results for Query Facets Evaluated on Users’ Indexed on their (c) Tertiary Interests, (d) Everything/All Interests Regions. ......................... 166
List of Tables

3.1 Evaluation Datasets with Median number of Tweets, Words pre tweet, Followers, and Friends. 69
3.2 Summary Statistics for Online User Trial Participants. 75
4.1 Number of Unique IDS in Expanded Users Dataset 96
4.2 Mean Number of Query Terms in Dataset Averaged over $k=10$ Splits using Random Subsampling Validation on Samples of 1000 Users. 97
A.1 Summary Statistics for Cold-start User Trial Participants. 158
A Multi-Faceted User Model for Twitter
John Hannon, Kevin McCarthy, Barry Smyth
In Proceedings of the 20th International Conference on User Modeling, Adaptation, and Personalization (UMAP ’12), pg 303-309

Content vs. Tags for Friend Recommendation
John Hannon, Kevin McCarthy, Barry Smyth
In Proceedings of the 32nd SGAI International Conference on Innovative Techniques & Applications of Artificial Intelligence (SGAI ’12). pg 289-302

Personalized and Automatic Social Summarization Of Events in Video
John Hannon, Kevin McCarthy, James Lynch, Barry Smyth
In Proceedings of the 16th International Conference on Intelligent User Interfaces (IUI ’11). pg 335-338

The Pursuit of Happiness: Searching for Worthy Followees on Twitter
John Hannon, Kevin McCarthy, Barry Smyth
In Proceedings of the 22nd Irish Conference on Artificial Intelligence and Cognitive Science (AICS ’11)
Bridging Recommendation and Adaptation: Generic Adaptation Framework-Twittomender Compliance Case-Study
John Hannon, Evgeny Knutov, Paul De Bra, Mykola Pechenizkiy, Kevin McCarthy, Barry Smyth

Finding Useful Users On Twitter: Twittomender The Followee Recommender
John Hannon, Kevin McCarthy, Barry Smyth
In Proceedings of the 33rd European Conference Information Retrieval (ECIR '11). Ireland pg 784-787

Recommending Twitter Users To Follow Using Content and Collaborative Filtering Approaches
John Hannon, Mike Bennett, Barry Smyth
1

Introduction

“You are what you share.”


Social networks have become a valuable and essential part of the modern web experience [Kleinberg, 2008]. They are now a key platform for the dissemination and consumption of digital information [Chen et al., 2010, Mangold and Faulds, 2009]. The larger social networking services such as Twitter\(^1\) boast hundreds of millions of users who generate thousands of messages per second\(^2\). This influx of data is often referred to as the real-time web (RTW). Similarly, Facebook’s user numbers have surpassed the 1 billion mark, with each contributing posts, photos, comments and likes\(^3\). This vast amount of information can be both a blessing and a curse. It is a blessing in the sense that real-time information from events is available to the consumer as it’s happening, but, one must often search through false stories or facts

\(^1\)http://www.twitter.com
\(^2\)https://blog.twitter.com/2012/human-face-big-data
\(^3\)http://goo.gl/NVxBtk
to find the desired information. Intelligent systems can help guide the end user in fulfilling their information discovery and filtering endeavours, see [Hurlock and Wilson, 2011, Resnick et al., 1994, Shardanand and Maes, 1995, Sriram et al., 2010], and such technology needs to be applied to social networks to improve our ability to benefit from them.

The value of social networks like Twitter is derived from the social connections made by people [Daly et al., 2010, Hannon et al., 2010]. By connecting to the right people, a user can indulge their interests and also discover entirely new topics of interest. In Twitter, a user’s own timeline is comprised of tweets from the people that they have explicitly chosen to follow; see Figure 1.1. A single tweet represents a micro-message or micro-blog with a limit of 140 characters to express one’s thoughts or views, share a link, ask a question, communicate with a friend, etc., (all in real-time). Following a user in Twitter denotes that you wish to consume that user’s content. However, the user must choose their friends wisely as timelines can quickly become polluted with noise. Friend recommendation systems help users connect with people who are likely to serve as a source of relevant and interesting content.

The main benefit of providing useful and accurate friend recommendations is a mechanism for coping with the “information overload” problem ([Toffler, 1984]), where users are presented with an excessive amount of data that can’t be processed by an individual without the aid of filtering tools. By providing recommendations of other users who share your interests you can use these others as your information filters. Without such recommendations our social graphs and friends lists can easily become corrupted or contaminated by noisy users. Twitter users follow friends with like minded interests because they want these friends to find new interesting snippets of information to keep them informed, be it in the form of pictures, webpages
or even breaking news. Interestingly users themselves are also followed by other users across the social network for similar reasons. We believe that users, their friends and followers, and the content they produce, can be harnessed as a rich source of information for the modelling of user preferences and interest data. Since users, their friends and followers share interests and topics, their associated data can be modelled to provide interesting cross sections and intersections of preference profiles. This information can then be used as the basis for a recommendation system.
In the next section we will firstly explore the source of our recommendation knowledge, the RTW and the content distributed across it, and we will explore related and state of the art research which utilises this source of information. In Section 1.2, we will discuss some motivating examples which helped guide this work and helped us in forming our research contributions (RC). These contributions are outlined in Section 1.3. Finally, Section 1.4 provides a general overview of the work carried out in the rest of this thesis.

1.1 The Real-Time Web

The Real-Time Web (RTW) is the great disruption arising out of Web 2.0 [Lewis, 2006]. The coined term “Web 2.0” represents an evident evolution in the history of the web. Service-driven and user-focused websites emerged to meet the needs of users, be it photo sharing (Flickr), music sharing (Soundcloud) or information aggregators (Reddit), to name a few. In Figure 1.2, an infographic of the proliferation of information and the speed at which the RTW grows is showcased. Simply put, the RTW represents a set of technologies/websites where information from these sites is available “as it’s happening”. The RTW represents the new eye witness news, the first responder. As a side effect of the temporal nature of these services, they carry a huge information burden. The RTW is driven by the masses of users commenting, publishing and consuming information, and as such information overload is even more of a real problem for these RTW services. Researchers have started to explore a number of important research contributions by utilising data from these RTW services. This includes work on content analysis [Chew and Eysenbach, 2010], network structure analysis [Java et al., 2007], and user modelling [Abel et al., 2011], all in the pursuit of helping users to better understand and utilise their evolving social
networks. We will pay particular attention to research that has focused on Twitter for a number of reasons. First, this body of research is reasonably representative of the broader set of RTW research. Second, a considerable portion of related research in RTW analysis has been carried out on Twitter, in large part because of the scale and openness of the Twitter eco-system.

![Figure 1.2: Proliferation of Activities on the RTW (Infographic by Shanghai Web Designers 2011 (http://www.go-globe.com)).](image)

Social networks such as Twitter provide an almost free-for-all of their users’ posts and demographic information simply by making a few API calls. There is currently considerable research attention being paid to Twitter and the RTW in general, with many academic conferences such as WWW\(^4\), RecSys\(^5\) having social streams or sessions. Before delving into this research

\(^4\)http://www.iw3c2.org/conferences  
\(^5\)http://recsys.acm.org/
though it’s important to get a handle on the scale at which data is both produced and disseminated across these RTW services. Using Twitter as our example again, it boasts record tweet numbers per second of just under 7,000 tweets at its peak, 50 million a day and around 140 million a week\textsuperscript{6}. These figures are from 2011, so no doubt have increased since then; to give another view of the scale of this data source, see Figure 1.3, this image represents every geo-tagged twitter message from 2009 in Europe\textsuperscript{7}, each dot seen in this image represents a single tweet (billions of them). The relative strength of the colour represents hot spots of activity. This image represents the non-centralised nature of these RTW services. Although, examining the graph some geo-tagged tweets seem misplaced (in the sea), it nonetheless gives an impression of the amount of data being produced in these geographical regions.

All of this information flows at a pace that is far beyond what traditional forms of communication and news broadcasts could hope to generate. Today Twitter is used by many as a form of RSS reader [Phelan et al., 2009], as users follow their favourite bloggers and news organisations. It has also proven to be a very popular way of sharing pages, causing some commentators to speculate about the potential for social media services like Twitter to represent a significant threat to the major search engines as the means by which users discover new content. And, of course, the advertisers and marketers have also recognised the potential of Twitter by focusing in on user influence [Cha et al., 2010], as a way to engage with customers in real-time.

It is no surprise that the recent literature includes a number of interesting analyses of Twitter’s real-time data, largely with a view to developing an early understanding of why and how people are using services like Twitter;

\textsuperscript{6}https://blog.twitter.com/2011/numbers
\textsuperscript{7}https://blog.twitter.com/2013/geography-tweets-3
see for example [Huberman et al., 2008, Java et al., 2007, Jensen et al., 2002, Kwak et al., 2010]. For instance, the work of Kwak et al. [2010] describes a very comprehensive analysis of Twitter users and Twitter usage, covering almost 42 million users, nearly 1.5 billion social connections, and over 100 million tweets. In this work the authors have examined reciprocity and homophily among Twitter users, they have compared a number of different ways to evaluate user influence, as well as investigating how information diffuses through the Twitter ecosystem as a result of social relationships.
and re-tweeting behaviour. The authors found that users on Twitter shared few known characteristics with other social networks, they also noted that the followers relationship in nearly 78% of cases isn’t reciprocated and connections are one-directional. This would indicate users seek out others to meet an information need. When they focused on the influence of users by measuring the number of followers a user had and then comparing this with traditional influence metrics like PageRank [Brin and Page, 1998], they found that the influence scores were quite similar. It wasn’t until they measured the number of retweets a user received that a difference was apparent, pointing to the fact that the quality or popularity of one’s tweets isn’t indicative of the number of followers they have.

One common RTW research theme relates to the role of Twitter as an information filter for users. For example, early work by Shamma et al. [2009] studied the micro-blogging activity that emerged during the 2008 US presidential debates to demonstrate how frequently occurring terms reflected trending debate topics, although vocabulary complexity did tend to obfuscate automatic topic identification. More recently the work of Phelan et al. [2011] demonstrated Twitter as a powerful news aggregator and filter via the Buzzer social news service. Buzzer extracts stories from a user’s RSS-feeds based on the topics being discussed by the user’s Twitter friends/followers. The content of tweets (the user’s own tweets and the tweets of their friends/followers) is used as a term-based interest profile for the user and matched against incoming RSS-feeds to select and rank matching stories. Related work by Garcia Esparza et al. [2010] considered the micro-blogging activity of users (on Twitter & Twitter-like services such as Blippr) as a source of product information to show how recommendations could be adapted to drive recommendation based on products that a user had liked or disliked.
The publicity surrounding Twitter in the recent past has stemmed from its ability to capture the occurrence of major events and happenings around the world. During the 2008 Mumbai attacks eyewitnesses used Twitter to post tens of tweets per second about the unfolding tragedy. Twitter users on the ground helped create a list of the dead and injured, while others posted vital information such as emergency numbers, hospital locations for blood donations. Also, the important role of Twitter in recent events such as the Arab Spring [Stepanova, 2011] or the Fukushima Disaster [Thomson et al., 2012] has received considerable attention in mainstream media; although the precise role that Twitter and similar services have played is still to be fully explored. Twitter and social networks in general are changing the face of news coverage (Figure 1.4) and leading to the emergence of a new type of news service that sources on-the-ground news content from curated communities on Twitter and other social networks.

Certainly there is evidence to support the ability of Twitter to help recognise and amplify real world events. For example, the work of Nichols et al. [2012] has examined using Twitter as a means to summarise sporting events, by using tweet frequency and content as a signal for ‘event spikes’. Related work by Hannon et al. [2011] demonstrated how such techniques could be used to create real-time video summaries of sporting events. This work tracked the increase in tweets around a sporting event (Soccer World Cup matches) and using the increase in the frequency of tweets as top “events” from a match they could build a highlight reel of video clips beginning prior and ending post event spike.

It is evident the pace, scale and amount of information RTW services such as Twitter can transfer/produce and it should also be evident the potential that is available from this data source as an information signal. In the next

---

8http://www.wired.com/dangerroom/2008/11/first-hand-acco
Figure 1.4: Changing Face of News Dissemination via Mashable 2012 (http://goo.gl/dfele6).
section we will explore some motivating examples which helped form our thesis research contributions which we discuss throughout this thesis.

1.2 Motivating Examples

By focusing in on the RTW and the content produced across it we wish to build a friend recommendation system to help users better filter this information by forming better connections to others. Our research is motivated by 3 core ideas. Firstly, we believe there is a need to extract and distill useful information from RTW services, such as Twitter. As research has shown, clearly there are information signals that can be identified from this onslaught of flowing information, but determining what information is useful and to whom it is useful to is our first motivating factor; see [Mathioudakis and Koudas, 2010, Signorini et al., 2011, Vieweg et al., 2010].

Secondly, there should be a clear focus on the content being produced by users of the RTW. Contrary to graph-based metrics which aim to complete close knit social graphs [Silva et al., 2010], we wished to harness one’s interests from the content they produce. Furthermore, by exploring other sources of profiling information, can we build a more effective friend recommendation systems?

Thirdly, we are motivated by the nature in which connections are formed on Twitter. On Twitter you follow other users to consume their tweets. If a user wishes to consume useful information on a topic they will choose to follow others who tweet about that topic. This connection building is befriending a user, in Twitter the user that has become befriended does not have to reciprocate the connection unless of course they find that user of interest to them. Could there be information to be gained here from these relationship
and, if so, what? In Figure 1.5 we show how connections are formed in Twitter; this asymmetric nature varies from the likes of other social networks such as Facebook, where both parties must consent to friendship and as such has an interesting impact on group dynamics.

![Figure 1.5: How Users Connect with Each Other Across Various RTW Platforms.](image)

In the next section we will formally outline the research contributions which we will answer throughout this thesis.

### 1.3 Core Contributions

When Twitter first emerged it offered very rudimentary search services to help people find new users to follow, even now the system explores a user’s
close social network and suggests users within one degree of separation of
that user. This represents a significant opportunity for a friend recommend-
dation system to help find useful users across the Twitter network. Our
basic assumption is that the Twitter activity of a user’s social graph (their
tweets, and the tweets of their followers, and friends) provides a powerful
source of profile information that can be used as the basis for recommenda-
tion. The availability of large quantities of content and the need for better
friend-finding functionality clarifies 4 key research contributions:

**Contribution 1 (RC 1)**
Can we develop an effective friend recommendation system that is
guided by user profiles, which are based on what users tweet about
and that allows us to recommend useful users to follow?

**Contribution 2 (RC 2)**
Can we exploit the structure of the social graph to aid recommenda-
tion, by looking beyond a user’s own tweets?

**Contribution 3 (RC 3)**
What other sources of content can be harnessed? Can we transition
beyond noisy tweet terms to tag-based profiles and in turn utilise these
profiles for friend recommendations?

**Contribution 4 (RC 4)**
How can we accommodate the complex and diverse interests of users
during friend recommendation to ensure that we can generate recom-
mendations that reflect these varied interests, rather than recommenda-
tions that are dominated by some single core interest?

The next section outlines how this thesis will flow, in relation to which
chapters aim to answer our identified research contributions.
1.4 Thesis Overview

In the following chapters we discuss and evaluate the contributions laid out in Section 1.3. The overall structure of this thesis is as follows; Chapter 2 comprises a review of relevant background research from recommendation systems and, more specifically, will focus in on friend-based recommendation systems. Chapter 3 introduces our user modelling framework for recommendation; we introduce the notion of harnessing users’ interests spawned from the content a user produces or that of the people who they choose to follow. These notions will help us in examining our RCs 1 & 2. Following that, in Chapter 4 we detail a large scale evaluation, again focusing in on RC 1 & 2 and how recommendations are formed, but now we will explore if we can utilise any other sources of recommendation knowledge for profiling users, which is in line with RC 3. A rigorous collection of experiments will evaluate our newly identified sources and their applicability at forming recommendations. Chapter 5, builds upon our modelling techniques by examining the diverse set of interests a user has and in line with RC 4 we ask how can we model those interests and factor them in when forming recommendations? We then incorporate this new model into our recommendation framework as described in Chapters 3 & 4 and evaluate its efficiency at recommending useful connections. Finally, in our conclusions chapter, Chapter 6, we will discuss the benefits and limitations of the work carried out in this thesis by showcasing an in-depth summation of our core contributions.
2

Background

2.1 Introduction

Recommendation systems are the driving force behind many modern web experiences both commercial and social. Sites such as Amazon\(^1\), Netflix\(^2\), Facebook\(^3\) and LastFm\(^4\) automatically adapt to the usage patterns of users by using user profiling, personalisation and recommendation technologies to deliver more personalised web experiences [Armentano et al., 2011, Schafer et al., 1999]. For example, a typical Amazon product page includes a variety of product recommendations based on our past purchase, browsing histories, the item being considered etc. Netflix and LastFm influence our entertainment consumption by recommending movies, TV shows and music playlists informed by our recent activities.

\(^1\)http://www.amazon.com
\(^2\)http://www.netflix.com
\(^3\)http://www.facebook.com
\(^4\)http://www.lastfm.com
By employing the use of recommendation systems to proactively deliver the right information at the right time, these sites can increase sales, promote user engagement and entice users to return by providing better suggestions [Sarwar et al., 2000, Schafer et al., 1999]. Figure 2.1 shows a sample set of recommendations from Amazon for a user. These recommendations are personalised because they are based on a user’s profile which contains prior purchases and/or products owned by that user. The ranked list of recommendations are based on other Amazon users who have similar purchase histories to that user, but who also have other items in their profiles which the target user does not. This screenshot also depicts some supporting features which both aid in recommendation quality and better inform the user. In Figure 2.1, feedback is presented to allow the user to tweak recommendations such that, if a suggestion is not to their taste they will not be shown similar in the future. Explanations aid the user in making an informed decision by explaining how a recommendation was formed for this suggestion (e.g. suggesting an iPad cover because the user previously bought an iPad).

In the following sections we will explore recommendation systems in more detail. In turn we will be focusing on exploring research such as content-based (CB) recommendation [Pazzani and Billsus, 2007], which will be the focus of much of this thesis, to complementary collaborative-filtering (CF) based [Resnick et al., 1994] approaches. Moreover, we will then discuss the core area of recommendation research in which this thesis positions itself, namely, social connection finding or “friend recommendation”.

2.2 Recommendation Systems

Recommendation systems have come a long way from their origins in systems such as Tapestry [Goldberg et al., 1992] a mail system and one of
Figure 2.1: Amazon Product Recommendations Page Showing Recommendations Based on a User’s Profile of Prior Purchases.

the first recommendation systems, which coined the term “collaborative-filtering”. Goldberg et al. [1992] recommendation systems allowed users to express their preferences by tagging items that could then be recommended to others based on input queries made up of preference terms. Recently, recommendation systems research has gone through a huge growth period partly incentivised by of the likes of the Netflix prize\(^5\), recommendation systems can now provide not only accurate, but diverse [Smyth and McClave, 2001] and serendipitous suggestions [Iaquinta et al., 2008] to their users to enrich their web experience. At the core of recommendation systems is the

\(^5\)http://netflixprize.com
idea for the potential to suggest an item that may be of interest to a particular user/s. A recommendation system builds a model of the users of these sites, this model is often referred to as the user profile. This profile is built using recommendation knowledge from the user or indeed the products (items) to compute a similarity between what a user has purchased previously and other available products.

In the following sections we will first briefly explore recommendation systems in the context of CF and CB systems. Both approaches to recommendation harness different knowledge available to the system when performing recommendation to the user. In part we will also highlight the core resources utilised by both approaches namely user ratings and item features. We will conclude this high level overview by discussing the generic features shared by all recommendation systems before we go on to delve deeper by discussing CB, CF research in detailed in later Sections (2.3 & 2.4).

### 2.2.1 Content-based vs. Collaborative-filtering

Both CF and CB approaches to recommendation tackle the recommendation problem from different viewpoints. For instance a CF recommendation system aims to simulate the action of “word of mouth” through algorithmic means [Shardanand and Maes, 1995]. By modelling users and items, a neighbourhood can be formed of likeminded others who like similar items. CF systems recommend items that users with similar rating patterns to you have liked. Whereas a CB system should be used when concrete assessment ratings aren’t present, but, item features are present in the descriptions of items or in raw text associated with items. A CB system recommends items that are similar to items that you have liked before. Both of these approaches to recommendation have pros and cons, such as data sparsity,
cold-start present in CF or noisy text, inaccurate features with CB. We will discuss these pros and cons in more detail in Section 2.5.

2.2.2 Items as Ratings and Features

At the core of a CF system are sets of user ratings for items. Ratings come in the form of unary, binary, or numeric scaled ratings. Unary denotes that a user has seen an item and rated it or the item is unknown, for example a Facebook “Like”. Binary is similar, although it has a positive or negative rating per item (Youtube’s thumbs up or down on videos). Finally, the numeric scale approach provides the users with a set of numeric ratings to show their preference towards an item (Netflix movie ratings). These ratings have no extra recommendation knowledge associated with them and as such, they represent a preference and nothing more towards an item.

In a CB system an item is represented differently. Items are traditionally represented as a set of descriptive features. These features describe an item and provide, along with a user’s preference for an item, extra knowledge which can be utilised to find similar items. For instance, a simple example might be a user who likes a lot of genre:comedy and genre:horror movies with actors such as actor:Tom Cruise all of these features genre, actor are part of his profile, these features can be used to find other items which match the user’s profile of features.

Both CF and CB systems make assumptions about the link between users and items. In Sections 2.3 and 2.4 we will explore these differences and similarities in more detail when discussing each approach to recommendation.
2.2.3 A Generic Recommendation Architecture

Before we delve deeper into the workings of recommendation systems we can firstly look at the generic parts which make up both types. Figure 2.2 illustrates these three generic core parts which are:

1. User Profile: A profile is formed for the user which represents their likes/preferences towards an item/s or information about an item/s. In CB this is information about the items (e.g. descriptions, etc.).
CF there is no information about items just preferences towards item identifiers (e.g. ratings).

2. Recommendation Knowledge: Extra information about an item can be utilised to better form recommendations. In CF no item information is known outside of the preference rating. In CB items descriptions, reviews, commentary can all act as a basis for extra knowledge.

3. Recommendation Algorithm: An algorithm which will compute similarity between users/items. In CF part of the algorithm involves creating a set of possible similar neighbours based on the users’ preferences. In CB, similarity is usually computed against a user profile. This could be a set of features compared with a search query of terms.

In the following sections we will discuss these 3 generic parts of a recommendation in the context of the approach we are exploring, namely CF or CB. In the next section though we will firstly examine one branch of recommendations systems CB research which is core to the work discussed in this thesis and revolves around the content associated with items.

2.3 Content-based

Content-based recommendation systems (CB) [Lops et al., 2010] base their recommendation on rich representations of items. Unlike CF approaches where items are represented as simple atomic structures, typically just a unique item id, in CB recommendation systems items are associated with detailed descriptions using either structured or unstructured data. Pazzani and Billsus [2007] introduces these two main sources of content and we will discuss how recommendation are formed for each in Sections 2.3.1 & 2.3.2
2.3.1 Structured Content

Structured CB systems utilise highly descriptive sets of features associated with items. In Figure 2.3, by way of an example of where features are sourced from we show a movie poster. There are many structured features which can be taken from this poster and used to model the item, a movie in this case. Features such as Actors, Studio, Budget, Release Date, etc. are all descriptive features of this movie and each can be used and compared against for similarity to other sets of items. The most common approach to utilising these sets of features is to develop a case-based recommendation system [Bridge et al., 2005]. Each item and its features are represented as a case that can be compared against other cases for similarity when forming a set of recommendations.

Researchers such as Bridge et al. [2005] and Smyth [2007] discuss case-based systems in more detail and outline extensive summaries of how to build these systems. They highlight the advantages of being able to form more robust similarity metrics to compare a query against features for a defined set of items. In Figure 2.4, the generic system architecture can be seen for a case-based recommendation system. For a user a case with a set of required features is submitted to the system. This case is compared against a case-base of recommendable items. Items are recommended based on the similarity between the case and the case-base using similarity knowledge. Similarity knowledge is often domain knowledge which is representative of the items being recommended or indeed models the weight a given feature should have in the similarity calculation for requested features. A simplified example of this is, if we look at features like genre using the movie example, this may be strongly weighted feature and even if other features match, a user who likes horror movies may not want romantic comedies in their set of returned recommendations.
Case-based recommendation systems have been applied to many domain problems from personal shopping to travel, etc. Burke [1999] utilises case-based techniques to discuss a domain independent case-based system for
personal shopping, the Wasabi system provides for guided customisations without the user having to explicitly note these customisations in their initial query. Ricci et al. [2002] utilise these case-based systems in the travel space to build a recommendation system which can suggest travel options by mediating the user’s query with modelled case-based features such as location, accommodation, activities and attractions.

One core asset of case-based recommendation systems is their ability to utilise complex similarity metrics when comparing a feature based query against a set of cases. Equation 2.1, summarises a weighted sum between the set of different similarity metrics used to compare different features. For instance in Equation 2.2, we can see a sample similarity for a numeric
moviebudget feature. Depending on the domain that the case-based recommendation system is modelling the difference in similarity might be computed as the same if the cases are higher or lower than that requested. Smyth [2007] discusses this and based on domain knowledge of the system the similarity metrics can be tweaked to favour a lower difference for instance, see Figure 2.5. For example a lower moviebudget could be favourable as users may prefer lower budget indy films as opposed to a hollywood blockbuster.

\[
\text{Similarity}(t, c) = \frac{\sum_{i=1..n} w_i \cdot \text{sim}(t_i, c_i)}{\sum_{i=1..n} w_i} \tag{2.1}
\]

\[
\text{sim}_{\text{moviebudget}}(p_t, p_c) = 1 - \frac{|p_t - p_c|}{\max(p_t, p_c)} \tag{2.2}
\]

Another example of similarity metric can be seen in Equation 2.3 where the requirement of a feature actor to be present in a case is very important and
as such if present is scored higher and if not receives no score

$$\text{sim}_\text{actor}(p_t, p_c) = \begin{cases} 
    p_t == p_c, & 1 \text{ if the same} \\
    p_t \neq p_c, & 0 \text{ if not same} 
\end{cases}$$

(2.3)

Case-based systems when forming recommendation often work in a single-shot, retrieval type manner. They work by reacting to a query by presenting recommendations of cases that match highest based on the weighted similarity across features. A downside with this approach to recommendation though is that diversity amongst returned cases is often lacking, for example if a user wanted to be recommended holidays based on features such as sunny, beach, nightlife and this matched highly with holiday cases for Ibiza, the user could be recommended a collection of Ibiza type holidays rather than a selection of options. Incorporating diversity into recommendations is an important adaptation to case-based recommendations and indeed researchers have been developing ways of doing just that [Bradley and Smyth, 2001, Smyth and McClave, 2001].

The next avenue of CB recommendations systems we are going to explore are far removed from the structured, feature rich cases we have discussed here. Recently, researchers of CB recommendation systems have started to explore less structured content where features have to be extracted from the content. In the next section we explore this in more detail when we discuss unstructured text as a form of recommendation knowledge.

### 2.3.2 Unstructured Content

The second approach to CB recommendation systems that we are going to explore is one which utilises unstructured, textual content see Figure 2.6,
Figure 2.6: Example of Unstructured Content where Features must be Elicited from the Text.

this source is utilised when no structured features are present for an item. This approach to building a recommendation system has its roots in textual case-based reasoning (CBR) [Weber et al., 2005] and in information retrieval (IR) research where researchers aimed to transition away from having to have a high domain knowledge to curate the features of items to eliciting them from text through text processing or machine learning techniques. Indeed early research into textual CBR from researchers such as Burke et al. [1997] and Lenz and Burkhard [1997], developed CBR systems which could utilise unstructured content as part of a CBR framework to provide users with suggestions. In both cases they developed question and answering systems built around answering users’ questions from lists of FAQs. Burke
et al. [1997] system at its core utilised sets of question-answer pairs as its case-base and then based on the user query entered it returned appropriate cases that matched the input query.

Much research from IR has spurred on CB recommendation from the representation of the content in documents such as commonly used bag-of-words [Baeza-Yates and Ribeiro-Neto, 1999] approaches to the way in which queries are treated by preprocessing, expansion [Xu and Croft, 1996], etc. Indeed the key difference between CB recommendation systems and IR approaches revolves around the search query. In IR the search query is compared against documents for similarity based on the co-occurrence of terms between the query and the documents, for CB systems the query is now defined as the user’s profile, formed from the terms from items they have liked or contributed upon previously and the returned documents are items recommended to the user.

Figure 2.7 shows a system architecture for a unstructured CB recommendation system utilising an IR type architecture. As content can be inherently noisy, malformed or obtuse, techniques from IR research are adopted to preprocess the terms when forming items. The textual content used to represent items can be anything from large documents to reviews, posts, comments, etc. The item is then indexed for later retrieval when forming recommendations. When a user enters the system the recommendations for that user are formed based on the content from the items they’ve previously liked and these contributions form the user’s profile query. This query is compared against the indexed items and recommendations for that user are computed by measuring the similarity between the users profile query and that of the indexed items.
There are many ways in which we can compute similarity between items when forming recommendations. We could use a simple frequency count as the term weighting function, so that the item could be made up of the frequency counts of the various terms used within the item. Although this is quite simplistic and will produce a bias or noise towards commonly used words. Another approach widely utilised is the TF-IDF weighting metric [Baeza-Yates and Ribeiro-Neto, 1999]. The TF-IDF score of term $t$ in document $d$ is proportional to its frequency of occurrence in that item and inversely proportional to its frequency across the other documents $D$, as shown in Equation 2.4 to 2.6. This results in a higher weighting for terms, which helps to distinguish items during retrieval by discounting matches on common terms present across documents.
Chapter 2. Background

\[ TF - IDF(t, d, D) = tf(t, d) \cdot idf(d, D) \] 

(2.4)

\[ tf(t, d) = \frac{f(t, d)}{\max\{f(w, d) : w \in d\}} \] 

(2.5)

\[ idf(t, D) = \log \frac{|D|}{|\{d \in D : t \in d\}|} \] 

(2.6)

Researchers such as Aciar et al. [2006] have utilised CB approaches and content in the form of movie reviews from IMDb as part of their movie recommendation system, similar approaches are discussed in [Wietsma and Ricci, 2005]. Both of these examples mine review content as an additional source of recommendation knowledge, but they rely on the availability of detailed item reviews, which may run into hundreds of words, which may not always be the case. Recently, researchers have started to examine user generated content (UGC) as a form of recommendation knowledge to describe items. Often UGC is short and succinct, and is only a few lines long as opposed to paragraphs in length. Indeed, with the emergence of the RTW, users participating and contributing on websites is at an all time high. The content produced by these users is often shorter in length but, when mined, can still release much recommendation potential.

Researchers are now focusing on UGC as a form of recommendation knowledge and are working to add structure to the text by classifying the information being produced by those users. Garcia Esparza et al. [2013] have developed a system which classifies tweets from the user’s own timeline. By focusing in on the URLs present within tweets they retrieve extra content to aid in the classification task. They have identified some generic 18 categories in which to group UGC on Twitter. Work by Guy et al. [2011] focuses too on the rewards of modelling users using UGC. Their research is set in a work-based social network and models UGC made up of activities such
as updates to wikis, the liking of items, etc. They note a benefit in the personalised nature of these news articles being presented to the user, by using UGC profiles over entity-based profiles. In a similar vein Phelan et al. [2012] identified using the URLs within tweets to build a RTW news search engine, that could provide highly personalised news content to the user.

There has also been some recent work by Sen et al. [2009] on the role of content tags. Many websites provide users with the capabilities of tagging items such as songs (soundcloud), photos (instagram) or movies (rotten-tomatoes). These user defined tags are commonly known as folksonomies [Jäschke et al., 2007], these provide for rich user defined classifications of items, and by harnessing the collective tagging for an item across a community of users, the strength of that tag can be weighted and used as an interest indicator of its popularity.

Indeed Zhang et al. [2011] outline the benefits of utilising tags for CB systems in their review of the state of the art in tag-based recommendation systems. The three principles they identify are:

1. By allowing users to freely associate tags with items, personal preferences and interests can be elicited.

2. Rich semantic relationships can be formed between the tags an item is given, these could simply be the order in which the tags occur or the frequency of a tag across items.

3. Co-occurrence properties of tags can aid in collectively clustering items into similar topics, genres, interests and such clusters can be harnessed to recommend similar items, friends etc.
Much research is ongoing to fully utilise the potential and benefits [Fi-
ran et al., 2007] of UGC and tag-based CB recommendation systems. Re-
searchers such as Godoy and Amandi [2008] and similarly De Gemmis et al.
[2008] utilise tags to create profiles for users, these tag-based profiles pro-
vide much more succinct and semantically enriched representations of users
when being utilised for CB recommendations. Whilst Sigurbjörnsson and
van Zwol [2008] tackle the problem of tagging an item by exploring the col-
lective intelligence and tagging folksonomy produced across the social graph,
on the picture sharing service Flickr. Zanardi and Capra [2008] showcase
using tags to semantically enrich a user’s profile and the reward of using
these structured tags for efficient content discovery.

In the next section we will briefly discuss the second main strand of rec-
ommendation research when we discuss CF systems and two approaches to
CF, user-based [Resnick et al., 1994] and item-based [Sarwar et al., 2001].

2.4 Collaborative-filtering

CF systems adopt a very different approach to recommendation than CB, of-
ten termed as content-free because there is no need for any item descriptions
instead items are represented as simple ids and item content is replaced with
user ratings. In this section we summarise the two common approaches to
CF (user-based) vs. (item-based) and highlight some important distinctions
between the two system tasks of prediction and recommendation.
2.4.1 Prediction vs. Recommendation

In recommendation systems there are two different but related tasks: prediction and recommendation. The prediction task involves generating a predicted rating for an item $i$ and a user $u$; that is to say a prediction is a rating for a single item. The recommendation task involves generating a set of items to suggest to user $u$; these items are often those with the highest prediction ratings.

Figure 2.8, depicts the tasks of prediction vs. top n item recommendation from a CF perspective. In both cases neighbourhood formation is key. A neighbourhood is a set of similar users or items and we will discuss how we form this neighbourhood in later sections but firstly, for the prediction task we utilise the neighbourhood to form a prediction for one item whereas in top n we return the items which are liked most across the neighbourhood. In the following sections we will discuss CF research from a mainly prediction point of view mainly due to the majority of CF research focusing on algorithms to improve the prediction accuracy of a system.

2.4.2 User-Based

User-based CF works by focusing in on other users who share the same or similar sets of items with our target user [Resnick et al., 1994]. Indeed this type of recommendation approach is similar to exploring the wisdom of the crowd, if a collection of users close to you like an item, the assumption is that you’ll like that item because previously you’ve rated your set of items similarly to your neighbours.

Each user is represented as a set of ratings over a set of items, so that users and ratings are represented by a ratings matrix, as in Figure 2.9. Then to
generate a set of recommendations for some user $Bob$, we first identify a set of similar users who share similar ratings to $Bob$. These similar users are $Bob$’s neighbours and our recommendations are those items that are frequently highly rated by these neighbours. In the case of Figure 2.9, we want to know would $Tron$ be recommended to $Bob$ based on his neighbourhood? If we compute a simple mean square difference similarity (see Equation 2.7) between users in $Bob$’s neighbour and himself (e.g. $sim(Bob, Lisa)$) which is based on the differences between ratings given on co-rated items and the number of co-rated items we can see that $Bob$’s neighbour $Todd$’s ratings are the most similar to $Bobs$, hence $Todd$’s rating for Tron can be used as a prediction of what $Bob$ might rate Tron.
**Chapter 2. Background**

**Figure 2.9:** User-Item Matrix Representation of Movie Ratings.

<table>
<thead>
<tr>
<th></th>
<th>TOY STORY</th>
<th>LION KING</th>
<th>TERMINATOR</th>
<th>TRON</th>
<th>IRONMAN</th>
<th>ELF</th>
<th>THE NOTEBOOK</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOB</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>?</td>
<td>4</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>LISA</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>TODD</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td></td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>JOHN</td>
<td></td>
<td></td>
<td>3</td>
<td>5</td>
<td>5</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>MARY</td>
<td>3</td>
<td></td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

\[
\text{sim}(u_i, u_j) = \sum_{\forall \text{item}_k \in \text{corated}(u_i, u_j)} \frac{(\text{rating}(u_i, \text{item}_k) - \text{rating}(u_j, \text{item}_k))^2}{|\text{corated}(u_i, u_j)|} \tag{2.7}
\]

Figure 2.10, shows user-based similarity and its focus on the rows to compute predictions. Formally, for each target user \(u_i\) we can form a prediction based on \(u_i\)’s neighbours and their collective ratings for an item \(k\), which \(u_i\) has not yet rated. In Figure 2.10, user-based CF is calculated on the rows of ratings and between similar users identified in a neighbourhood.

Two common approaches for computing the similarity between users and their neighbourhood are Cosine [Steinbach et al., 2000] and Pearson [Resnick et al., 1994]; See Equation 2.8 & 2.9 for implementation. These approaches can identify potential skewed rating patterns between users more effectively.
than the mean square difference approach shown earlier. Pearson calculates the extent that corated items are linearly related to each other. If user’s ratings are correlated this produces a 1, if they’ve no correlation 0 and if they’ve a negative correlation -1.

\[
\text{sim}(u_i, u_j) = \frac{\sum_{\forall \text{item}_k \in \text{corated}(u_i, u_j)} (r(u_i, \text{item}_k) - \bar{r}(u_i)) \cdot (r(u_j, \text{item}_k) - \bar{r}(u_j))}{\sqrt{\sum_{\forall \text{item}_k \in \text{corated}(u_i, u_j)} (r(u_i, \text{item}_k) - \bar{r}(u_i))^2} \cdot \sqrt{\sum_{\forall \text{item}_k \in \text{corated}(u_i, u_j)} (r(u_j, \text{item}_k) - \bar{r}(u_j))^2}}
\]

(2.8)

Cosine on the other hand treats users’ ratings as a vector, as such by computing the difference in the angle between two users’ vectors their similarity
can be computed. If there is no difference between the angle i.e. \( \cos (0) \) which equals 1, the users are perfectly similar. Similarly to Pearson if users are wholly dissimilar, this would be produce a similarity of -1.

\[
\text{sim}(u_i, u_j) = \cos(u_i, u_j) = \frac{\sum_{\forall \text{item}_k \in \text{corated}(u_i, u_j)} r(u_i, \text{item}_k) r(u_j, \text{item}_k)}{\sqrt{\sum_{\forall \text{item}_k \in u_i} (r(u_i, \text{item}_k))^2} \sqrt{\sum_{\forall \text{item}_k \in u_j} (r(u_j, \text{item}_k))^2}}
\]

(2.9)

Once the similarity is computed between users this can be utilised when forming predictions. For a user \( u_i \) what is their prediction for \( \text{item}_k \)? Equation 2.10, shows one approach to computing a prediction in user-based CF, here one of the similarity metrics discussed prior is incorporated along with the user’s neighbourhoods ratings to compute a rating for this \( \text{item}_k \). The similarity between \( u_i \) and one of their neighbours essentially weights a rating and this is summed over the neighbourhood.

\[
\text{prediction}(u_i, \text{item}_k) = \bar{r}(u_i) + \frac{\sum_{u_j \in \text{neighbourhood}(u_i)} (r(u_j, \text{item}_k) - \bar{r}(u_j)) \cdot \text{sim}(u_i, u_j)}{\sum_{u_j \in \text{neighbourhood}(u_i)} |\text{sim}(u_i, u_j)|}
\]

(2.10)

In the next section we will discuss our second approach to CF namely item-based. This approach utilises the similarity between items rather than users, in turn we will discuss how to compute similarity and form predictions for users in this approach.
2.4.3 Item-Based

Item-based focuses on the columns in the user-item matrix to compute predictions, Figure 2.10. To perform a prediction for item-based firstly we take a target user $u$ in which a prediction for one of their items, lets call it item $i$. Firstly, all items similar to $i$ are computed for $u$. Then we use those similar items to predict a rating for $i$ based on the ratings for these similar items. Similar to the user-based neighbourhood formation in item-based it now groups similar items. Much research has been carried out to improve upon the prediction accuracy of this approach to CF [Sarwar et al., 2001].

To compute the similarity between items we can modify our Pearson similarity metric from user-based to now deal with similar items as opposed to users. Equation 2.11, computes the item-item similarity between ratings a user has given to their items. As before, if item’s ratings are correlated this produces a 1, if they’ve no correlation 0 and -1 if it’s a negative correlation.

$$
sim(item_i, item_j) = \frac{\sum_{u \in U} (r(u, item_i) - \bar{r}(item_i))(r(u, item_j) - \bar{r}(item_j))}{\sqrt{\sum_{u \in U} (r(u, item_i) - \bar{r}(item_i))^2} \sqrt{\sum_{u \in U} (r(u, item_j) - \bar{r}(item_j))^2}}
$$

(2.11)

Predictions for item-based are computed in much a similar way to user-based although now the focus in on a user’s set of similarly rated items. As such to perform a prediction that a user $u$ will give a candidate item $i$ we compute the weighted sum of ratings $u$ has given to items that are similar to $i$; see Equation 2.12;
Chapter 2. Background

39

\[
prediction(u, \text{item}_i) = \frac{\sum_{item_j \in \text{similar items to \text{item}_i rated by } u} \text{sim}(\text{item}_i, \text{item}_j) \cdot r(u, \text{item}_j)}{\sum_{item_j \in \text{similar items to \text{item}_i rated by } u} |\text{sim}(\text{item}_i, \text{item}_j)|}
\]

(2.12)

In summary, for user-based and item-based, Figure 2.11 shows the flow for each type of CF recommendation system. User-based is focusing on similar users in the neighbourhood and item-based is focusing on similar items to form their predictions. In the next section we will explore the pros and cons associated with both CB and CF systems when forming recommendations.

Figure 2.11: Summary of how the User-based and Item-based Prediction Task Works.
2.5 Pros & Cons

Both CB and CF approaches have their disadvantages and advantages in this sections we are going to highlight these in relation to either design choice and then we will discuss briefly hybrid approaches which aim to curtail the disadvantages of each approach and amplify the advantages.

2.5.1 Content-based

The advantages of CB approaches are:

(i) **Transparency:**
Due to the fact that recommendations within a CB systems are formed by the items you have liked previously. CB recommendation systems can provide the user with explanations as to why an item was recommended to them based on the content-features in the user’s profile that matched the recommended item [Lops et al., 2010].

(ii) **New Item Recommendation:**
Unlike CF systems where an item has to receive a collection of ratings prior to recommendation, a new item that enters a CB system can be recommendable straight away as the content-features associated with it are what the recommendation system utilises when forming a recommendation.

The disadvantages attributed with CB approaches are:

(i) **Overspecialisation:**
CB approaches harness the content a user has in their profile and
recommends items similar to those they liked before. This only provides for items to be recommended that are similar to what a user has seen before. This is often referred to as the serendipity problem [McNee et al., 2006] if an item falls outside of those items you’ve previously liked you will not be recommended it. Smyth and McClave [2001] have looked into adding the likes of diversity to improve the recommendations produced by returning accurate, different and useful recommendations.

(ii) Rich Descriptions Needed:
For CB recommendations to perform well the description of the items being recommended must be well structured and descriptive of the item. Language properties such as polysemy, synonyms, stop-words etc. can create noise and adversely effect the quality of recommendations. Much of these problems though have been catered for in IR research with text processing techniques which aim to remove this noise by stemming, stop-word removal, term expansion etc., as well as more sophisticated techniques such as LDA [Blei et al., 2003].

2.5.2 Collaborative-filtering

The advantages of CF approaches are:

(i) Quality Of Recommendations:
CF approaches require no item knowledge other than the ratings for that item to form recommendations. This allows the system to provide for diversity because it allows the user to discover new items regardless of features of that item such as in a movies case: genre, cast, etc. CF systems can provide highly accurate recommendation once enough
ratings are present for the item or the user has rated enough items to compute similarity.

(ii) **Well Understood:**
CF algorithms are easy to understand and the relationship between the user and the items is clear. New items can be added easily as user rates them and incorporated into their profile. Indeed distributed computing projects such as Apache Mahout\(^6\) have built CF recommendation libraries to allow developers and researchers to integrate their models into a distributed framework capable of handling large quantities of data for use in research and production.

The disadvantages attributed with CF approaches are:

(i) **Cold-start:**
Cold-start or new user/item ramp-up can affect both user-based and item-based approaches where no user ratings or no item ratings are available to compute similarities. Researchers have focused much on trying to curtail the new user/item cold-start problem [Lam et al., 2008, Leung et al., 2007] with many approaches exploring hybrid or CB alternatives when this is the case.

(ii) **Latency:**
Latency in CF systems is the lack of sufficient amounts of ratings. Some influences on latency in CF are refereed to as the grey/black sheep problems [De Gemmis et al., 2009]. Grey sheep refers to a set of users who are constantly fluctuating with their preferences and as such creating a community consensus is impossible, similarly, black sheep are the inverse where all of their rating are so insular it again is impossible to form recommendation for these users.

---
\(^6\)http://mahout.apache.org/
(iii) **Scalability:**

As the number of users and items grow within a system this will influence the time required to perform a set of recommendations. Much of the reasoning behind moving from user-based were forming recommendations via neighbourhood similarity has a large overhead to item-based is because a large pre-processing effort of pair-wise similarities can be computed in advance between items. Indeed sites such as Amazon have made the switch to item-based to try tackle their scalability issues [Linden et al., 2003].

(iv) **Robustness:**

If CF systems are not built with robustness [Mehta et al., 2007] in mind they can fall prey to malicious users who’s aims could be to skew or alter the natural order of recommendations for other users, to promote items they may have a vested interest in. Indeed shilling [Lam and Riedl, 2004] as it is commonly referred to can have a huge impact on prediction performance for items. O’Mahony et al. [2004] present finding that show the prediction shift when attack users are entered into a system. To counteract these attackers more sophisticated neighbour selection algorithms have to be utilised along with exploring hybrid recommendation approaches.

In the next section we will briefly discuss hybrid systems which are built to try diminish the effects of the likes of cold-start and latency on a system. Hybrids can comprise of various different approaches to recommendation which allow them to adapt and change depending on different recommendation scenarios or needs.
2.5.3 Towards Hybrid Recommendation Systems

Hybrids essentially are a combination of various approaches to recommendation, CB working with CF, user-based CF and item-based CF combined etc. The aim of hybrid systems is to account for the potential shortcomings with one approach and supplement these shortcomings with another approach, such as in a cold-start scenario.

Many researchers have [Adomavicius and Tuzhilin, 2005, Burke, 2002, Melville et al., 2002] focused in on developing hybrid systems to increase precision or, indeed, curtail cold-start. Hybrid systems can decide to utilise different weightings or combinations of recommendation information to form better suggestions depending on the use case.

Categorisation for Hybrid recommendation systems has been described in work by [Burke, 2002], these 7 classifications outline the combination characteristics when combining recommendation strategies to form hybrids.

1. Weighted: Different recommendation approaches provide a score and they are combined [Claypool et al., 1999].

2. Switching: Depending on use case the recommendation system will choose one approach [Tran and Cohen, 2000].

3. Mixed: Recommended items from different strategies are presented together [Smyth and Cotter, 2001].

4. Feature Combination: content and collaborative features could be combined to offer recommendations [Basu et al., 1998].

5. Feature Augmentation: One approaches feature act as input to another recommendation approach [Sarwar et al., 1998].
6. Cascade: Collection of recommendation systems work together in a set order and weighted based on significance of their position in the order [Lampropoulos et al., 2012].

7. Meta-level: Recommendation technique produces a model/profile for use with another recommendation system [Schwab et al., 2001].

Researchers such as Bell and Koren [2007] recorded their progress developing one of the most state of the art hybrid systems. This system was developed as part of the Netflix prize competition. Their initial solution produced an RSME of 8.43% and comprised of combining multiple techniques in an ensemble strategy. They noted that when preforming item-based recommendations, forming an appropriate neighbourhood is key to harnessing the intuitive similarities between rated items, but, traditional approaches which utilise Pearson correlation and evaluate items on an individual basis is too naive an approach when the dataset is sparse and the habits of users not uniform. They proposed minimising the use of some neighbours when similar items aren’t present and form prediction rules based on the combination of interpolated weights between an item and its neighbours. Indeed, the final Netflix prize solution by Koren [2009] and their competitors solution Piotte and Chabbert [2009] showcase that it took some 100 plus predictors to get the final result over the line of the 10% marker. The hybrid system showcased approaches such as matrix factorisation [Koren et al., 2009], restricted Boltzmann machines, latent factors, to name but a few, working together to form better recommendations. In the next sections we will transition from how we build recommendations systems to how we evaluate them.
2.6 Evaluating Recommendation Systems

Once the recommendation approach is chosen and the model developed, next it must be evaluated. Generally speaking there are two approaches to evaluating recommendation systems offline and online testing. The first evaluation approach we will discuss now though is online testing. Online testing evaluates the “look and feel” of a system and has a basis in interface design, human computer interaction. Online testing though also accounts for the qualitative examinations of a system, by allowing live users to interact with a system many features can be evaluated at once. Along with the merits of a particular recommendation algorithm users can be presented with different variations of a system such as in A/B testing, they can be asked for feedback on what they liked/disliked etc. One downside with online testings is picking the right split of users who can act as a representative sample of the wider user-base when evaluating the system, such that there are no present biases.

The second evaluation approach is offline testing; this approach is usually utilised to hone the model prior to online testing with live users. With offline testing, algorithms and models can be tweaked and large scale simulations carried out, something which may not be possible with a live system. Offline testing is the standard approach in recommendation research when evaluating a new approach or methodology. Offline testing can also utilise a large selection of evaluation metrics and validation techniques to compute the quantitative merits of an approach. We will examine in the next sections how we evaluate each approach using various metrics such as precision, prediction error and diversity.
2.6.1 Validation

When evaluating a dataset offline, traditionally, one partitions the data into two specific sets, test and training. These sets allow for the testing of recommendation models. They allow for an experiment to test a variety of different adaptations of the model on the dataset. These approaches also facilitate testing on a smaller subset of the dataset which reduces computation time. Below we will outline some commonly adapted approaches to validated recommendation models:

Our first validation approach, random subsampling [Efron and Efron, 1982, Kohavi et al., 1995], performs K splits of the dataset. Each split randomly selects a (fixed) number of examples i.e. \( N = 1000 \). \( K=10 \) would perform 10 K splits on 1000 N examples. This approach allows the experiment to test a smaller portion of the dataset but, by random selecting the examples still will be representative of the model’s performance. The performance of a model can be measured as the average performance across all folds (see Equation 2.13).

\[
P = \frac{1}{K} \sum_{i=1}^{K} P_i
\]  

(2.13)

The second approach to validation is K-fold cross validation [Efron and Efron, 1982, Kohavi et al., 1995], this approach splits up the dataset into K-1 folds made up of training and testing sets. These splits are evaluated such that each example gets evaluated eventually as part of one of the splits. Again this approach measures performance of a model as per Equation 2.13.

The last validation approach which we shall explore is leave-one-out. This is another form of validation similar to K-fold in that all examples will
be evaluated, in this case though the split each time is of size 1 test, N-1 training and it’s run for each example. If we utilise a dataset of 1000, such that N = 1000 is the number of examples. For each experiment each example will be tested in turn against the N-1 training set. This approach measures the performance of a model by evaluating the average over the number of examples N (Equation 2.14).

\[
P = \frac{1}{N} \sum_{i=1}^{N} P_i
\]  

(2.14)

### 2.6.2 Prediction Metrics

When evaluating the recommendation system’s ability at performing predictions we employ prediction metrics. Equation 2.15 and 2.16 represent two offline metrics to calculate the difference in a predicted rating the system believes a user will give an item and the actual rating. These metrics are used when a user’s rating are again split into training and test sets when being simulated. The test set is then used to try predict the rating the user gave in the training set and the difference is used to calculate Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) per user. RMSE penalises those predictions that are highly different to the actual ratings, whilst MAE is more simplistic and averages the difference between the predicted and actual rating across all predictions. Both metrics utilise the difference between the predicted rating the system gives the item and the actual rating a user has given the item.

\[
RMSE(\text{predicted}, \text{actual}) = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left( \text{predicted}_{u,i} - \text{actual}_{u,i} \right)^2}
\]  

(2.15)
Chapter 2. Background

\[ MAE(\text{predicted, actual}) = \sum_{i=1}^{N} \left| \frac{\text{predicted}_{u,i} - \text{actual}_{u,i}}{N} \right| \tag{2.16} \]

### 2.6.3 Recommendation Metrics

When evaluating recommendations systems we also employ metrics to measure the system’s ability at forming quality recommendations. These metrics are again mainly used offline and are utilised on a dataset of users and items etc. to simulate a set of recommendations happening within a system. One metric, precision calculates the system’s ability to recommend quality items and is measured based on the number of relevant items returned to the user in a recommendation list. For example, if a number of observations are removed from a user’s profile, then recommended back to the user these would denote relevant recommendations. N.B Precision can also be calculated in an online scenario with users being presented with new unseen items. If the user has the ability to provide feedback on whether a recommended item was relevant or not, this can be used to measure precision.

\[ \text{Precision} = \frac{|\{\text{relevant items}\} \cap \{\text{recommended items}\}|}{|\{\text{recommended items}\}|} \tag{2.17} \]

\[ \text{Recall} = \frac{|\{\text{relevant items}\} \cap \{\text{recommended items}\}|}{|\{\text{relevant items}\}|} \tag{2.18} \]

Another metric to evaluate the quality of a set of recommendations is recall. Recall measures the number of relevant items over the number of available relevant items that could’ve been recommended. For example if 5/5 relevant recommendations were formed for a user but, the number of relevant items available was 1000, the precision would be perfect but recall would be low. This would indicate that precision alone could be biased. On the other hand
getting perfect recall is trivial by returning all items, all the relevant items will be a subset of this and as such 100% recall can be achieved.

\[
F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.19}
\]

F1 or harmonic mean measure Equation 5.1 is another measure of accuracy for a set of recommendations [Sasaki, 2007]. It combines both precision and recall to produce a score of between 1 (best) and 0 (worst) to show the disparage between either approach individually.

\[
\text{DCG}_p = \text{relevance}_1 + \sum_{i=2}^{p} \frac{\text{relevance}_i}{\log_2(i)} \tag{2.20}
\]

Another important focus for recommendation research is to examine the position in which a recommended result appears. Metrics such as the average position of relevant items in a recommendation list can be used to check if your systems can produce recommends that are accurate and also high up the list a user will be presented with. Other position metrics such as discounted cumulative gain Equation 2.20 factors in the position a relevant recommendation appears and depreciates the score of a set of recommendations based on items appearing further down a recommendation list.

### 2.6.4 Beyond Accuracy

Moving beyond accuracy measured in the form of precision or reducing prediction error, recommendation systems research is moving towards a focus on the relative reward a user gets out of a returned recommendation list. Indeed, research has shown that a level of novelty and diversity within a set of recommendations makes the user perceive those recommendations as
better quality than focusing on variations of a singular precise item. In the following sections we will discuss approaches to incorporate both of these features into a set of recommendations.

2.6.4.1 Novelty

Castells et al. [2011] describes novelty of a piece of information as "how different it is with respect to "what has been previously seen", by a user, or by a community as a whole". Essentially, novelty is a measure of how different an item is from a user’s profile preferences whilst still being accurate. In Equations 2.21 to 2.23 are some approaches to evaluating novelty in recommendations from [Castells et al., 2011, Vargas and Castells, 2011].

\[
\text{Novelty}(R) = - \sum_{i \in R} p(i|R) \log_2 p(i) \quad (2.21)
\]

Equation 2.21 describes the measure of the overall novelty of a set of recommendations. It measures the sum of the probabilities of items being novel given whether they have been observed before by the user. It also uses the inverse of the popularity of the item in the calculation and the resulting novelty score if high represents items from the long tail (unseen items) and if low represents popular (seen) items in the recommendations.

\[
\text{Novelty}(R) = - \sum_{i \in R} p(i|R) \log_2 p(K|i) \quad (2.22)
\]

Equation 2.22 describes an adaptation to computing novelty as seen in Equation 2.21, it this case \( p(K|i) \) is the probability that an item is known by a random user rather than popularity of the item when computing the novelty.
Chapter 2. Background

\[
\text{Novelty}(R|u) = \sum_{n,j \in u} \text{disc}(n)p(\text{rel}|i_n,u)p(j|u)d(i,j) \quad (2.23)
\]

Equation 2.23 focuses on novelty from a different perspective focusing on distance such as Euclidean distance [Suh et al., 2010] between items recommended and the user’s profile. Formally, this approach factors in a discount function at position \(n\), the probability that user \(u\) finds item \(i\) at position \(n\) relevant and additional relevance factors \(p(j|u)\) [Castells et al., 2011].

2.6.4.2 Diversity

Diversity’s role in a recommendation list is the measure of how different each returned item is from each other whilst still being relevant. For example if a system is recommending a user a set of holidays, if all the recommendations are variations of Spanish holidays, whilst they may match perfectly with the criteria of “beach holidays”. They allow for little choice in options if you don’t want to visit Spain. The aim of incorporating diversity is to improve on the quality of recommendations by maintaining the precision and maximising choice. In Equations 2.24 to 2.27, Smyth and McClave [2001] describe approaches to calculate the diversity in a set of recommendations.

\[
\text{Similarity}(t,\text{item}) = \frac{\sum_{i=1..n} w_i \ast \text{sim}(t_i,\text{item}_i)}{\sum_{i=1..n} w_i} \quad (2.24)
\]

\[
\text{Diversity}(\text{item}_1, ... \text{item}_n) = \frac{\sum_{i=1..n} \sum_{j=1..n} (1 - \text{Similarity}(\text{item}_i, \text{item}_j))}{\frac{n}{2} \ast (n - 1)} \quad (2.25)
\]

Smyth and McClave [2001] aim via Equations 2.24 & 2.25 to compute diversity as a function of the similarity between a target query \(t\) and an item. Indeed, diversity in Equation 2.25 can formally be defined as the average
dissimilarity between all pairs of items in a recommendation list. The trade off in these calculations is that if similarity is high, diversity is low. In the follow Equation 2.26 & 2.27 Smyth and McClave [2001] tackle this by measuring quality as the middle ground between similarity and diversity.

\[
\text{RelDiversity}(\text{item}, R) = \begin{cases} 
0, & \text{if } R = \{\} \\
\frac{\sum_{i=1}^{m}(1-\text{Similarity}(\text{item}, r_j))}{m}, & \text{otherwise}
\end{cases} \quad (2.26)
\]

\[
\text{Quality}(t, \text{item}, R) = \text{Similarity}(t, \text{item}) \times \text{RelDiversity}(\text{item}, R) \quad (2.27)
\]

The Quality of a set of recommendations as defined in Equation 2.27 is the similarity between the item and the current target \( t \) and the diversity of the item relevant to the items already selected in the recommendation list \( R \). In the case of Equation 2.27, the diversity metric can be interchanged with that of Equation 2.25 amongst other variations. The addition of diversity to a set of recommendations allows the returned recommendations to be perceived as covering a wider choice of options. In the next section we will depart from the discussion of evaluation metrics for recommendation systems and focus in on the core research area for this thesis namely friend recommendation systems.
2.7 Friend Recommendation Systems

The last avenue which we must explore as a basis for our background research is an overview of work relating to friend finding or friend recommendation systems. Friend recommendation can aid in finding similar users to connect and share information with. Friend recommendation is a type of recommendation that instead of recommending products, users are being recommended. In a CB friend recommendation the content users produce is one example of recommendation knowledge to profile those users. Golder et al. [2009] focus in on the psychological reasoning behind users selecting someone to connect with in a social network context. They identify four structural characteristics that promote connections 1) reciprocity: a mutual interaction between users, 2) shared interests, 3) shared audience: mutual friends and 4) filtering content: if a friend redistributes a user’s content.

There has been considerable research over the past number of years to help users find and connect with people online [Guy et al., 2009, Hsu et al., 2006, Seth and Zhang, 2008]. For example the work of Guy et al. [2009] has looked at the use of recommendation systems to identify people that you might wish to invite into your social network, focusing on an enterprise context. In this work the researchers explored profiling users across a number of different sources of information in enterprise so that explicit relationships could be highlighted based on the fact that two users contributed in similar ways to similar information sources; for example users may share patent authorships or they may be closely related according to the organisational hierarchy within the enterprise, or they may co-author papers together or contribute to the same wikis. This information can then be used to identify similar users and proactively make recommendations to users as a way to drive relationship creation within social media. The results of the live-user
trial confirmed that this type of contextualised friend recommendation was capable of driving a significant uplift in the formation of new relationships within the enterprise.

Related work, at least in terms of its core motivation to drive relationship building, has been carried out by Freyne et al. [2009] and Geyer et al. [2008] who have explored a number of recommendation techniques for improving user engagement within social media and social networks. Most researchers focus on identifying many of these traits algorithmically as opposed to formally, with work such as Silva et al. [2010] using practices borrowed from graph analysis and graph theory to suggest friends that are within two degrees of separation to a user within their social graph. Kim and Shim [2011] uses both the content and graph analysis type approaches to build a probabilistic model for CF to form their friend recommendations. Also, it’s worth noting that there is much research surrounding friend recommendation in the social context of Twitter or similarly large scale social networks such as Weibo⁷, MySpace⁸ [Armentano et al., 2011, Chen et al., 2012, Moricz et al., 2010]. The 2012 KDD prize winners [Chen et al., 2012] used Weibo as their testbed and matrix factorisation and feature detection to build their friend recommendation models.

Chen et al. [2012] utilised a hybrid combination of the latest CF research into matrix factorisation and additive forests to identify latent recommendation potential in the dataset and thus increase precision. Other researchers such as Chechev and Georgiev [2012] focus back on the content being produced by users on these social networks to recommend friends, their approach again extracts features, but this time they are extracted from the text. Features such as links and hashtags (keyword/s pertaining to the topic of

⁷www.weibo.com
⁸www.myspace.com
a message) are used to build richer user models, they then utilise text-based similarity techniques to recommend similar users using these features. We will adopt similar approaches to friend recommendation in this work, focusing primarily on CB recommendation techniques.

There has also been other research into identifying the connectivity or “friendship” amongst people but in a more implicit information driven way. Work by Quercia and Capra [2009] monitors the unique mac addresses associated with bluetooth devices in a location to build a social weighted graph-based on the frequency of interactions and durations. In this manner friends could be recommended based on their co-location and interactions with each other. Other work which harnesses a graph-based metric is that of Roth et al. [2010]. They introduce the implicit user graph for suggesting friends when writing emails. Their products “Don’t forget Bob” and “Got the wrong Bob” integrate with Google mail, when a user is composing a message he/she can be recommended a correspondent from that user’s email contacts. Using a weighted graph users can be ranked based on the group of contacts who are contained upon an email correspondence together, the frequency of appearance on emails of those other users who are contacted, and the recency of email contact. This weighted graph allows for a user’s preferences to be learned and implicit recommendations for recipients formed when an email is being composed.

2.8 Summary

In this chapter we have shown an in-depth overview of relevant research as a foundation for the work we shall present in this thesis. We have described CF and CB recommendation systems and also their pros and cons and some
recent transitions into hybrid approaches. In this thesis, we will focus mainly on using CB approaches to friend recommendation. In the next chapter we will introduce our recommendation framework and discuss how we plan to evaluate its ability to recommend friends.
3

Tweet-based Profiling & Recommendation

3.1 Introduction

In this chapter we consider Twitter from a user modelling and recommendation viewpoint [Hannon et al., 2010, Pennacchiotti and Popescu, 2011]. We are motivated by Twitter’s potential as a powerful source of profiling data and recommendation knowledge [Garcia Esparza et al., 2010, Guy et al., 2011, Phelan et al., 2011]. This is a novel take on profiling and recommendation in itself. For example, up until now, most profiling and recommendation approaches have assumed the availability of high-quality interest and preference information, such as user ratings, purchase histories, or other forms of transaction logs. The RTW and Twitter, in particular, seems far more limiting from a profiling and recommendation viewpoint. After all, user tweets are limited to only 140 characters in length and users tweet on a wide variety of topics, often in a shorthand that is likely to be opaque to
conventional natural language processing techniques [Sriram et al., 2010]. Nevertheless, the sheer volume of real-time data that is available on Twitter makes for a tantalising profiling proposition and it is in this context that we seek to explore what might be feasible in practice.

In this chapter we will begin to examine the research contributions outlined in Section 1.3. We will focus in on RC 1 & 2 which question the ability to build an effective friend recommendation system powered by tweet-based profiles and the ability to leverage the user’s social graph when profiling. To answer these questions we, firstly, will develop a friend recommendation system that can suggest interesting and relevant friends to users, then we will carry out extensive evaluations to gauge the system’s quality at recommending friends. In this work we will focus on the terms within the tweets themselves and aim to identify a user’s interests based on these terms. This approach will hopefully allow for personalised friend recommendations for each user based on their tweet contributions.

The act of following users can often be quite subjective, what a system may decide as a perfect recommendation based on a certain set of overlapping characteristics or metrics, may not always suit the desired intent a user may have. In the following sections we will introduce our profile and recommendation techniques, which aims to model a user’s interests and form recommendations of like minded individuals who share similar interests. We have developed a testing framework, that provides Twitter friend recommendations, based in part on a user’s contributions to the network in the form of their tweets, which allows for the system to be utilised in offline and online testing scenarios.
3.2 Profiling User Interests

Traditional recommendation systems profile creation utilises a set of concrete interests or likes core to a user. With these interests a user can be modelled and similarities drawn between our user and the dataset (Section 2.2). In this work we propose using terms extracted from the content of tweets, under the basic assumption that what we talk about on Twitter is a good reflection of what we are likely to be interested in. As such, in the following sections we will start to build our recommendation system powered by what users tweet about (RC 1).

3.2.1 From Tweets to Profiles

Tweet content is varied and noisy and there are a number of important caveats to consider when utilising it. People tweet on a wide variety of topics and the language used in tweets is often truncated to meet Twitter’s strict 140-character limit. An example of an unprocessed tweet can be seen in Figure 3.1, here we can see key terms such as iOS, web and HTML used within the tweet; these terms can be used to indicate potential interests for this user. It is not safe though to assume that a user’s own tweets are a fair reflection of all of their interests. Many users have interests in topics that they themselves rarely tweet about, and generally they will follow other users in order to satisfy these interests. Many of those tweets that may not have been originally composed by the user, but instead by the users they follow can appear as retweets associate with the user. These retweets represent a user sharing some sentiment with the composer of the original tweet and as such are treated as if the tweet were their own.
When profiling a target user $U$ to reduce potential noise we can given a set of tweets ($\{t_1, ..., t_k\}$) use term-based weighting techniques such as TF-IDF [Baeza-Yates and Ribeiro-Neto, 1999] to build a weighted index of tweets. In effect, we treat the set of tweets for a given user as a document containing the terms of these tweets. We do this for all users within the dataset.

$$tweets(U_T) = \{t_1, ..., t_k\} \quad (3.1)$$

![Figure 3.1: User Tweet from Twitter Profile of @johnhenryhannon.](image)

When profiling users on their tweets we firstly consider the simplest source of profiling information, the user’s own recent tweets. Thus, as per Equation 3.1, for a target user, $U_T$, let $tweets(U_T)$ be the set of recent tweets for $U_T$; in this work we will assume that $tweets(U_T)$ is the user’s 100 most recent tweets. In this way $tweets(U_T)$ provides the basis for a CB approach to user profiling, obviously under the assumption that users are likely to tweet about things that interest them. The number of tweets chosen was a product of two conclusions, firstly we did not what to focus solely on the last status update from a user, as this would represent a shallow representation of that user’s profile. Instead by giving a threshold of 100 tweets this allowed for the user to comment/interact with a wider variety of interest topics and thus build a richer profile for themselves. The secondary reasoning behind this threshold is a limitation with Twitter’s API. Users are limited to the number of API calls per hour they can make and thus getting every status for each individual in the dataset would quickly become unfeasible. With this
approach to modelling users now identified we can evaluate its potential at effectively recommending friends, as per RC 1. Before we assess that though we must first explore if there are any other avenues of recommendation knowledge that will provide insights into the interests of a user? To answer this in the next section we will examine the user’s social graph to see can any profiling techniques be formed based on the way in which users connect to each other.

### 3.2.2 Harnessing The Social Graph

One issue we could face by only modelling a user’s own tweets is that we are only getting a representation of the set of tweets they wish to share on their stream. We have no indication of the interests they have consumed from their friends other than via user retweets. This potentially masks the user’s whole set of interests. By its very nature Twitter is social and users seek out others to satisfy an interest need and in this section we hope to address whether a user’s tweets are a good representation of them by looking at the user’s friends and followers as a profiling source.

Each Twitter user follows a set of other users, their friends, and each user is followed by a set of users called their followers; see Equations 3.2 and 3.3. If we examine Figure 3.2 each of these social groupings are producing content representing their sets of interests. By exploring $U_T$’s friends and followers we can identify two new sources of recommendation information by harnessing $U_T$’s social graph. We can reasonably assume that the tweets of their friends and followers may provide further insights into a user’s interests. Users actively select their friends, probably because they expect their tweets will be of interest, and thus we can use their tweets in much the same way as a user’s own tweets, as a complementary source of profile
information. Thus, $\text{friendtweets}(U_T)$ is the set of tweets of the friends of $U_T$ (see Equation 3.4).

$$\text{friends}(U_T) = \{f_1, \ldots, f_m\}$$

(3.2)

$$\text{followers}(U_T) = \{g_1, \ldots, g_n\}$$

(3.3)

$$\text{friendtweets}(U_T) = \bigcup_{\forall f \in \text{friend}(U_T)} (\text{tweets}(f_i))$$

(3.4)

$$\text{followertweets}(U_T) = \bigcup_{\forall g \in \text{followers}(U_T)} (\text{tweets}(g_i))$$

(3.5)

In turn, the followers of $U_T$ make the active decision to follow $U_T$, presumably because they ($U_T$’s followers) expect $U_T$’s tweets to be of interest. But
will the tweets of these followers be of interest to $U_T$? And can they be used as a viable source of profiling information as per Equation 3.5? Neither questions can be affirmed with confidence yet, since, in the case of the majority of Twitter users at least, users exert little control over their followers; users rarely prune away followers that don’t interest them and there are many cases of followers who generate very few tweets themselves. Nevertheless, the tweets of followers certainly provides an intriguing source of profile information worth exploring. In the next section we will explore how we will use these identified profile sources to aid in forming our recommendation strategies and in evaluating their merits (RC 1 & 2).

### 3.3 Recommending Social Connections

Now that we have a basis for profiling Twitter users, based on tweets, we can index these profiles and develop the recommendation framework to deliver results based on a target user profile. We have chosen to develop this framework using the open source Lucene platform\(^1\). There are numerous advantages to proceeding in this fashion, as opposed to developing a bespoke recommendation framework. For a start, Lucene provides a proven, robust, and scalable indexing and retrieval platform that is designed to cope with Web-scale data and usage. In addition, it provides access to powerful indexing and term-weighting features that will accommodate a more sophisticated approach to user profiling than a simple frequency-based term-weighting scheme. Finally, Lucene’s retrieval functions can be used directly for the query-based retrieval of profiles and can be readily adapted for recommendation. Lucene will handle the retrieval, indexing and index storage as set out in our system architecture in Figure 3.3.

\(^1\)http://lucene.apache.org/java/docs/
Since Lucene is a text-based search engine, its basic units of information are documents to be indexed and stored for retrieval. We can treat profiles as documents, which, after all, are simply collections of words (from tweets). Thus, using Lucene’s indexing features we can represent each, $U_T$, as a weighted term-vector, $profile(U_T, source)$ (see Equation 3.6), such that the $i^{th}$ element of this vector represents the $i^{th}$ unique term in $source$, and the weight of this $i^{th}$ term ($w_i$) represents the importance of this term for $U_T$. In the case where $source$ is one of the content sources ($tweets(U_T), friendtweets(U_T)$ or $followertweets(U_T)$) then these terms will be the words used in the tweets of the relevant users. In what follows we will use $profile(U_T)$ instead of $profile(U_T, source)$ without loss of generality in cases where the $source$ parameter is clear.
Chapter 3. *Tweet-based Profiling & Recommendation*

\[ \text{profile}(U_T, \text{source}) = \{w_1, \ldots, w_n\} \quad (3.6) \]

We use Lucene’s TF-IDF as our weighting metric. The TF-IDF score of term \( t_i \) in \( U_T \) is proportional to its frequency of occurrence in \( \text{profile}(U_T) \) and inversely proportional to its frequency across the other profiles, \( U \). This results in a higher weighting for profile terms that are frequent in a given profile but infrequent across the profile-base as a whole, which helps to distinguish profiles during retrieval by discounting matches on common terms. For example, if we represent each user just by their own tweets, then the TF-IDF weighting will give a higher weight to terms that are common to \( U_T \) but unusual across the rest of the user population. These high-scoring terms serve to better distinguish \( U_T \)’s interests relative to the other users.

Query-based retrieval and profile-based recommendation are then implemented using Lucene’s standard retrieval function, with the target user’s *profile document* serving as the search query in the case of the latter (see query \( Q \) in Figure 3.3). All of this provides a very powerful and flexible retrieval and recommendation framework, since profiles can be represented and indexed by a combination of source terms, effectively harnessing a variety of different recommendation strategies. For instance by using the content sources we can generate a space of CB recommendation systems.

As way of a summary, recommendations for user \( U_T \) would be carried out as follows: firstly the system will profile \( U_T \)’s tweets. These terms form that user’s profile query \( Q \). \( U_T \)’s query is sent to the recommendation engine which then forwards the query \( Q \) to the retrieval engine to be queried against all indexed users (documents). A ranked list of recommendations are returned from the retrieval engine based on the similarity to the query and in turn these are recommended to the user (Figure 3.3).
3.4 Evaluation

The success of our recommendation system will ultimately depend on its ability to suggest new users who are likely to be worth following, by the target user; remember, we are interested in recommending friends as opposed to followers. In this section we describe our evaluations of a variety of different recommendation techniques based on a comprehensive dataset generated from real Twitter users. Firstly, we wish to carry out an evaluation to test the merits of using tweets as our profiling source, as per RC 1. Secondly, we will evaluate the socially driven profiling techniques described in Section 3.2.2 and test their recommendation effectiveness. To this end we will carry out an initial offline evaluation. This particular approach to evaluation is commonplace amongst recommendation systems research and allows us to compare the recommendation effectiveness of a variety of different profiling and recommendation strategies in an offline manner.

This evaluation will act as our justification for later evaluations which will tackle our remaining research contributions. It is worthwhile reiterating the main desired outcome of this evaluation, namely, can we provide effective and accurate recommendations using only real-time Twitter tweet term-based profiles via our recommendation framework? In the next sections we will explore this in more detail by discussing the evaluation dataset and our recommendation methodology.

3.4.1 Dataset

For the purpose of this evaluation we needed access to a critical mass of existing users. To do this we imported 20,000 users directly using the Twitter
Chapter 3. *Tweet-based Profiling & Recommendation*

API. Twitter’s API gave us an access point to collect user generated information in the form of users’ tweets, user profiles and social connections.

We began with a small seed-set of 15 users (basically the friends and colleagues within our research group) and expanded the user-base by following their followers and friends links, see Figure 3.4. This dataset was collected over a 3 month period at the start of 2010. For each user we also downloaded up to 100 recent tweets. We split the dataset into two sets of users – one containing 1000 users to act as test users, and a larger training-set of 19,000 users; see Table 1 for a summary of these datasets in terms of their median tweets, friends, followers and the median words per tweet per user. These numbers are quite indicative of other Twitter datasets, although the numbers could be seen as relatively low and this can be down to the nature of the dataset collection. The dataset was seeded via Irish Twitter accounts.
Table 3.1: Evaluation Datasets with Median number of Tweets, Words pre tweet, Followers, and Friends.

<table>
<thead>
<tr>
<th>Users</th>
<th>Tweets</th>
<th>Words</th>
<th>#Followers</th>
<th>#Friends</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>80</td>
<td>15</td>
<td>664</td>
<td>321</td>
</tr>
<tr>
<td>19,000</td>
<td>78</td>
<td>14</td>
<td>465</td>
<td>520</td>
</tr>
</tbody>
</table>

accessed in 2010, these accounts could be seen as early adopters and less active than their 2013 counterparts.

### 3.4.2 Methodology

During this initial evaluation we evaluate 5 different profiling and recommendation strategies based on the different sources of profile information, in isolation, and in combination, on users in the 1000-user test-set. For each of the profiling strategies we constructed a separate Lucene index that contained the appropriate index information and then used the standard Lucene retrieval engine to generate recommendations. We implemented 4 content-based strategies that rely on the content of tweets namely:

- **S1** users are represented by their own tweets ($tweets(U_T)$);
- **S2** users are represented by the tweets of their friends ($friendstweets(U_T)$);
- **S3** users are represented by the tweets of their followers ($followerstweets(U_T)$);
- **S4** a hybrid strategy in which users are represented by the combination of tweets from $tweets(U_T)$, $friendstweets(U_T)$, and $followerstweet(U_T)$;

For each of these target users we generate a query by taking the top-20 highest frequency terms from the user’s profile eliminating stop-words; in
other words, we just use a portion of the users’ full profile information. The reasoning behind selecting the top-20 terms was to provide for quick search over a large pool of potential recommendations (Lucene index) and to highlight the topics that the user uses most frequently. This profile query is used to generate a recommendation list for each of the 4 CB strategies.

Finally we implemented one hybrid, ensemble strategy, S5. This strategy is an ensemble composed of the previous basic component recommendation strategies, S1 – S4 plus we include a collaborative approach which utilises the numeric IDS of the user’s friends and followers. We then utilise the union of these recommendations from these independent strategies and they are scored and ranked. The addition of this approach will act as a baseline, although evaluating collaborative approaches is outside the scope of this thesis as they represent more of a social graph approach and act only to fill in ones social graph, rather than extend it by finding users based on shared interests, which is our goal.

**S5** the scoring function is based on the position of the user in each of the recommendation lists so that users that are frequently present in high positions are preferred over users that are recommended less frequently or in lower positions.

For each target profile we count how many of the recommendations are in the user’s known friends list. We call these relevant recommendations; in other words, we look to see how often the recommendation system suggests people that the target user is known to have followed, as each user profile is structured as a Lucene document we utilise a standard document based precision metric as our quality metric (Equation 2.17). A relevant document is a recommendation, that is, a friend of our test user, and the retrieved documents is the set of recommendations returned from the user query. In the
sections that follow we will describe the summary results across these different algorithms, focusing on the average overlap between recommendation lists and friends lists, and the position of these relevant recommendations. And we do this for different recommendation list sizes \((k)\) from the top-5 recommendations to the top-20 recommendations. We capped our recommendation list size at 20 due to the nature of following someone on Twitter. Unlike many other social networking sites, users plan who they are going to follow more stringently because it impacts on the stream of information that they have to consume [Krishnamurthy et al., 2008]. Following a large set of users in one sitting is generally not typical behaviour for Twitter users.

### 3.4.3 Detailed Analysis

Our basic measure of recommendation performance is the average percentage overlap between a given recommendation list and the target user’s actual friends list; this is effectively a precision measure. To begin with, we will evaluate our 5 strategies, 4 content and 1 ensemble using this metric as our measure of recommendation quality. In Figure 3.5 we have graphed the average precision versus recommendation list size for the 5 different recommendation strategies using the 1000-user test-sets. Each test user will be presented with \((k)\) recommendations and precision is then calculated per test user. Where \((k)\) will range from 5-20 potential friend recommendations.

Overall, the different recommendation strategies appear to perform well across the different recommendation list sizes, generating precision scores, in the 1000-user test-set, of between 11% (for strategy S2 at \(k = 20\)), strategy S2 being user profiles made of tweets of that user’s friends. This should be viewed as a very positive result since the success metric here — namely, that the target user is a friend of a recommended user — can be viewed as
Chapter 3. *Tweet-based Profiling & Recommendation*

**Figure 3.5:** Average Precision vs. Recommendation List Size for the 5 Different Recommendation Strategies using the 1000-user Test Set.

setting a reasonably high relevance standard; in Twitter, becoming a friend of a user is a deliberative act and most users limit who they follow to avoid being swamped with irrelevant messages. We can also see that relevant recommendations tend to be clustered towards the top of recommendation lists since the precision of all strategies is seen to decline within increasing recommendation list size. Both of these conclusions validate our initial RC 1 & 2 by justifying tweets as a profiling source and the benefit of utilising a user’s social graph for extra recommendation knowledge.

Interestingly, we find that strategy $S_2$ tends to perform poorly compared to $S_3$. For example, for recommendation lists of size 10, strategy $S_2$ delivers a precision score of only 15%; so only 1 or 2 of these 10 recommendations are actually friends of the target user. In contrast, we find the tweets of a user’s followers (that is the people who follow the target user), $S_3$ to perform significantly better. For instance, for recommendation lists of size 10, strategy
S3 delivers a precision score of 20%; a 33% relative increase in precision over S2. This suggests that the tweets of your friends are not necessarily a good predictor of these same friends. Of course the percentage of overlapping recommendations is just one measure for evaluating recommendation performance; another key metric is the position of a recommendation. As part of our initial evaluation we aimed to see how high are relevant recommendations appearing in returned lists.

3.4.3.1 Ranking Effectiveness

The position of relevant recommendations is an important consideration, especially since we know that users focus the lion’s share of their attention on items at the top of result or recommendation lists [Silverstein et al., 1999]. Thus, two strategies may perform well in terms of their overall precision, but if one consistently produces relevant recommendations in the top-half of the list, while the relevant recommendations for the other tend to appear in the bottom-half of the list, then, all other things being equal, the former strategy can be considered to be superior.

In Figure 3.6 we plot the average position of the relevant recommendations versus recommendation list size for the 5 recommendation strategies. Generally the strategies perform similarly across varying recommendation list sizes – the average position of relevant recommendations ranges from approximately 1.7 (when \( k = 5 \)) to just over 7 (when \( k = 20 \)) – and it should be clear that all of the strategies are capable of positioning relevant recommendations towards the top-end of their recommendation lists.

Interestingly, the CB strategies are able to produce recommendations closer to the top of a recommendation list than that of the hybrid ensemble strategy S5. This outcome is encouraging for our CB friend recommendation
Chapter 3. *Tweet-based Profiling & Recommendation*

**Figure 3.6:** Average Relevant Position vs. Recommendation List Size for the 5 Different Strategies using the 1000-user Test Sets.

 system as users generally will focus mostly on this section of a list, it bodes well for a live system.

### 3.4.4 Beyond Precision and Coverage

It is worth returning to the manner in which we evaluate precision in the initial Twitter offline experiments. Precision is calculated as the percentage overlap between recommendations and the target user’s existing friends list, but it would be unwise to discount the non-overlapping recommendations as definitively not relevant to the target user. They are ‘not relevant’ only in the sense that they are not already friends of the target user, and it would be incorrect to assume that these recommendations are not of interest to the target user. They may indeed be of great interest to the user. As
such we view these results as providing a useful baseline with respect to likely recommendation precision in a live-user context. In this section we describe just such a trial, which we coupled with our initial evaluation. It’s based on the usage patterns of 34 trial participants during March 2010. These participants were all existing Twitter users. Summary information is presented in Table 3.2.

### 3.4.4.1 User Recommendations

As part of our live user evaluation we developed a web application called Twittomender, see Figure 3.8. In this figure we present the user interface and show results from utilising the search mode of Twittomender which we shall discuss in Section 3.4.4.2. This interface remains the same when run in recommendation mode with the notable exception that the query is a user’s profile as opposed to search terms. The Twittomender System was made available online for the purpose of this trial and we invited interested users to synchronise their Twitter accounts with Twittomender so that we could include their tweets and social graphs in the Twittomender database. These users were gathered by putting out an open call tweet on a personal Twitter account, then this tweet disseminated across Twitter’s social network by potential trial users seeing that tweet or a “Retweet”, mention of the trial [Suh et al., 2010]. For the purpose of this trial, Twittomender was configured to

<table>
<thead>
<tr>
<th></th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number Of Participants</td>
<td>34</td>
</tr>
<tr>
<td>Median Number Of Friends</td>
<td>66</td>
</tr>
<tr>
<td>Median Number Of Followers</td>
<td>71</td>
</tr>
<tr>
<td>Median Number Of Tweets</td>
<td>273</td>
</tr>
</tbody>
</table>
use profiling and recommendation strategy $S_5$, meaning that a combination of all of the different sources of profiling information was brought to bear on recommendation. The goal of this experiment was to evaluate the difference between the offline evaluation precision scores and that of the online, and to also question the conservative nature of our relevant recommendation criteria. Upon synchronising their Twitter account with Twittomender (so that their profiles could be generated from their tweets, friends, and followers) each participant was presented with a list of 30 recommended Twitter users (from a Twittomender database of approximately 100,000 users at the time of this trial) and the user was asked to indicate which of the recommended users they would likely follow. There are two important points to make here: (1) none of the users existing friends or followers were included in these recommendation lists – they were filtered out pre-recommendation; (2) participants understood that for the purpose of this trial by indicating that they would likely follow a given user, it would not in any way affect their live-Twitter social graph (in other words they wouldn’t actually end up following the users in question).

On average, the 34 participants indicated a willingness to follow an average of 6.9 users per recommendation list and the majority of these relevant recommendations appeared towards the top of the recommendation lists. For example, Figure 3.7 shows a histogram of the number of relevant recommendations for different recommendation list position ranges and we can see that 120 out of a total of 236 relevant recommendations appeared in the top 10 recommended users. We view this result to be very positive. Every participant found at least some new users worth following in their recommendation list and an average of almost 7 new users to follow per recommendation list would certainly help to drive ‘friend relationships’ in the Twitter universe.
3.4.4.2 User Search

During this trial we also provided users with an opportunity to test the query-based search functionality of Twittomender. Very simply, users were encouraged to enter a standard query with a view to receiving recommendations for users who are likely to be relevant to this query. This search functionality allowed the user to search out serendipitous recommendations based on inputed queries. Out of the 34 participants, 31 tested the search service. They submitted queries with an average of 3.7 terms on a variety of topics and once again received recommendation lists containing 30 users; for convenience we have summarised their query terms in the form of a Wordle tag cloud (wordle.com) in Figure 3.9. The relative weight of the terms in Figure 3.9 represents the overlapping of terms that users entered during their user searches.
Chapter 3. Tweet-based Profiling & Recommendation

Figure 3.8: An Example of the User Trial Interface using Search Mode for the Query “social search”.

We tracked how often a user received a search result representing a user they would be inclined to follow; once again we filtered-out any users that were existing friends from these result lists. This time users indicated they would be willing to follow an average of 4.9 of the suggested users, per search. Obviously this indicates a lower number of relevant users, compared to the 6.9 relevant users per recommendation list for the profile-based recommendations, but this is probably to be expected since the short search queries provided by users carry far less (albeit perhaps more focused) information than the richer profile queries used during recommendation. Nevertheless, we view an average of 5 new users worth following per search to be again a
very strong result from a Twitter perspective.

In Figure 3.10 we show a histogram of the number of relevant results for different recommendation list position ranges. We see a relatively even spread of relevant users across the top 30 results, with a bias towards the top ranking results. We can see Twittomender performs well as an effective search tool. The results that were followed are positioning high in the returned search lists. This manual search functionality that allows users to define the terms that interest them and to find people who share and talk about these interests, has another benefit, we believe. It has the potential to be a solution to users who are new to Twitter and do not have the knowledge gained with experience to find good users to follow. As way of a summary in the next section we highlight the core lessons learned from all of our evaluations.
3.4.5 Lessons Learned

Based on our offline evaluations, which utilised only Twitter tweet data, there is room to be optimistic about the potential for a friend recommendation system for Twitter using the techniques described here. From this evaluation we learned some lessons to justify our choice of research contributions. As way of a summary, below are some of our insights:

1. In each case, we have found our CB profiling approaches to be capable of delivering recommendation lists that include a reasonably high percentage of relevant users appearing towards the top of these lists. We did find that the various strategies are influenced by the size of the recommendation list, but our approaches did showcase tweets as an effective source for profiling users (RC 1). As way of a summary
Figure 3.11: Summary of (a) Average Precision and (b) Position Results across the 4 Content (S1-S4) and 1 Hybrid (S5) Strategies.

Figure 3.11 (a & b) present the mean average precision and the mean average position results across recommendation list sizes 5-20.

2. We also learned that by including the content from the user’s social graph connections, namely the friends and followers of the user, we could form quality recommendations that can match or indeed outperform those of the user alone. This case is most evident when we
compare the average positions of recommendations between the user based and the social graph based approaches (RC 2).

3. Although the offline study didn’t facilitate an evaluation of novel recommendations, our live user tests discussed in Section 3.4.4 suggest that a significant percentage of recommendations which were not yet followed by the target $U_T$ were worthy of following.

Based on these findings the merits in further developing these profiling and recommendation techniques are self evident. In the next chapter we will discuss further development of our CB approaches and build upon the results achieved for RC 1 & 2 by exploring other sources of profiling information.

### 3.5 Conclusions

In conclusion, we have introduced our approaches to profiling and indexing users using our friend recommendation framework. We have suggested 3 basic profiling strategies per data source, which utilise tweets and the user’s social connections: (1) representing users by their own tweets ($tweets(U_T)$); (2) by the tweets of their friends ($friendtweets(U_T)$); (3), by the tweets of their followers ($followertweets(U_T)$). We have tackled two of our initial research contributions (RC 1 & 2), namely that indeed tweets are a good signal of profiling information for a user and that the user’s social graph has latent recommendation potential. These profiles can then be used to build an effective friend recommendation system which can identify 3 prior friends using our precision metric at $k = 20$. Also, these recommendations appear at the top end of the recommendation list which bodes well as this is where users spend the lion’s share of their time.
In the next chapter we will further develop our framework by expanding to a larger dataset. This dataset will comprise of Twitter tweets as before but, our focus will shift to explore different sources of CB profiling information with the view of evaluating our RC 3.
From Tweets to Tags

4.1 Introduction

Given the importance of social connections in public and private social networking services it is not surprising that most networks have explored ways to help users find new people to connect with. For example, Twitter\textsuperscript{1} and Facebook\textsuperscript{2} allow users to find new friends via simple search functions. However search-based approaches can have their limitations and add some friction by relying on explicit user activity. Twitter and Facebook also both support the creation of communities of users that share common interests (\textit{groups}\textsuperscript{3} on Facebook and \textit{lists}\textsuperscript{4} on Twitter) as a way for users to engage with topics of interest. For example, Twitter introduced user lists in late 2009, allowing users to be grouped according to user-defined topics or themes. Other users can subscribe to this list to benefit from members’

\begin{footnotesize}
\begin{enumerate}
\item https://twitter.com/search-advanced
\item https://www.facebook.com/find-friends/browser
\item https://www.facebook.com/about/groups
\item http://support.twitter.com/articles/76460-using-twitter-lists
\end{enumerate}
\end{footnotesize}
tweets. Lists on Twitter have become an important way for users to curate content on topics or themes that matter to them and they have been widely adopted by media outlets, news agencies, and marketing departments as a way to better organise content streams and connect with communities [Greene et al., 2012, Nasirifard and Hayes, 2011].

In this chapter we are interested in exploring the ways in which we form recommendations for a given target user by evaluating different profiling content sources. To this end we wish to evaluate other sources of profiling information which complement the tweet source we have already identified in Chapter 3. Raw tweet content can be inherently noisy because it’s constrained by length or is written in non-conventional speech and so might be expected to serve as a less reliable signal of user interests. In this respect, and in line with RC 3, we have identified two curated, whilst still freeform, sources of tag-based profiling information. Our first new profiling source is spurred on by the communities of like minded users grouped into Twitter Lists. These lists are usually curated collections of users on a particular topic. For example, a user might create a list and include all of the users they deem to be relevant; by following this list another user will see the tweets of these members in their own stream. Our source of list data and, moreover, tag data for these lists comes from Listorious\(^5\). Listorious provides a searchable index of categorised Twitter lists. But instead of categorising the individual users within the lists, Listorious users tag the list with its category descriptors. This second source of tags is far more straightforward, web services such as WeFollow\(^6\) allow their users to assign interests tags to their profile to indicate their sets of interest. By comparison, user-tag information, such as that used by WeFollow, is directly assigned with a user and so should provide a stronger interest signal than that of list-tags

\(^5\)http://listorious.com
\(^6\)http://www.wefollow.com
from Listorious. Although, this may turn out to be not the case, and we shall see later on in the evaluations section.

Before, we continue to describe how we utilise these new sources it’s interesting to explore the assumptions these tags and tweets make in relation to the user and how they’re associated with them. For instance, an interesting profiling dimension relates to whether the interest information is directly or indirectly linked to a user. For example, using their tweet content or the WeFollow tags of a user is an example of direct interest information; the information is associated with the particular user. But we can also avail of indirect information. For example, we can harness the tweet content and tags of a target user’s followers or friends. These interests are indirectly associated with the target user, but are likely to include relevant interest indicators for the target user because their friends and followers will inevitably share interests.

We can now represent a design space of profiling/recommendation strategies by categorising our different sources of information along two distinct categories: noisy vs curated and direct vs indirect. This is presented in Figure 4.1 with different approaches organised according to these dimensions. For example, mining a user’s own tweet content is an example of a noisy, direct approach, whereas mining the tweets of their friends or followers is also noisy but more indirect; we view followers to be less direct than friends because in the case of the latter the user must choose a friend, whereas they have less immediate control over their followers. Similarly, mining a user’s WeFollow tags is curated and direct whereas mining tags from the Listorious lists of a follower is curated but extremely indirect.
In the following sections we shall outline how we plan to use these new profiling sources and evaluate their efficiency at recommending social connections. By building upon the offline framework introduced in Chapter 3 we will evaluate RC 3, which asks can we transition beyond tweets as a profiling source of our users? We will revisit RCs 1 & 2 also in this chapter as they aid us in forming our recommendation strategies that will be powered by our two new tag-based and tweet-based sources. Firstly though, we will now discuss the transition from tweets to tags and what this means for our recommendation system.
4.2 From Tweets to Tags

The objective of this chapter is three-fold. Firstly, our primary aim is to understand the effectiveness of recommendation, in terms of the classical evaluation metrics of precision and coverage. We also wish to re-examine our profiling source, namely tweets, and verify its effectiveness as an indicator of users’ interests (RC 1). Secondly, we are particularly interested in understanding the relative effectiveness of different types of interest data when it comes to making good recommendations. By leveraging the social graph-based profiling techniques we will further evaluate these approaches firstly described in Section 3.2, but now in a larger scale evaluation. Thirdly, we are interested in exploring the content sources available to model user interests and if there is a trade-off if we move between noisy but plentiful tweet terms to scarce but informative tags? Also, does the source of these tags have a bearing on the effectiveness of the framework at recommending friends to follow? In line with RC 3 we will evaluate these questions and the individual merits, if any, of using our new sources of profiling information.

4.3 Tag-based Interests

As discussed, one of the main purposes of this chapter is to identify different sources to use as profiling information for our recommendation framework. Are there other sources of profiling information that may provide a fundamentally stronger signal than potentially noisy tweets? In what follows we describe two tag-based alternatives. Because the tag-sets are manually curated and generally succinct they should provide for an interesting comparison when profiling against tweets. But, as we shall see, these different
tag-based approaches make very different assumptions about the relationship between tags and users. Firstly, in an attempt to structure and remove noise from our profiles, we will examine user-based tagging. User-based tags differ from their list-based counterparts, as these tags have been identified by the user themselves as their set of ‘interests’. Whereas list-based tags are equally attributed to all members of a list.

4.3.1 User-tags

To source our user-tags we used WeFollow. WeFollow users provide succinct interest tags for their profiles, using the example of social media expert Pete Cashmore of Mashable\(^7\), Pete is tagged with tech, socialmedia and news on WeFollow. We can use these tags as another type of profiling information and, as before, with tweet profiling, draw on the tags that are associated with the target user, their friends and their followers, as shown in Equations 4.1 to 4.3. This approach to profiling a user will hopefully provide for a clearer representation of the user, by harnessing the user’s explicitly indicated interests. In Equation 4.1 shown is a profile formed for a user where the utilised source is user-tags and contains the tags directly associated with that user. Equation 4.2 profiles \(U_T\) based on the tags associated with the set of \(U_T\)’s friends; likewise, Equation 4.2 does similar, but focuses on the set of \(U_T\)’s followers tags.

\[
P(U_T, userTags) = tags(U_T) = \{t_1, ..., t_k\} \quad (4.1)
\]

\[
P(friends(U_T), userTags) = \bigcup_{\forall f_i \in friends(U_T)} (tags(f_i)) \quad (4.2)
\]

\(^7\)http://mashable.com/
There are, of course, inherent downsides with the succinct nature of these tags. Are they so refined that they are unique to users? And, is the amount of content available too succinct to model users? Conversely, are the tags so generic that a multitude of users are tagged with the same tag e.g. *tech*.

Another potential problem with using tags is that a lesser concern with tweets is the nature of a user’s personal impression of their profile. If a user considers himself/herself a football aficionado, but only contributes tweets about technology. This would surface in tweet-based profiling but, potentially not in tag-based. Although, profiling based on friends and followers should alleviate these concerns.

### 4.3.2 List-tags

A large number of lists have been curated on diverse topics from jazz music to artificial intelligence, and everything in between. These lists are interesting for a couple of important reasons. First of all they represent independent opinions about the interests of members. If *John* has added *Bob* to a list on *Gadgets and Technology* then it suggests that *Bob* is relevant to this topic. Moreover, third-party services such as Listorious have created a directory of these lists and annotated them with tags. For example, at the time of writing the *Social Media* list, curated by Pete Cashmore of Mashable it included 102 people and has attracted more than 10,000 followers and has been tagged with terms such as *twitter, marketing, socialmedia, tech, web*. 
These list-based tags can also be used for the purpose of profiling users. For example, we can define a set of tags for a user $U_T$ based on the lists that $U_T$ is a member of (Equation 4.4) and the tags associated with these lists (Equation 4.5). We can then profile $U_T$ based on the tags from the lists they are contained within (Equation 4.6). Likewise, the user can be profiled based on the tags from friends contained within lists (Equation 4.7) and profiled on the tags from followers of $U_T$ that are contained within lists (Equation 4.8). In Figure 4.2 we illustrate how these tags are applied to users in lists.

\[
\text{lists}(U_T) = \{L_1, \ldots, L_m\} \quad (4.4)
\]

\[
\text{tags}(L) = \{t_1, \ldots, t_n\} \quad (4.5)
\]

\[
P(U_T, \text{listTags}) = \bigcup_{\forall L_i \in \text{lists}(U_T)} \text{tags}(L_i) \quad (4.6)
\]

\[
P(\text{friends}(U_T), \text{listTags}) = \bigcup_{\forall f_i \in \text{friends}(U_T)} \text{tags}(f_i) \quad (4.7)
\]

\[
P(\text{followers}(U_T), \text{listTags}) = \bigcup_{\forall g_i \in \text{followers}(U_T)} \text{tags}(g_i) \quad (4.8)
\]

List-based tags are associated with lists of users and not the users themselves. And to profile users using these tags we assume that tags for the list can be transferred to the users who are members of the list so that the tags associated with a user is the union of the tags of the lists that contain the
user. This is, of course, an imperfect assumption in at least two respects. First, it is unlikely that lists containing a user will reflect all of the interests of the user since these lists are typically created by third-parties. Although, these lists can conversely be well moderated and highly representative of an interest too. Secondly, the tags were assigned to the list as a whole and not to individual users and so they may not apply to all list members to the same degree. Nevertheless this approach is worth exploring. It will likely provide a more plentiful source of tags when compared to a user-based tagging approach and it will be interesting to compare the recommendation performance of this type of profile data to that obtained for profiles built using the more direct approach of using user-based tags.
4.4 Evaluation

The natural progression in the evaluation of the merits of our CB user profiling strategies from Chapter 3 is to identify additional recommendation strategies to further explore new avenues of precision, adding to the three content strategies $S_1 - S_3$, now named $Source/User$, $Source/Fri$, and $Source/Fol$. As identified previously, we can create Hybrids between the identified groups (user, friends, followers). Approaches such as the new $Source/User+Fri$ to $Source/User+Fri+Fol$ below will explore the merits in merging these individual approaches. In all these cases we can substitute $Source$ for one of our profiling sources (Twitter user-tweets (T), WeFollow user-tags (W) or Listorious list-tags (L)). For example $user-tweets/User$ would represent a profile sourced from the tweets of a user. In this experiment we evaluate these 7 different profiling and recommendation strategies using the 3 different sources of profile information, in isolation, and in combination, to answer our RC 3. This makes some 21 different tweet or tag-based profiling strategies for indexing each user; below we show the generic strategies, in each case $Source$ will be replaced by the content source being utilised (e.g. $user-tweets/User$, $user-tags/User$ or $list-tags/User$):

1. ($Source/User$) users are represented by their own data.
2. ($Source/Fri$) users are represented by the data of their friends.
3. ($Source/Fol$) users are represented by the data of their followers.
4. ($Source/User+Fri$) a hybrid strategy in which users are represented by the combination of data from the User and Fri strategies.
5. ($Source/User+Fol$) a hybrid strategy in which users are represented by the combination of data from the User and Fol strategies.
6. \textit{(Source/Fri+Fol)} a hybrid strategy in which users are represented by the combination of data from the Fri and Fol strategies.

7. \textit{(Source/User+Fri+Fol)} a hybrid strategy in which users are represented by the data from the User, Fri, and Fol strategies.

For each user, a query will be formed from the user’s own profile data and one of their sources (e.g. user-tweets/User, user-tags/User, list-tags/User). This query will be used to search against one of the profiling strategies mentioned above for indexing users. Finally, before we continue further, it is worthwhile examining the origin of all three sources of profile information once again. Figure 4.3 depicts the 3 websites from which profiling information is sourced. Using the example of the popular technology blog @Mashable Twitter account, from left to right, we can see that user-tweets based profiling will examine the terms in tweets of the user, but also those terms in the tweets of their friends and followers. The WeFollow tags of Pete Cashmore, who moderates the @Mashable account, will be used as user-tags when profiling and, likewise, his friend/followers tags will be used, when profiling, using those strategies. Lastly, the Listorious tags, in this case only one list is depicted, but, every list that contains @Mashable and its associated tags will be linked to @Mashable when profiling with list-tags, similarly, we will do the same for his friends and followers list-tags.

4.4.1 Dataset

To begin with, the core dataset for our evaluation is contained within two key datasets of Twitter users collected from Twitter over 5 months in 2012/2013; see Figure 4.4. These core users are a collection of 20,455 Twitter users. Next we expand this core to include, for each core user,
100 of their friends and 100 of their followers. These expanded friends and followers are not contained within the core user set; we will refer to these users as the expanded users. The expanded users sets in all cases are greater than 130,000 users in size with the lowest expanded friends set made up of 115,698 users and expanded followers of 123,378; see Table 4.1. The varying figures refer to the friends and followers availability across the three services Twitter, WeFollow, Listorious. These expanded sets will form the basis for our friend and follower-based strategies described previously. In addition, we used the Twitter API, Listorious and WeFollow to extract the necessary user tweet (up to 100 tweets), user-tag and list-tag, friend, and follower information for all of these 20,455 users; We also verify that all core users and extended users are present in Twitter, WeFollow and Listorious so that we can source profiling information for them. This size of dataset was capped at 20,455 for a few reasons, our system requires not only the users data, but, also all of their friends/followers connections and tweet data. This represents a data pull in excess of 400,000 data points. We are limited in
the number of requests which can be made to the Twitter API and thus this dataset size represented the upper bounds of a feasible pull. Coupled with that, the users found via the Twitter data pull must also be present on WeFollow and Listorious, both third party applications. This adds another level of constraint to the size of dataset that can be built.

**Table 4.1:** Number of Unique IDS in Expanded Users Dataset

<table>
<thead>
<tr>
<th></th>
<th>Union Unique</th>
<th>Followers Unique</th>
<th>Friends Unique</th>
</tr>
</thead>
<tbody>
<tr>
<td>Twitter</td>
<td>131,486</td>
<td>123,378</td>
<td>115,698</td>
</tr>
<tr>
<td>WeFollow</td>
<td>156,659</td>
<td>136,617</td>
<td>129,008</td>
</tr>
<tr>
<td>Listorious</td>
<td>291,028</td>
<td>230,315</td>
<td>207,911</td>
</tr>
<tr>
<td>(T \cup W \cup L)</td>
<td>401,996</td>
<td>343,331</td>
<td>320,020</td>
</tr>
</tbody>
</table>
Summary statistics can be seen in Table 4.2 for this dataset, the table shows the mean number of user query terms used per target user query, per profiling source, averaged over $k=10$ splits using random subsampling validation on samples of 1000 users (Equation 2.13). The Twitter queries represent terms that appear more than 3 times across a user’s tweets and up to top 50. Listorious queries are sourced from the lists a user is contained in and WeFollow tags are taken from a user’s web profile. It is interesting to note that the length of these queries becomes smaller as we transition away from potentially noisy tweet terms to more succinct directly assigned tags.

**Table 4.2**: Mean Number of Query Terms in Dataset Averaged over $k=10$ Splits using Random Subsampling Validation on Samples of 1000 Users.

<table>
<thead>
<tr>
<th>Query:</th>
<th>User-tweets</th>
<th>List-tags</th>
<th>User-tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Terms</td>
<td>32.71</td>
<td>21.14</td>
<td>4.78</td>
</tr>
</tbody>
</table>

### 4.4.2 Methodology

Our evaluation includes 21 different recommendation strategies: the 7 user-tweets based and 14 tag-based strategies outlined in Section 4.4, where $SOURCE$ is replaced with either user-tweets ($Twitter$), user-tags ($WeFollow$) or list-tags ($Listorious$) depending on the source being used. In each case, we averaged over $k=10$ splits using random subsampling validation on samples of 1000 users to measure our precision for each strategy. Specifically, for each 1000 user subset, each user is treated as a test user. Their profile is matched against the relevant user index and a set of recommendations are generated. To measure precision, we count the number of these users who are already friends of the test user; these are people the test user
has already chosen to follow and the ability to recommend these users is thus a strong indicator of recommendation relevance.

4.4.3 A Summary Analysis

A summary of our evaluation results is presented in Figure 4.5 as a bar chart of mean average precision (MAP) for each of the recommendation approaches combined. The mean average precision for each source (user-tweets, user-tags, list-tags) is simply the average precision for all the strategies outlined in Section 4.4 across all results for the $k=10$ splits using random subsampling validation on samples of 1000. We can clearly see that tag-based approaches perform better than user-tweets, both tag-based achieve a MAP of over 15% compared to 12% for user-tweets. Overall, the clear winner is the list-tag profiling approach with MAP of 20%. So even though list-tags reflect indirect connections to individual users, compared to more direct user-tags, they still provide a stronger signal. Moreover, despite the relative sparsity of tags, compared to tweet terms, both tag-based methods outperform the user-tweets based approaches. In the next section, we will carry out a detailed exploration into these results to further explore the reasoning and individual benefits of the various strategies.

4.4.4 Tweets vs. Tags

The overall precision results are presented in Figures 4.6 - 4.8. In each case we present a graph of precision and coverage results, and each graph presents the 7 different recommendation approaches for each one of the 3 sources averaged over $k=10$ splits using random subsampling validation on samples of 1000 users.
Focusing on the results for the CB strategies first (Figure 4.6), and using the same recommendation framework as our initial evaluation and in Chapter 3, albeit on a different dataset of users, we can see that in general the different CB approaches generate recommendations with precision scores ranging from 13-16%.

The user-tweets/Fol and user-tweets/Fri approach performs better than the user-tweets/User approach but the winner is the user-tweets/User+Fol followed closely by the strategy that models all the user information user-tweets/User+Fol+Fri with precision scores that are higher than any of the other strategies, we note similar results in Chapter 3.

It is interesting that the user-tweets/User strategy does worse than either of the user-tweets/Fri or user-tweets/Fol strategies, indicating that the user’s
own content isn’t a better predictor of friends than the tweets of friends or followers. Maybe this is to be expected though as a user will selectively choose Friends to follow to consume tweets based on certain sets of topical interests that they themselves may not produce on those topics.

It is also worth noting that both user-tweets/Fri and user-tweets/Fol perform very similarly from a precision standpoint. Remember a user does not connect to a follower, a follower connects to her, but yet followers turn out to be a useful source of recommendation content when compared to a user’s explicit friends. Coverage for all approaches remains in the high 90’s, which allows for the ability to perform recommendations for users in most cases.

Figure 4.6: Precision (bar) and Coverage (line) Results for Twitter user-tweets based Strategies Averaged over $k=10$ Splits Using Random Subsampling Validation on Samples of 1000 Users.

Figure 4.7 presents the precision results for the 7 tag-based recommendation strategies for the list-tag based approaches. Broadly speaking, we can see
an increase in the range of recommendation precision values when compared to the user-tweets based strategies.

For a start, the list-tags/User+Fri+Fol based approach again produces far and away the best precision values for this set (just about 21%) and this is about 4-5% better than the best performer of the user-tweets strategies. Notably, across the board the results share a similar pattern to that of the user-tweets approaches, albeit with higher precision scores ranging from 18-21%. The secondary benefit with this list-tag based approach is that the coverage averaged across the 10 splits hits 100% for 5/7 strategies whereas the user-tweets approach gets highs of 99%.

The next source we are going to explore is that of the hopefully more succinct representation of the user from user-based tags. Figure 4.8 shows again the precision averaged across the $k=10$ splits. Two interesting points with this
graph are, succinct tags give a performance increase, as we expected, over user-tweets strategies. The second point is that although these succinct tags give better performance than user-tweets, the shortcomings come with a hit in coverage scores.

Contrasting with the list-tag based approach, user-tags perform lower, with precision scores ranging from 13-18%. Interestingly, there seems to be a trade off in precision when it comes to, firstly, quantity of tags, and, secondly, their quality. Whilst both tag-based approaches have well structured tags, the quantity that list-tag based approaches have seem to substantially aid in building better user profiles.

**WeFollow Tag Recommendation Strategies**

![Figure 4.8: Precision (bar) and Coverage (line) Results for WeFollow User-tag Strategies Averaged over $k=10$ Splits Using Random Subsampling Validation on Samples of 1000 Users.](image)

We can see that the tags of friends perform slightly worse than the tags of followers across all sources. This is interesting because, once again, users
choose to follow their friends whereas, by and large, they exert little or no control over who follows them. One would think that a follower list is likely to contain low quality or even spam users. The explanation for this probably depends on a number of factors. First, users follow other users based on a want to consume information that reflects their interests, likewise, the followers of a target user follow that user because of shared consumption needs. Followers, whilst sometimes spammers, tweets that these spammers would produce wouldn’t contain terms that would reflect a target user’s interests and would be flagged as useless spam at recommendation time.

Finally, as all of our friend/followers in the dataset had to be members of Listorious, WeFollow and Twitter, it is unlikely that users are going to be listed or tagged if they aren’t of interest to others. In the next section we will discuss in more detail the lessons learned and the core contributions to be taken from this chapter in relation to our research contributions.

4.5 Lessons Learned

Our evaluations thus far has outlined the utility in our framework and its ability to recommend friends to follow on Twitter. Throughout these battery of evaluations we have tested various metrics and strategies. We have learned some key lessons and some of the core findings are outlined below.

4.5.1 RTW Data Effective at Modelling users

In keeping with RC 1 & 2 there is reason to be encouraged that friend recommendation and user modelling, which utilises real-time content produced
by users, their friends and followers, is effective at producing both good recommendations and modelling a user’s interests accurately. Our initial evaluation in Section 3.4 showcased that high accuracy could be achieved with quantitative offline trials but, also qualitative precision could be achieved through a live user trial, Section 3.4.4. We note here that a similar live user trial to test WeFollow or Listorious tags as a source isn’t as straightforward to set up because neither service has full Twitter user coverage and, as such, for a trial user, we could not guarantee the trial user or their friends/followers would have content available to profile them. Nevertheless the offline evaluations are standard practice in recommendation systems research, and through the validation we carried out, we would be confident of their applicability if content is available to model users.

We also showcased that freeform generated tags from web services such as WeFollow and Listorious can be used to increase the recommendation precision using our framework and this insight justifies RC 3. We also learned that the social aspect of the chosen strategy matters, and utilising a user’s friends and followers, when modelling, helps form a better representation of the user. Interestingly, too, these friend-based and follower-based techniques outperformed those of the user only approach, allowing us to conclude that the people a user follows can define them, but, secondly, the users that follow our user can also define their interests.

### 4.5.2 A Comparison of Tags and Tweets

Our second insight is that by transitioning from potentially noisy tweets to more refined tags can have an impact on performance. Although the abundance of terms allows for this approach to be considered as a viable
approach over better performing tag-based solutions, because it doesn’t fall prey to the pitfall of lower coverage as with the user-tags approach.

4.5.3 A Comparison of List-tags and User-tags

Our initial intuition would be that the succinct nature of the user-tags would provide for a clean representation of our user’s interests and provide for better recommendations for users modelled in this way. This did not hold true though, contrasting the effectiveness of the user-tags to the performance of their list-tags counterparts, list-tags perform better. Potentially, though this can be explained by the nature of query formation for our test user, user-tag profile queries are composed of the set of tags a user has defined as their set of core interests although these tags could be too novel or too focused. As such, these queries perform poorly at finding others who share those sets of interests. In comparison, list-tags are aggregated across a collection of lists a user is in, so tags will be shared amongst users in those lists, so are less inherently novel. The tags may appear frequently across lists too and as such are potentially generic and less focused allowing for better query formation.

4.6 Conclusions

In this chapter we have proposed that Twitter users can be usefully modelled by both the terms in tweets and tags associated with that user. We have demonstrated how these profiles can be used as the basis for friend recommendation. An offline evaluation, based on real-user data, suggests that this recommendation system which utilises these sources is capable of
delivering high-quality recommendations. Our analysis suggests and reiterates our sentiments discussed in Chapter 3 that noisy as Twitter tweet content is, it does provide a useful profiling and recommendation signal but, we have also shown that by bootstrapping our profiles with a number of novel tag-based approaches using tags extracted from a large collection of manually annotated, curated Twitter lists and direct tagged users we can increase the recommendation precision of our framework.

The main conclusions to be drawn from this chapter are that, whilst friend recommendation on micro-blogging sites, such as Twitter, can be challenging, we believe we have come up with various techniques to aid recommendation. Whilst some techniques perform better than others, such as the list-tags based approaches, the recommendation precision is quite high across all techniques.

Up until this point we have been modelling users based on the sets of interests we could identify for them by utilising different sources of profiling information. This approach represents a traditional approach to assigning interests to a user, but perhaps we now need to move beyond this. These traditional profiles we feel don’t fully reflect users on social networks. Users on Twitter, in particular, have many interests and each part of their profile should reflect those interests. Indeed, those interests shared with their social graph must also have an impact on the way in which a user is modelled. The fine grain differences and similarities between the user, their friends, and followers has spurred us on to re-examine how we form our user profile.

In the next chapter we will explore this further as we develop and evaluate a new user profile to use in our recommendation framework which will better reflect the multifaceted nature of users’ interests.
5

Profiling Users Multifaceted Interests

5.1 Introduction

Most friend-finding services adopt a similarity-based perspective that is commonplace in traditional search and recommendation tasks by selecting recommendations that are in some sense similar to a given query, or by selecting recommendations from a community of users who are similar to the target profile [Armentano et al., 2013, Chen et al., 2012, Kim and Shim, 2011]. Accordingly, the basic pattern that drives recommendation is some assessment of shared interests between users. This makes sense in many social networking contexts: on services like Facebook and Twitter it is reasonable to make it easy for users to connect with others who have similar shared interests. However, sometimes an over emphasis on direct similarity through shared interests may limit the value of friend-finding services for the
end user. Indeed, in some social networks, shared interests play a less important role. For example, recently a new type of social network service has emerged in the area of work markets as services like TaskRabbit\footnote{https://www.taskrabbit.com} and SkillPages\footnote{http://www.skillpages.com} allow people to post paid errands or short-term job opportunities for others to complete. In SkillPages for example, registered users can specify their own particular skills and can also post opportunities for other different skilled users. In this type of social network there is little direct overlap between the skills or interests of a user and the opportunities they post and yet a key task is to recommend users who share complementary skills in order to proactively rank candidates for particular opportunities. We wish to build a model that can reflect these and other different connection finding intents and incorporate that model into our friend recommendation framework. A simple scenario in Twitter for instance would be that we can connect with other users because of our shared interests, but there is no mechanism to connect with others who have different but complimentary interests. We want our model and framework to cater for this amongst other scenarios.

One of our core aims in this work is to move away from the type of one-user-one-profile (monolithic profile) approach to profiling and recommendation that is commonly adopted. We believe that such monolithic profiles serve only to disguise different sets of user interests. We refer to these different sets as profile facets and argue that by making this type of profile structure explicit we can enable a more flexible approach to recommendation and personalisation, one that reflects different profile contexts, user intents, and recommendation needs. Szomszor et al. [2008] have explored unifying interests from various contexts when modelling the user by merging all the user’s various profiles from different web services. Whilst allowing for enrichment of the user’s profile we believe this could also add noise and as we
have shown the user’s assigned interests aren’t always the best indicators when trying to find others they would follow.

In this chapter our object is to evaluate all of the RC’s 1-4 set out in Section 1.3. For RC 1, which was first explored in Chapter 3, then further evaluated in Chapter 4, we will again revisit it here by focusing on the source of recommendation knowledge namely, users’ tweets. Although, in this chapter and in line with RC 2 we will make different assumptions about how this source is utilised and what, indeed, the impact of a user’s social connections have on our new model. We will discuss these assumptions and our model in Section 5.2. RC 3 will be revisited also in this chapter as we investigate the alternative sources of recommendation knowledge identified in Chapter 3 and how these sources impact on the precision, coverage and diversity of recommendations. Finally, we will take our first look at RC 4 as we discuss our new model that aims to reflect the diverse sets of interests a user can have and we will evaluate our identified approach in the evaluation Section 5.4. In the next section though we will take our first steps away from traditional monolithic profiles as we introduce our new friend recommendation profiling technique.

5.2 A Multifaceted Profile Structure

Many social services are based around a similar social graph structure by allowing users to connect to each other. On Facebook, when two users become “friends”, a link is created in the social graph between them. It’s a two-way link because, in Facebook, the friendship model is symmetrical. So, if John is a friend of Ashlyn, then Ashlyn is a friend of John and vice versa. On Twitter it’s asymmetrical: John can follow Ashlyn, creating a
Figure 5.1: (a) Multifaceted User Profile, (b) Multifaceted User Profile of Twitter User @BarackObama.

one-way link from John to Ashlyn; Ashlyn is a friend of John. But John is not by default a friend of Ashlyn, although Ashlyn can of course choose to follow John if she wishes.

These structures provide a starting point for a multifaceted user profile. Specifically, for a given target user we can consider the relationships between three different social groupings: (1) the target user’s own interests; (2) the interests of the user’s friends; and (3) the interests of the user’s followers. These interests can overlap in some obvious ways. In fact Figure 5.1 (a) and via Equations 5.1 - 5.7 naturally partitions a target user’s profile-space into seven distinct facets \((F_1, ..., F_7)\). We first discussed this partitioning in [Hannon et al., 2012] with an initial attempt to building a multifaceted user model from raw Twitter data.

At the centre of the profile-space is facet \(F_1\) (Equation 5.1), which represents the user’s core interests. These are interests shared between the user,
their friends, and their followers. Facets $F_2$, $F_3$ and $F_4$ (Equation 5.2-5.4) capture interests shared between pairs of sources: user and followers, user and friends, and friends and followers. For example facet $F_4$ corresponds to those shared interests between a user’s friends and followers but not the user’s own interests. Finally, facets $F_5$, $F_6$ and $F_7$ (Equation 5.5-5.7) reflect exclusive sets of interests either those unique to the user, or their friends, or their followers. For instance here facet $F_5$ corresponds to those interests that are unique to the user’s followers that aren’t interests of the user nor the user’s friends.

![Figure 5.2: Forming a Multifaceted Profile for a User using their Own, their Friends, and their Followers Tags.](image)

In Figure 5.2 we can see a worked example into just how our multifaceted model is formed for test user Bob. Taking the three social groupings, Bob
Chapter 5. Profiling Users Multifaceted Interests

himself, his friends and his followers, each group has a set of tags core to that grouping. To form the facets for our model we take the union and intersection of tags between groups, for example $F_1$ the facet representing the shared interests between Bob, his friends and his followers only contains the tag *iPhone*. This is because *iPhone* is the only shared tag between those social groupings.

\[
F_1(U_T) = P(U_T, type) \cap P(\text{friends}(U_T), type) \cap P(\text{followers}(U_T), type) \tag{5.1}
\]

\[
F_2(U_T) = P(U_T, type) \cap P(\text{followers}(U_T), type) - F_1 \tag{5.2}
\]

\[
F_3(U_T) = P(U_T, type) \cap P(\text{friends}(U_T), type) - F_1 \tag{5.3}
\]

\[
F_4(U_T) = P(\text{followers}(U_T), type) \cap P(\text{followers}(U_T), type) - F_1 \tag{5.4}
\]

\[
F_5(U_T) = P(\text{followers}(U_T), type) - (F_1 \cup F_2 \cup F_4) \tag{5.5}
\]

\[
F_6(U_T) = P(U_T, type) - (F_1 \cup F_2 \cup F_3) \tag{5.6}
\]

\[
F_7(U_T) = P(\text{friends}(U_T), type) - (F_1 \cup F_3 \cup F_4) \tag{5.7}
\]

By partitioning the profile-space in this way we can expose different collections of interests that can be used in different ways for different types of recommendation scenarios; e.g., recommending users that match a user’s
core interests ($F_1$) or other users that match their peripheral interests (such as $F_2$, $F_3$ or $F_4$).

Moreover, different facets can be grouped according to their likely relevance to the target user and for this reason we can also organise the profile facets into *regions* as shown in Figure 5.3. For example, as mentioned above facet $F_1$ corresponds to the user’s core or *primary* interest region. Facets $F_2, F_3, F_6$ correspond to the user’s *secondary* interests; interests that are partially shared with the target user. While facets $F_4, F_5, F_7$ correspond to peripheral or *tertiary* interests region. This region contains interests that are not directly shared by target user at all, but are found among the user’s friends and/or followers. With these facets and regions we can now model user’s interests in a much more focused way and form recommendations for these users based on desired interest intents (e.g finding me users based on my core interests).

### 5.3 Multifaceted Profiles for Friend Recommendation

In Section 5.2 we introduced our multifaceted model and in this section we shall discuss how we can harness it for friend recommendation. In Figure 5.4, we identify the role in which our facets and regions play within our framework. Simply put the user model described in Figure 5.1 and via Equations 5.1 to 5.7 represent the facets of the user’s profile which we can use on their own as profile queries or by grouping facets together into regions we can describe the user by their identified core, secondary, tertiary interests regions, as shown in Figure 5.3. These regions can then be combined to form user indexes used when forming recommendations for users in Figure 5.4.
Specifically we will use different facets as different types of user query to elicit different types of friend recommendations. e.g by using facet $F_4$ as a query we can look for users who share interests with the target user’s friends and followers while excluding users who share interests with the target user. And we will use different regions as descriptors of users based on their core, core+secondary, tertiary or indeed all interests. For example the recommendable users can be indexed on their core interests, but the test user can search over these using one of their faceted queries, e.g. facet $F_3$ can be used as a query over the index of users profiled on their core interests to form recommendations. In Figure 5.5, as way of a summary we show the updated system architecture from the one seen in Figure 3.3, the profile queries and index now represent our multifaceted model but, the remainder
of the framework remains unchanged utilising Lucene as the retrieval and indexing system and our sources as user-tweets, list-tags and user-tags.

To evaluate the performance of our framework combined with the new multifaceted model we will follow the evaluation setup as in Section 4.4 by focusing on the two evaluation metrics of precision and coverage. Precision as before will be the measure of the number of user’s friend we can identify with a set of recommendations and coverage will rely on an overlap between the recommendation facet query and the region index, such that some recommendations can be formed. In the next section we will focus in on our evaluations which will aid us in gauging the effectiveness of our setup.
5.4 Evaluation

So far we have described in Sections 5.2 & 5.3 our new approach to modelling users based on their faceted interests by incorporating this model into our recommendation framework, see Figure 5.5. In the following sections we will test its ability at suggesting new social connections to users of services like Twitter. Our desire is to develop our framework so that it is flexible enough to make different types of recommendations, for different user needs, but powerful enough to make accurate and relevant recommendations. We will also endeavour to evaluate this recommendation framework using a large scale, live-user dataset collected from Twitter.
5.4.1 Methodology

Using the dataset of 20,455 users and their associated Twitter, WeFollow, Listorious information as discussed in Section 4.4.1, we evaluate a total of 84 (3 sources x 7 facets x 4 regions) new recommendation strategies based on the combination of three different types of profiling sources (user-tweets (Twitter), user-tags (WeFollow), list-tags (Listorious)), 7 different profile facets as recommendation queries \((F_1, ..., F_7)\), and 4 different regions as profile indexes (core/primary, core+secondary, tertiary, everything). We selected these regions to index our users because we wanted to test the impact on recommendations as we transition from modelling the user on their core interests only and then adding on their secondary, tertiary interests. We chose to also highlight the tertiary index on its own too in our evaluations as we wanted to gauge if there was anything to be gained from modelling users on interests that have no overlap with their own set of interests. For clarity, in the following sections, when we refer to facets, we will be discussing them from a profile query point of view, sources will refer to either user-tweets, user-tags or list-tags, regions will refer to the indexes core/primary, core+secondary, tertiary and everything. These regions are formed from grouping facets together.

For each condition we carry out a standard leave-one-out evaluation that is commonplace in recommendation systems research. Specifically, we treat each of the 20,455 users in turn as a new test user, \(U_T\), for recommendation purposes. For each \(U_T\) we produce 21 (3 sources x 7 facets) different recommendation queries, based on all combinations of profile types and facets. The remaining 20,454 core users and all expanded users form the basis of the recommendation index. The 20,454 as recommendable users and the expanded users as content facets of those users’ region indexes. Each one of
these 21 facet queries is used to generate a set of \( k \) \((k = 20)\) recommendations for \( U_T \) from 4 different recommendation indexes based on the different profile regions. Our choice at selected \( k=20 \) is again spurred on by the fact that it’s unlikely a live user would follow excessive amounts of users in one sitting and 20 represents a digestible size of user recommendations.

### 5.4.2 A Summary Analysis

To evaluate the accuracy of a set of \( k \) recommendations we compute a conventional **precision** metric by comparing the \( k \) recommended users to \( U_T \)’s actual friends, calculating the percentage of recommendations that are already friends of the user. As discussed in Section 3.4.4, this is understood to be a very conservative precision metric because in reality it is unsafe to conclude that a recommended user is irrelevant just because he/she is not one of \( U_T \)’s current friends. Nevertheless this approach provides a useful baseline measure of recommendation accuracy.

To evaluate the recommendation system’s ability to produce recommendations we calculate **coverage**, this metric represents the percentage of times a recommendation can be made for a given user. Remember that the ability to make recommendations relies on an overlap between the recommendation facet query and the region index. Such an overlap is not guaranteed to exist and in fact facet queries and indexes made up of sparse interest tags or poor vocabulary are likely to suffer low coverage compared to facet queries and indexes made up of more plentiful descriptive terms or tag-based interests.

Using this approach we can calculate mean precision and coverage scores for each of the 84 (3 sources x 7 facets x 4 regions) different combinations of interest source, profile facets as recommendation queries, and profile regions as recommendations indexes. That’s a lot of evaluation data to absorb. To
better help us to identify interesting results, and focus on revealing findings, we will initially present summary precision and coverage results, across interest sources, profile facets, and recommendation indexes and discuss in more detail in the subsequent sections.

These summary results are shown in Figure 5.6 as a combination of bar-chart (precision results) and line-graph (coverage results). In each case we calculate a mean precision and coverage score for each one of the three different interest sources (user-tweets vs user-tags vs list-tags), the seven query facets ($F_1, \ldots, F_7$), and the four types of recommendation indexes ($C, C + S, T, E$). These results do not represent a like-for-like comparison between different setups but, they do provide us with a high level overview of the different aspects of our recommendation system approaches. For example to calculate the precision score for user-tag conditions we compute the mean precision

![Figure 5.6: Summary Precision (bar) and Coverage (line) Results Averaged across Recommendation Indexes, Profile Facets, and Interest Sources.](image-url)
for the relevant 28 different test-runs (all combinations of the 4 region indexes and the 7 query facets) when profiles are composed of interests that are made of of user-tags, and similarly for coverage. In the following sections we will explore some initial findings we can conclude from Figure 5.6.

5.4.2.1 Precision and Coverage by Interest Source

We can see that there is a difference in precision when we vary the type of interests that are used to profile users. The average precision for user-tweets source is just over 6% compared with approximately 8% with their tag-based counterparts; in other words between 6-8% of the recommendations are actually followed by the target user using this source. As expected the noise that is inherent in user-tweets limits their effectiveness when it comes to delivering accurate recommendations. Likewise, the more direct user-tags do offer a minor improvement in precision compared to list-tags, which is as predicted, based on the previous observation that list-tags are only indirectly associated with individual users and thus are less likely to be on-point than user-tags. Although the indirect list-tags do still outperform the user-tweets source.

We can see very stark differences between user-tweets and tag-based sources when it comes to recommendation coverage however. User-tweets sources achieve a coverage rate of 96% – they can generate at least some recommendations for 96% of target users – compared to only 67% and 70% for user-tags and list-tags, respectively. List-tags are more plentiful than user-tags so it is not surprising to see a coverage benefit for the former here, although the scale of the benefit is small relative to the difference in availability of these different types of tags. And in general, depending on the recommendation scenario, the small reduction in average precision, due to
user-tweets source, may be comfortably offset by the much greater coverage when compared to the tag-based sources.

5.4.2.2 Precision and Coverage by Query Facet

The next group of results focus on the average precision and coverage scores but this time focusing on facets that are used as recommendation queries. Remember, by segmenting user profiles into facets this makes it possible to generate recommendations that focus on certain types of user interests whether core or peripheral, etc. This provides greater recommendation flexibility than if monolithic profiles were adopted, as in Chapter 3, 4. But how effective are these query facets when it comes to generating relevant recommendations? And are there any differences in performance across the different facets? Figure 5.6 helps us to start to answer these questions and this time there are some very major differences in both precision and coverage across these seven different query conditions.

Two query facets stand out in particular, both in terms of their precision and coverage characteristics. Firstly, focusing on $F_1$ the set of interests (whether the source is user-tweets, user-tags, or list-tags) that is shared between the target user, their friends, and their followers. Using $F_1$ as the recommendation query generates an average precision of almost 12% and coverage in excess of 98%. Perhaps it is not surprising that this type of recommendation query generates among the most relevant recommendations? After all it is a query that captures a user’s core interests, those that are not only held by the user but also by their followers and friends.

What is more surprising, perhaps, is that the performance of $F_1$ does not come out on top in terms of overall recommendation precision or coverage. Instead we see that, on average, queries based on facet $F_4$ generate the most
relevant recommendation lists, achieving a precision of more than 14% and with 100% coverage. Query facet $F_4$ contains those user interests that are shared between the friends and followers of the target user but not by the target user themselves. In other words, $F_4$ represents those topics that the target user is interested in but that are not topics that they themselves tend to tweet about.

Indeed it could be argued that $F_4$ probably represents those topics for which the user is most receptive when it comes to receiving friend recommendations. This is because $F_4$ probably represents topics that the user has an interest in but is not so much of an expert on. These interests are shared by the user’s friends and followers but, not themselves. To ensure that they keep up with these interests they have carefully curated their social network to ensure that it contains trusted sources to cover these interests.

The precision and coverage scores are less impressive and more variable when it comes to the other profile facets. For example, facets $F_2$ and $F_3$ correspond to those interests that the target user uniquely shares with their followers ($F_2$) or their friends ($F_3$). Both facets have virtually identical precision and coverage characteristics. Precision and coverage are both low indicating that these facets are reasonably sparsely populated and fail to provide a strong recommendation signal. Maybe it is to be expected that these facets are sparsely populated because if interests are shared by a user and their friends, this would encourage followers to follow our user and as such the shared interests between the three groups would move towards facet $F_1$.

A similar effect is seen for facets $F_5$ and $F_7$ (that is, identical precision and coverage). These facets correspond to interests that are unique to either the target user’s followers ($F_5$) and friends ($F_7$). This time, however, coverage
is very high (approximately 99%) because these facets are well populated with interests drawn from a large number of followers or friends. It is somewhat interesting that we do not find $F_7$ to offer better precision than $F_5$. We might have expected to find this because, as a practical matter, Twitter users are probably a lot more careful about curating their friends than caring about who follows them. A user has to explicitly choose a friend but can accumulate followers more or less automatically. Consequently, there is always the risk that our followers quickly become polluted with spam Twitter accounts and other types of cyber-stalkers whose interests are unlikely to fit our own. Although this is more likely to affect user-tweets approaches rather than tag sourced due to the initial user effort involved in tagging users. Nevertheless, since the unique interests of our followers are just as effective at generating relevant connections as the unique interests of our (carefully curated) friends, this risk does not seem to be borne out.

Using $F_6$ as the recommendation facet query offers the poorest precision (5%) across all of the conditions coupled with only moderate coverage (63%). At first this might seem surprising. After all, $F_6$ represents interests that are drawn directly from the target user. However, in our faceted model these are not considered core interests ($F_1$) precisely because, by definition, they are not shared by any of the user’s followers and friends. To the extent that they are interests of the target user that are either not so strongly held that the user has elected to follow others who share these interests, or are so niche that the target user has been unable to find others who share them. Nor have other users elected to follow the target user for these interests. And thus, in all likelihood, these interests will correspond perhaps to fleeting interests on the part of the user, topics that they may have tweeted about a few times but are far from persistent interests, or peculiar niche subjects. Certainly the lower precision bears this out.
5.4.2.3 Precision and Coverage by Recommendation Index

Finally, in this part of the evaluation we come to look at how recommendation performance is impacted when we change the data used to index profiles. In particular, we consider four different types of region indexes based on the users’ core interests \((C)\) from profile facet \(F_1\), the combination of their core and secondary interests \((C + S)\) from facets \(F_1, F_2, F_3, F_6\), their tertiary interests \((T)\) from facets \(F_4, F_5, F_7\), and finally their entire profiles \((E)\) using facets \(F_1, ..., F_7\).

Recommendation coverage is not significantly impacted by different recommendation indexes; it remains close to 78% across the 4 conditions. But precision is very much affected. Unsurprisingly, representing and indexing users by all of their interests \((E)\) delivers the best precision (almost 10%); in other words it is worthwhile considering complex region indexes of users even if we only query using different facets of their profile.

At the other end of the indexing continuum we can see that indexing users by their core interests \((C)\) alone exacts a significant cost as precision falls to less than 6%. We might reasonably speculate that the absence of secondary and tertiary interests severely limits the type of recommendations that can be made. But that speculation would only be partly correct because we can see that in fact the addition of secondary interests into the recommendation index \((C + S)\) does surprisingly little to improve precision. Instead it is the tertiary interests that bring so much more when it comes to improving our ability to make relevant recommendations. In fact the precision of recommendation indexes made up only of the user’s tertiary region is more than 8%, almost as high as when we use a complete profile. It is surprising that recommendation indexes made up of just tertiary interests seem to be better drivers of good recommendations than the user’s secondary interest,
let alone their core interests. But, perhaps this is too harsh a conclusion, after all, tertiary interests represent those interests of the user’s friends and followers. Those friends are actively sought out to satisfy an interest of our user but, in this index’s case our user isn’t representative of that interest enough for it to be associated with them. The followers in this instance, of course, could just be reciprocating the befriending, but more likely our user’s followers have identified this user as a good filter and whilst they may not contribute solely on an interest, like the user’s friends, he/she may when it’s an interesting piece of information and as such warrants following.

5.4.3 Initial Findings

In summary, the above analysis highlights some important initial findings:

(i) Recommendation coverage is strongly influenced by the type of interest source (whether user-tweets or tags), while the influence on precision is somewhat weaker. In general, curated tags offer better precision than user-tweets, but at a cost to coverage.

(ii) Using different profile facets as recommendation queries has a big impact on precision and coverage, with facets $F_1$ (the user’s core shared interests) and $F_4$ (those interests shared by only friends and followers) both offering significant precision benefits with excellent coverage. This would allow a user to form recommendations for themselves based on different intents, such as looking for users who facilitate their tertiary or core interests.

(iii) Making recommendation region indexes as rich as possible does benefit precision, but it is the user’s peripheral or tertiary region interests that
appear to offer the lion’s share of precision benefit, while secondary interests offer little or no precision uplift.

Using these initial findings, in the next section we will dive deeper into the various recommendation approaches on their own and focus in on modelling of users varied interests in line with RC 3. By focusing in on the individual techniques we will uncover some surprise findings which were masked by the high level overview discussed in our initial findings. In Sections 5.4.4 we discuss the merits of our approaches in the context of the facets, regions and sources. By focusing in on this evaluation in more detail we can now start to propose which techniques can be utilised for different recommendation intents and when different profiling sources etc. are available.

5.4.4 Detailed Analysis

These findings in Section 5.4.3 are based on a summary analysis, averaged across a variety of experimental conditions and this averaging may disguise some interesting effects across individual conditions. For this reason it is important to unpack these findings by looking at recommendation performance across individual conditions. For this, detailed precision and coverage results are presented in Figures 5.7 & 5.8 (a-d).

There are four different graphs and each one corresponds to the different type of profile region used for indexing Core (Figure 5.7 (a)), the combination of the Core and Secondary interests (Figure 5.7 (b)), their tertiary interests (Figure 5.8 (c)), and finally indexes that are made up of everything, that is, the entire set of user interests (Figure 5.8 (d)). Each graph shows the precision (bar chart) and coverage (line graph) results for recommendation queries generated from the seven different user profile facets.
(F₁, ..., F₇). And finally, within these bar charts and line graphs we present the precision and coverage scores obtained for different sources of profile interests (user-tweets, user-tags, and list-tags).

For example, we can see from Figure 5.7 (a) that when we use the user’s core interests (facet F₁) as a recommendation query, generating recommendations from a region index made up of core interest, using the source user-tweets delivers recommendations with precision of approximately 12%, while the source user-tags and list-tags achieve 15% and 20% precision respectively. The source user-tweets achieves 100% recommendation coverage while user-tags and list-tags are slightly lower at between 90% and 95%.

When we focus back on our key findings from Figure 5.6 in our individual charts we can see that the conclusions reached generally prevail. By using list-tags or user-tags interests sources this reduces the ability of the framework to perform recommendations, seen in the diminished coverage values for facets F₂, F₃, F₆.

Firstly, the benefit of using certain facets when forming the query maintains having an impact on precision and coverage across sources. Where a facet performs well for one source (e.g. user-tweets), this is generally the same for the facets using the other two sources (list-tags, user-tags). For each of our best performing facets in one of the charts in Figure 5.8, the top respective facet for the other two sources is the same, i.e F₁ in Figure 5.7 (a & b) and F₄ in Figure 5.8 (c & d).

Secondly, it is evident from these charts that the quantity of tweet terms or tags used as a source for a region index does indeed matter. The increase for users’ indexed on their tertiary region can be explained by firstly the increase in terms/tags in that index. This quantity of terms/tags seems to counteract in part for any noise present in the index. Tertiary facets
represents a collection of friends/followers of our user. This increase in interests to profile can aid when forming recommendations and couple this with the ability to form stronger queries for the individual friend/follower facets this also aids in better friend recommendation discovery. The payoff is our top strategy $F_4$, which also boasts 100% coverage. Which means that not only can precise recommendations be made but, they can be made for all users tested.

5.4.4.1 Tweets vs. Tags

Our first summary finding from Section 5.4.3 (i) was that recommendation coverage is strongly influenced by the type of interest source (user-tweets vs. tags). The influence on precision was found to be weaker, with curated tags offering somewhat better precision than user-tweets, but at a coverage cost. However the detailed results presented in Figures 5.7 (a & b) & 5.8 (c & d) show that this finding does not tell the full story.

In particular, the benefits due to tags that we saw in Figure 5.6 – tags offer about 20% relative precision improvement compared to user-tweets – are not evenly distributed across the different experimental conditions of Figures 5.7 (a & b) & Figure 5.8 (c & d). For certain types of facets, tags offer a much greater precision benefit than user-tweets. This is true for the $F_1$, $F_2$, $F_3$ conditions in Figures 5.7 (a & b) and for $F_4$, $F_5$, $F_7$ in the case of Figures 5.7 (c & d); in each case tags deliver precision scores that are at least 30% higher than the corresponding precision scores for user-tweets with their best performing facet. Moreover, while list-tags tend to perform best for the $F_1$ condition in Figures 5.7 (a & b) it is the user-tags that deliver consistently better precision in the case of $F_4, F_5, F_7$ in Figures 5.8 (c & d). In other words, list-tags significantly outperform user-tweets when it comes to
Chapter 5. Profiling Users Multifaceted Interests

Figure 5.7: Precision (bar) and Coverage (line) Results for Query Facets Evaluated on Users’ Indexed on their (a) Core Interests, (b) Core + Secondary Interests Regions.
Figure 5.8: Precision (bar) and Coverage (line) Results for Query Facets Evaluated on Users’ Indexed on their (c) Tertiary Interests, (d) Everything/All Interests Regions.
Chapter 5. Profiling Users Multifaceted Interests

recommending users that share a target user’s core interests ($F_1$), and based on region indexes composed of the user’s core and core+secondary interests. But user-tags are best when it comes to generating recommendations based on the user’s tertiary interests ($F_4$, $F_5$, $F_7$), using recommendation indexes composed of tertiary interests or complete profiles.

Moreover, in most of the above scenarios, where tags significantly outperform user-tweets, we also find relatively minor compromises in terms of recommendation coverage, especially when compared to the coverage compromises that were associated with tags overall, as discussed in Section 5.4.2.1. For example, for query facet $F_1$ in Figures 5.7 (a & b) and for query facets $F_4$, $F_5$, $F_7$ in Figures 5.8 (c & d) we can see that the coverage for tag-based methods is just as high as that offered by user-tweets; it is not true in the case of query facets $F_2$, $F_3$ for Figures 5.7 (a & b) where coverage falls to 10% or less rendering tag-based recommendations impractical. It is especially encouraging to see the very strong precision and coverage results for $F_4$, $F_5$, $F_7$ in Figures 5.8 (c & d). This means we have a very effective and practical approach when it comes to the challenging task of recommending social connections based on a target user’s non-core interests.

5.4.4.2 Profile Facets vs. Recommendation Indexes

We concluded from Figure 5.6 discussion in Section 5.4.3 (ii) that using different query facets as recommendation queries has a significant impact on precision and coverage, with facets $F_1$ (the user’s core shared interests) and $F_4$ (those interests shared by only friends and followers) both offering high precision and excellent coverage. But Figures 5.7 (a & b) show us that the situation is somewhat more complex because of the interaction between facets and region indexes (the $F_4$ peak is not present); similar precision
peaks are not found in Figures 5.8 (c & d) (the $F_1$ peak is not present). In particular, the best precision for $F_1$ is only achieved when we use recommendation region indexes that are based on the core and core+secondary interests of users (Figures 5.7 (a & b)). In contrast, the best precision results for $F_4$ are found when we use recommendation indexes that are based on the tertiary or complete interests of users as in Figures 5.8 (c & d) – we also see very strong precision results for $F_5$ and $F_7$ using these indexes – but no such precision peaks are found for these facets in Figures 5.7 (a & b).

This is important for a number of reasons. It helps to highlight the distinction between generating recommendations based on a user’s core interests ($F_1$) versus their peripheral interests ($F_4, F_5, F_7$) and that different recommendation strategies need to be considered when responding to these different recommendation scenarios. In addition, it is significant that the best overall precision is achieved when making recommendations based on a user’s peripheral interests ($F_4$, those interests that are uniquely shared by friends and followers but absent from the target user’s interactions) and users indexed on tertiary or everything regions.

It seems reasonable to suppose that generating recommendations with $F_1$ should have been the easier task compared to generating recommendations for $F_4$. After all we should expect $F_1$ to provide a more complete picture of a user’s core interests than $F_4$ provides of a user’s tertiary interests. But, clearly, this is not the case. In fact, we are producing more relevant recommendations from $F_4$ (tertiary interests) than from $F_1$ (core interests). We believe that the reason for this can be traced back to the origin of interests for $F_1$ and $F_4$. The former are mined from the contributions of a single user, namely the target user. But the latter are mined from the contributions of a much larger set of users, namely the many friends and followers of the target user. This larger set of users provides a much richer
source of interests than the lone target user and because $F_4$ is based on interests that are shared between friends and followers it provides a natural way to help validate the likely relevance of these interests; if both friends and followers share an interest it is more likely to be of interest to the user.

It is also clear from these more detailed results that the core and core+ secondary recommendation region indexes are playing a very different role to indexes based on tertiary interests or a complete user profile of interests, as discussed in Section 5.4.3 (iii). Recommendation indexes made up of a user’s core interests, for example, are a good way to recommend new connections based on these shared interests, achieving a precision of 20% for list-tags for example. However, and perhaps somewhat paradoxically, using the richer recommendation index composed of all user interests (Figures 5.8 (d)) actually degrades $F_1$ precision by comparison ($F_1$ precision tops out at only 15%). In contrast, recommendation indexes made up of just the tertiary interests of users or all of their interests, Figures 5.8 (c & d) respectively, deliver superior recommendation precision when it comes to recommendations based on a user’s tertiary interests ($F_4$). In this case $F_4$ achieves a maximum precision of almost 30% for user-tags in Figures 5.8 (c & d), far in excess of any other evaluation condition.

### 5.4.5 Beyond Precision and Coverage

Precision and coverage are important evaluation metrics but provide a one-dimensional way to evaluate recommendations. At the very least, our precision metric is necessarily conservative, and as such is likely to consistently underestimate true precision. In addition to these classical metrics it can be useful to examine the average similarity between recommended users and the target user (we call this the target user similarity or TUS) and also
Chapter 5. Profiling Users Multifaceted Interests

Figure 5.9: Summary Analysis of Target User Similarity (TUS) Averaged across Recommendation Indexes, Profile Facets, and Interest Sources.

the average similarity between all pairs of recommended users (we call this the inter-recommendation similarity or IRS). These metrics can be seen as the relative diversity between the users and a set of recommendation, or the diversity within that set of recommendations. In each case we evaluate these measures using a straightforward overlap metric based on profile sources (user-tweets, user-tags, list-tags).

In Figure 5.9 we show the summary results for the TUS metric, averaged across the various experimental conditions, varying recommendation index regions, query facets, and interest sources. These results show that in general the TUS scores remain relatively stable across the various conditions; on average a recommended user shares between 30% - 40% of interests with the target user. Interestingly, once again, we see strong performance for query facets $F_4$, $F_5$, $F_7$. These query facets are designed to support the
recommendation of users with complementary rather than shared interests; remember that $F_4$, $F_5$, $F_7$ are made up of interests that are not directly present within the target user’s profile but rather reflect the interests of their friends and/or followers. As such, we might expect to see the recommendation of users with lower $TUS$ scores. But, in fact, the opposite is true. Even though these recommended users are selected based on tertiary interests of the target user, they nonetheless share more interests with the target user than when we recommend users based on the target user’s own core interests (c.f. $F_1$ bar in Figure 5.9).

We can also see a larger difference between the $TUS$ scores for user-tags compared to list-tags. The former produce recommendations with the lowest average $TUS$ scores (about 28%) whereas the latter delivers recommendations with $TUS$ scores of just over 40%. Remember that, in general, as per Figure 5.6, both of these approaches are delivering similarly precise recommendations, but user-tags does this on the basis of less target user similarity than list-tags. This could be advantageous as a way to identify relevant “long tail” [Park and Tuzhilin, 2008] recommendations and may indicate that by using user-tags as our interests source we can generate more diverse recommendations than other sources of interests especially list-tags.

In Figure 5.10 we show corresponding summary results for the $IRS$ metric, once again averaged across the various experimental conditions. They show that the average similarity between recommendations is again fairly consistent across the various conditions (approximately 35% for most conditions). Once again we see that the greatest $IRS$ difference is associated with user-tags and list-tags. The former delivering recommendations with $IRS$ scores of about 28% and the latter just over 40%, confirming our hypothesis above that, in general, user-tags are capable of generating recommendations that are more diverse than other approaches.
Figure 5.10: Summary Analysis of Inter-recommendation Similarity (IRS) Averaged across Recommendation indexes, Profile Facets, and Interest Sources.

We can focus in on the IRS values for list-tags, remembering that list-tags originate from the lists users find themselves in and, as such, it’s not unimaginable that users who share a common list/s are friends/followers and have some common tags. Although, this similarity of tags doesn’t seem to adversely affect the diversity.

By exploring recommendation TUS and IRS along with precision and coverage we can now offer recommendations that cater for the range of user’s interests whilst also providing for diversity within a set of returned recommendations. Our findings show that if we want to produce highly diverse and precise recommendations, this can be achieved by utilising user-tag sources and tertiary facets and regions. In the discussion section next we will explore in more detail the take home messages from these evaluations.
5.5 Discussion

In this chapter we set out to explore a new approach to profiling users for social recommendation systems, to generate social connections based on the type of UGC and curated tags that are now commonplace on the social web. In particular, we explored three major design choices — (1) the type of interest regions to profile users; (2) different ways to generate facet queries from user profiles; (3) different sources of profile data for using in indexing — to comprehensively explore a wide range of configurations.

The preceding sections have endeavoured to explore a wide range of evaluation results and highlight some general conclusions that may improve our understanding of this type of social recommendation system and perhaps help those tasked with implementing recommendation systems in similar settings. Our key findings can be summed up as follows.

5.5.1 A Comparison of Tags and Tweets

In general, using curated tags to represent interests out-perform using user-tweets, in terms of precision. The extent of the performance improvement depends greatly on both the profile facets used as recommendation queries and the nature of the recommendation index; the best improvement is found for $F_1$ with a recommendation region index made up of core or core+secondary interests and for facet queries $F_4$, $F_5$, $F_7$ when indexed on the tertiary region or complete profile regions. Moreover, different tag types perform differently depending on the type of recommendation index used. For example the precision benefit associated with $F_1$ in Figure 5.7 (a & b) is based on list-tags, whilst user-tags offer only marginal precision benefits over user-tweets. In contrast, the precision benefits for $F_4$, $F_5$, $F_7$ in Figure
5.8 (c & d) are associated with user-tags and this time it is the list-tags that offer much less in the way of precision improvement.

### 5.5.2 Tags and Recommendation Coverage

Recommendation coverage is adversely affected by the relative sparseness of tag-based interests when compared to user-tweets. However, there are some conditions where there is little or no coverage compromise associated with the use of tags. Moreover, these conditions are precisely those that are associated with significant precision improvements, query facets $F_1$, $F_4$, $F_5$, $F_7$, as in Figures 5.7, 5.8 (a - d). This tells us that, rare as tags may be, there is high availability for certain profile facets ($F_1$, $F_4$, $F_5$, $F_7$), allowing recommendation coverage to be preserved while delivering greatly superior recommendation precision when compared to user-tweets.

### 5.5.3 Managing New User Cold-start

Although not the primary focus of this chapter a secondary benefit of this approach to modelling users comes in the form of a mechanism to tackle new user cold-start. The high-precision performing query facets ($F_1$, $F_4$, $F_5$, $F_7$) are important because they facilitate the generation of high-quality recommendations based on the target users core interests ($F_1$) as well as their peripheral interests as dictated by their friends and followers ($F_4$, $F_5$, $F_7$). This means that we can be confident when it comes to recommending new contacts that share the target user’s core interests as well as recommending new contacts that complement and possibly extend the user’s core interests. Moreover, the high precision achieved for $F_4$, $F_5$, $F_7$ provides a practical
solution to the cold-start problem that plagues many recommendation approaches. Even if the target user has not been an active Twitter user in terms of posting their own tweets (and in fact many Twitter users are not active tweeters but rather use the service to keep up with the tweets of others) as long as they have connected to friends and/or have attracted followers, then we can still be confident to make high quality recommendations. By prompting a new user to form some connections our model can utilise these connections and form recommendations for them.

5.5.4 The Influence of the Recommendation Index

We also found that the type of the recommendation region index matters. Indexing users based on their core or core+secondary interests benefits the recommendation of other users who share these interests. But adding more peripheral (tertiary interests) compromises this ability to generate such high quality recommendations; see the reduction in precision associated with $F_1$ in Figure 5.7 (a & b) compared to Figure 5.8 (c & d). Conversely, in order to generate high quality recommendations, to complement or extend the target user’s core interest, demands recommendation indexes that include tertiary user interests; see the improvement in precision associated with $F_4$, $F_5$, $F_7$ in Figures 5.8 (c & d) compared to Figures 5.7 (a & b).

5.5.5 Tags and Diversity

And finally, aside from the matters of precision and coverage, there is evidence that by representing user interests with user-tags we can generate more diverse sets of recommendations that are less similar to the target user without necessarily compromising precision. In contrast, list-tags tend
to produce less diverse collections of recommendations that are more in keeping with the interests of the target user. Depending on the type of recommendation scenario diversity may or may not be important. In our setting it makes sense to want to favour recommendations that maximally extend a user’s current network, suggesting that the diverse recommendations offered by user-tags are likely to be of interest.

5.6 Conclusions

In this chapter we set out to evaluate all of the RC 1-4. RC 1-3 were discussed in Section 5.2 as we utilised user-tweets, list-tags and user-tags as our sources of recommendations knowledge when profiling users. In this chapter again we found that these sources, whilst potentially noisy or indeed freeform, can provide a strong signal of the interests of users on Twitter. We also revisited the social connections users have and by again incorporating these connections and their contributions, via tweets or tags, into our new profiling technique, we were able to boost recommendation quality. Indeed, identified interests of the user’s friends and followers, that weren’t associated with the user directly, proved to be the best approach to recommending friends, with precision scores of nearly 30%.

Finally, we evaluated our last remaining RC 4 in Section 5.4 by introducing a particularly novel contribution which has been to move beyond the traditional monolithic view of user profiling to consider how user interests can be captured by different groupings of interests that naturally occur within the social graphs of users. We have focused in on how to select an appropriate recommendation strategy for different recommendation tasks (e.g.
precision, cold-start, diversity, etc.). Overall, we have developed a recommendation model that can satisfy traditional recommendation metrics, such as precision and coverage, as well as diversity.

In the next Chapter 6 we will revisit the lessons we have learned from these evaluations and our previous evaluations in Chapters 3 & 4 to give an opportunity for a full discussion of the core outcomes of this thesis.
6

Conclusions

6.1 Introduction

“We are drowning in information but starved for knowledge.”

– J. Naisbitt, Megatrends. 1982

The real-time web (RTW) is truly the greatest by-product of the information age. With news, communications, and interactions happening across the world, all at lightning speed, it is no wonder that from a technology point of view there is a clear focus on mechanisms to harness this information and gain knowledge [Chen et al., 2010, De Francisci Morales et al., 2012, Garcia Esparza et al., 2010, Phelan et al., 2009]. Throughout this thesis we have explored the topic of information overload as a friend finding problem. When part of a social network, how does one best disseminate this information across millions of users? Moreover, how does one find the information of interest to them that facilitates their information needs? We
believe that connecting to the right users from which to consume information is key to solving this problem. These connections can act as information filters, sieving through the thousands of updates, posts, likes, etc. to deliver the content that is most important to you, the user.

Information dissemination via valuable connections allows for users to keep up to date and informed on the interests that matter to them. The core contributions of this thesis are motivated by the availability of RTW data and the potential use of this unique form of user generated content (UGC) in the form of tweets, tags, etc. to drive recommendation systems. We believed that this potential source of recommendation knowledge contained latent potential when applied to the task of modelling users and recommending friends. Throughout the evaluations carried out in this thesis our initial insights discussed in Section 1.3 (RC 1-4) have held true. Indeed, the content produced by users on social networking sites such as Twitter does provide insights into the interests of users. Moreover the social connections these users make also lend themselves to modelling the interests that they want to consume from others. At the beginning of this thesis we outlined some contributions which guided our research. Below we reiterate these 4 research contributions and in the following sections we will discuss these contributions in the context of this thesis.

**Contribution 1**

Can we develop an effective friend recommendation system that is guided by user profiles, which are based on what users tweet about and that allows us to recommend useful users to follow? (Discussed in Section 3.2.1 and evaluated in Sections 3.4 & 4.4)
Contribution 2

Can we exploit the structure of the social graph to aid recommendation, by looking beyond a user’s own tweets. (Discussed in Sections 3.2.2 & 5.2 and evaluated in Sections 3.4, 4.4 & 5.4)

Contribution 3

What other sources of content can be harnessed? Can we transition beyond noisy tweet terms to tag-based profiles and in turn utilise these profiles for friend recommendations? (Discussed in Section 4.2 and evaluated in Sections 4.4 & 5.4)

Contribution 4

How can we accommodate the complex and diverse interests of users during friend recommendation to ensure that we can generate recommendations that reflect these varied interests, rather than recommendations that are dominated by some single core interest? (Discussed in Section 5.2 and evaluated in Section 5.4)

In the following sections, and in part, as way of a concluding summary we will focus in on each of these research contributions. We will discuss each contribution in detail by exploring what we have achieved. Then we will move on to discuss some of the identified limitations with this research and some potential future works that would be interesting to pursue but which lie outside of the scope of this thesis. We conclude this chapter with some closing remarks and reiterate the take home message from this piece of work.
6.2 Contribution 1 - Tweet-based Profiling

The first research contribution of this work relates to RC 1, Section 1.3. We proposed that we can harness tweets as a source of profiling information when forming friend recommendation. In Section 3.2.1, we outline our approach to profiling a user based on their tweets. We describe the transition from tweets to profiles, for each user let $tweets(U_T)$ represent the set of tweets for a given user. These tweets a user produces represent the interests of the user, we base our profiling technique on the basic assumption that what you talk about is what interests you. With this profile we could then start to focus on recommending users based on their shared sets of interests.

In Section 3.3 we introduce our friend recommendation framework which utilises tweets as our source of profiling information. We utilised an IR-based approach to building our framework where we structured users as documents and their profiles as queries. By utilising TF-IDF similarity we could search out and recommend user who shared similar interests to our user. By utilising the tweets of the user alone we showed that we could recommend friends effectively to the user using our stringent precision metric. Across the two evaluations in Sections 3.4 and 4.4 precision scores of between 11-13% could be achieved for this profiling technique with recommendation lists of size 20. Another outcome of this evaluation showed that coupled with precision relevant recommendations appeared closer to the top of recommendation lists, which bodes well for a live system as users spend the majority of their time looking on the top of lists [Silverstein et al., 1999].
6.3 Contribution 2 - Harnessing Social Graph

Our second contribution builds upon the tweet-based profiling technique identified in RC 1. In Section 3.2.2 we considered how to harness the social graph for profiling and recommendation as part of RC 2. By focusing in on the connections a user has namely, their friends: users who are followed by out target user, and their followers: users who follow our target user. We wanted to evaluate the merits of modelling users based on those tweets contributed from their social connections. We formed two new profiling approaches which harnessed these social connections, namely profiling the user based on $friendstweets(U_T)$, $followerstweet(U_T)$.

We utilise these techniques by incorporating each approach into our recommendation framework and evaluated these approaches on our large scale dataset of live Twitter users. In Sections 3.4 and 4.4, our two socially driven tweet-based evaluations improved upon the accuracy of the user only driven content strategy. The friend and follower-based approaches could produce precision scores ranging from 11-14% and have average relevant recommendations positions of 6, compared to 7 for the user only approach. Interestingly, we found that when it comes to modelling the user, the content produced by the user is surplus to demand as their friends and followers are as useful, or more so, to represent our user’s interests. We find this to be case also in our other evaluation in Section 4.4.

6.4 Contribution 3 - From Tweets To Tags

Chapter 4, discusses our third contribution in this work our aim here was to again build upon the work we have carried out and the contributions we have
already achieved. Our contribution here explored two new complementary sources of user interest information. In Section 4.2 our sources come in the form of tags, but each tag-based approach made quite different assumptions about the link between a tag and a user. Researchers have shown that tag-based approaches can perform effectively as a source of profile information [Milicevic et al., 2010, Zanardi and Capra, 2008] and we wanted to evaluate if this was the case for our friend recommendation framework. First off we focused on user-tags, which were gathered from WeFollow, a social tagging service where users define their own interests via tags. The second source discussed was list-tags from Listorious, where tags were assigned to lists of users and then assigned to the users therein.

We evaluated these new sources in Section 4.4 by adding them as profiling knowledge to our identified strategies within our framework. We collected a large dataset of over 20,000 users who were present across the three services Twitter, WeFollow, and Listorious, discussed in Section 4.4.1. By developing 21 strategies based on individual and combinations of strategies, again we evaluated the effectiveness of our approaches at being able to recommend friends a user had already been following. The two new tag-based sources did indeed provide for better modelling sources for user’s interests. With list-tags, user-tags, and user-tweets, ranked in that order respectively, performing the best with our best strategy (users modelled on their friends, followers and own content) producing precision scores of 21%. Our framework could now produce precision scores in excess of 20% when requesting 20 recommendations, with list-tags proving themselves as the most effective profiling source. This was a 9% increase over our user only tweet-based approaches identified in Contribution 1, Section 6.2. Although for one of our tag-based approaches, user-tags, a precision increase came with a hit in coverage compared to user-tweets. This downside provides
scope for a fall back to tweet-based approaches when the user-tag based source can’t profile and form recommendations effectively.

6.5 Contribution 4 - Profiling Users Multi-faceted Interests

The final contribution of this work is discussed in Chapter 5 and focuses on the way in which users are currently modelled by recommendation systems. Up until now traditional approaches looked at the profile of a user in a one dimensional way, each user is modelled in essence in a monolithic form. By only focusing in on the user, their friends, and their followers as single entities. This monolithic view of the user is far less representative of the diverse collection of interests users can have, moreover the relative strength of those interests core, peripheral, etc., is lost when we treat them all as equally important. We introduce a new approach to modelling users on social networks in Section 5.2. By faceting the current interests profile we’ve used in Chapter 3 & 4 we can identify the sets of core, secondary, and tertiary interests of a user.

In Section 5.4 we discuss our evaluation to test the effectiveness of this model; again, using the large dataset of over 20,000 users mentioned in Section 4.4.1. Some 84 different combinations of recommendation strategies are evaluated by combining facets, regions and sources discussed in Section 5.4.1. Across the spectrum of evaluations the tertiary interests focused approaches provide for the most accurate recommendations for a user, again these interests are from the user’s friends/followers and share no overlap with the user’s set of identified interests. Our best performing recommendation strategy produced precision scores of 28% and coverage scores of 100%, this
was seen on facet $F_4$ the user’s friend and followers, with user-tags being used as a recommendation profile query and the recommendable users being indexed on their tertiary or all interests regions. Two noteworthy points can be gained from this insight; again as with prior contributions it’s clear that it’s time to look beyond the user solely when modelling and forming recommendations and start harnessing the user’s social graph as a source of recommendation knowledge and secondly by faceting a user’s profile we can build different recommendation strategies for different user intents, e.g. find users who share my core interests. The multifaceted model also produces increased recommendation precision over our traditional monolithic approaches, with highs of 28% an 8% increase between the best performers of each.

6.6 Limitations

As always, there are many limitations that must be acknowledged in order to temper the conclusions that can be drawn from this work. In the following sections we will explore these in more detail and how they impact on the evaluations we have carried out.

6.6.1 Contrasting With Other Approaches

Recommendation systems have abundant research focusing on utilising alternative approaches to forming connections between users, from location-based [Li et al., 2008, Zheng et al., 2009, 2011] to graph-based [Lo and Lin, 2006, Perugini et al., 2004], etc. One of the most widely utilised approaches is graph-based analysis which aims to close in missing links within
a user’s social graph. This approach is limited though to suggesting possible friends’ of friends or potentially known acquaintances. Indeed, even the location-based approaches are focused more towards prior friendships or acquaintances than recommending similar users based only on their identified interests. Closing off the user’s social graph is not the focus of this thesis. Whilst wanting to generate an increase in the number of users our test user would follow, our focus is on finding users who originate from any part of the social network but, who also share a specific set of shared interest with our target user.

Examining other friend recommendation research is often useful to contrast ones approaches with the current state of the art in the field. Chen et al. [2012] published a winning paper from the 2012 KDD cup focusing on building friend connections. The setting in this case was Weibo, a Twitter like service which is also built around the idea of sharing short micro-blog messages between users. Their approach used adaptive forests and matrix factorisation to firstly understand user be friending behaviour and secondly utilise this insight to gauge whether they would follow other users or not. Their approach utilised user features such as age, gender and peripheral information like temporal following patterns. Their best in class approach achieved a MAP @ 3 score of 43% (roughly 1.5 users out of 3). Contrasting with our approaches we can see that both are very different ways to tackle the same problem. The closest comparison that we could potential make is from our first analysis Figure 3.5, taking k=5 recommendations, we achieve nearly 30% precision, although it’s fair to say we’re not comparing like for like here as [Chen et al., 2012] didn’t focus on the content of the posts as we have and the list sizes are different. Nevertheless it’s encouraging to see that different approaches to similar problems have comparable results.
6.6.2 Evaluation Reinforcement

The second limitation is the use of quantitative over qualitative measures to evaluate our approaches. Whilst we were able to carry out a preliminary live user trial to evaluate Twitter tweets as a profiling source, in Section 3.4.4, doing the same for the tag and multifaceted model approaches wouldn’t have been as straight-forward and as such this type of evaluation wasn’t part of our testing criteria for our evaluations in Sections 4.4 & 5.4.

To be evaluated by our system users must contribute by posting tweets, forming connections and be generally a frequent user of Twitter. When we incorporate third party sources to model the users such as tags from WeFollow and Listorious, we add an extra dimension of needs from our test users. These websites do not contain all Twitter users and, as such, the amount of users tagged or contained within lists represents a much smaller subset of Twitter users. Taking this into account, and to perform a like for like comparison between approaches, we could not guarantee that users who participated in the trial would be present across the three of these information sources. But, by not evaluating qualitative metrics this does not remove from the findings from our evaluations and indeed these results can provide insight in scenarios where tweets/tags are available, into the appropriate recommendation source and type that should be selected.

6.6.3 Dataset Scale and Sampling

Our dataset is to just over 20,000 users and just over 400,000 friends and followers connections. While certainly not insignificant, it is far from the scale of today’s larger social networks which count users in the hundreds of millions. Nevertheless, we feel that the scale of this study is sufficient to
draw at least some initial conclusions about the relative merits of certain design decisions for friend recommendation as we have done. Also to our knowledge this dataset represents one of the largest cross web service user profile collection of data, when Twitter, Listorious, and WeFollow sources along with the user’s social connections are counted.

On the matter of users, we must also acknowledge a potential bias in the sampling of our evaluation datasets. It is likely that these user datasets are representative of the more active (Twitter) users. In building our datasets we ensured that all users were represented on Listorious and WeFollow. We did this to ensure that we could represent users not just by user-tweets but also by list-tags and user-tags. The guaranteed availability of tags facilitated a direct like-for-like comparison. Nevertheless we must acknowledge that the average Twitter user is unlikely to be included in Listorious or WeFollow and it is much more likely that this type of coverage is only reasonable to expect from the more active Twitter users. As such, we must be careful to interpret any results as they relate to more active users. This does not undermine the results that we have presented, but rather frames their applicability, at least until further work is conducted on a broader sample of users.

6.7 Future Work

Throughout this work we have maintained a constant focus on the field and relevant research methodologies from other researchers. As with any substantial piece of work, there are always other secondary lines of enquiry that could have been added, but which fell outside of the primary focus of this thesis. As ongoing future work we have identified four such research contributions that provide interesting complementary research to the work presented here.
6.7.1 Transitioning To Other Contexts

We believe some exploratory research could be carried out into friend recommendation in different recommendation contexts. Although, much research into social networks are carried out via Twitter because it’s highly indicative of many other social networks and has an open API to gather information about users. It would be interesting to explore different sites which still require valid connections to be formed but, for different reasons e.g. LinkedIn to form business connections, Facebook where friends are more likely to be real life friends, as opposed to information filters in the Twitter context. These two identified contexts for connection generation would be interesting to evaluate, although their information is often proprietary and harder to gain access to.

6.7.2 Evaluating Cold-start

We would like to explore complementary research which focuses on the cold-start problem, specifically new user cold-start [Schein et al., 2002]. In this scenario, new users who join Twitter, for instance, will have no tweets or connections from which we could elicit recommendation information. Without recommendation information to form profiles, personalised recommendation cannot be formed. In Appendix A, we briefly discuss allowing users to explicitly enter interests to find users. But, as part of future work, we would explore different mechanisms to garner user information at sign up or through implicit actions whilst they are using the website [Claypool et al., 2001] and then compare these approaches with our prior evaluations.
6.7.3 Towards Improving Precision Metrics

Earlier in Section 3.4.4 we highlighted the conservative nature of our friend-based precision metric; a recommendation is only considered relevant if the target user is already connected to them. Clearly it does not allow for a good recommendation who is not yet followed by the target user. This is a common bias in conventional recommendation systems evaluation methodologies. The concern is that this may skew the performance results in a particular direction and not provide a true insight into practical performance in the field. To address this, we believe that other approaches should be evaluated. We took initial steps into exploring other metrics such as using a friend of a friend-based approach; see Appendix B. But, we believe that this approach could be further expanded with accompanying testing to evaluate its merits.

6.7.4 Utilising Alternative Approaches

The last piece of future work we would like to evaluate is the further development of our model to incorporate other approaches to recommendation. One interesting avenue which could be explored is the temporal windowing of interests and the classification of users prior to modelling. Research by Garcia Esparza et al. [2013] has looked at classifying the tweets of a user, but if these individual classifications of the user’s tweets could be utilised as weighted tag interests we could start to recommend others based on these weighted similarities between users. Also by utilising the temporal features of tweets [Cataldi et al., 2010] and their timestamps one could identify what might have been a keen interest a few months ago, that may be a passing interest now and as such the user’s profile should reflect these changes in
the people being recommended. As always other approaches to recommendation could be incorporated such as collaborative-filtering or graph-based to re-rank users who are of interest or other techniques could be incorporated into the content sourcing stage through forming ontologies [Cantador et al., 2008] of interests from content associated with users. All of these approaches could be additional add-ons to a friend recommendation system but, with our approach we have shown that content linked with the user can be harnessed to form effective recommendations on its own and as such evaluating each of these add-on poses different research questions that are outside of the scope of this thesis.

6.8 Closing Comments

In this thesis we have showcased our recommendation framework that can harness RTW sources such as tweets and tags to profile the interests of users on social networking sites such as Twitter. Our recommendation framework can produce precision scores in excess of 20% when recommending friends a user is already following. The battery of evaluations carried out justified our initial choices for research contributions and in all cases we were able to provide answers to these research problems. We have also discussed some limitations that could be associated with this research and in all cases we have discussed these limitations and how we believe they could potentially impact our research.

In conclusion in this thesis we have shown that a) the RTW is an effective source of profiling information for users, b) users social connections can add extra insight into the interests of users, c) the traditional user profile needed to be re-evaluated and we showcased an approach that facets the
user into their core, secondary and tertiary interests all of which represent different needs of the user. By gathering a dataset of live users on Twitter we carried out extensive evaluations which measured the reward that our system could provide to the user either by the quality or indeed diversity of recommendations returned to users. Since the beginning of this PhD in 2009, the RTW has grown at an enormous rate and the need we identified then for mechanisms to aid with forming connections is now even more essential. We believe that the research herein provides just such a solution to this problem and adds benefits to both modelling users on the RTW and to friend recommendation research.
Appendix A

Evaluating New User Cold-start

A.1 Introduction

In this Appendix we showcase a potential partial solution to new user cold-start which happens when users initially join Twitter. Our approach requests the user to actively search out users to follow by entering search queries instead of forming recommendations based on their user profile. Our friend recommendation system which comprises of users indexed by their previous tweets can be searched over to find users who talk about the interests a new user is looking for. Once these new users start to form connections and contribute content the system can revert to using the users’ profiles as the search queries and recommendations can be formed based on the techniques outlined throughout this thesis.
A.2 Evaluation

To test the performance of our framework at helping cold-start for new Twitter users we carried out a second live user evaluation of the search functionality using the Twittomender web application built on top of our recommendation framework. We invited 80 participants to carry out up to 5 or more searches for Twitter users they would be interested in following. These participants were all first year students from a computer applications course in Dublin City University. The break down of the 80 participants was 16 females and 64 males with the majority of participants aged 18-22. As we were evaluating the ability to aid new users form connections the test group of users was formed of all relatively new adopters of Twitter, with many only recently joining. Table A.1 shows a summary of their user statistics. We can see that on average participants of the trial had produced 12 tweets, followed 20 Twitter users and were followed by about 8 users. This would be considered relatively low and indicative of new adopters. To bridge the gap between being a new user and an active user, some help would be needed.

Table A.1: Summary Statistics for Cold-start User Trial Participants.

<table>
<thead>
<tr>
<th>No. Of Tweets</th>
<th>No. Of Followers</th>
<th>No. Of Friends</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.75</td>
<td>7.875</td>
<td>19.99</td>
</tr>
</tbody>
</table>

A.2.1 Search Relevance: MAP

In order to test the relevance of our returned results, we used the mean average precision (MAP) measurement. We looked at the results across all participants searches and averaged them across all of our participants. When presented with a results list users voted on whether or not a result
was relevant to their search query and whether they would follow those suggested users. In Figure A.1 we can see on average across all of the searches, participants would choose to follow 20% of the returned suggestions. On average this is 5 new users that each participant would follow per search. Also in this graph users indicated that 45% of the results were deemed relevant to the query that was entered. So, by using the search functionality of our framework users could find new, previously undiscovered and relevant Twitter users to follow. Precision is only one factor when evaluating a system. We also evaluated the position at which of relevant results appeared.
A.2.2 Relevance Positions

In Figure A.2 we have graphed the selections made by the participants in the trial and their position within returned user suggestions lists. In Figure A.2 we can see, of all the relevant suggestions that users would follow on Twitter, that 50+% of results appear in the top 10 users of the returned lists. And, we can see in Figure A.2 that likewise, when we examine just the relevant results (those results that a participant deems relevant to the query regardless of if they’d follow them), we again see that over half are contained within the top 10 of the returned lists.

Figure A.2: Position of User Searches Indicating Total Relevant Results from Relevant and Would Follow and Relevant and Wouldn’t Follow.
A.3 Discussion & Conclusions

This live user evaluation on 80 new or recently joined participants on Twitter showed an effective mechanism for users to search out others to follow. Each participant carried out a set of user searches and on average across all their searches they found 45% were relevant, that they would follow 20% of those suggested and those relevant tended to appear towards the top of suggestion lists. The next progression we would see for new users after connecting with relevant users through our search system would be to now harness their improved user profiles and form recommendations for them as we have discussed in Chapters 3 - 5.
Appendix B

Evaluating Friend Recommendation Metrics

B.1 Introduction

Earlier we highlighted the conservative nature of our friend-based precision technique. Clearly it does not allow for the obvious possibility of a good recommendation who is not yet followed by the target user. To address this, in this section we would like to propose an alternative precision metric that provides for a more fine-grained measure of precision and one that can be adjusted in terms of its inherent strictness.
Appendix B. Evaluating Friend Recommendation Metrics

### B.2 Friend-of-a-friend Precision

Many friend-based recommenders harness graph-based degrees of separation techniques when suggesting users to follow, e.g., your friend’s friend potentially could be your friend. We plan to utilise this graph-based approach but rather than aid in the recommendation stage we will use it to identify potentially relevant users after recommendations are formed. The underlining theory behind this approach is that the friends you have chosen to follow have found these other users useful and added them to their social graph, much as you would do and as such they maybe useful (Equation B.1).

\[
FoF(U) = \bigcup_{\forall f_i \in \text{friends}(U)} (\text{friends}(f_i)) \quad (B.1)
\]

The friend of a friend (FoF) precision metric for each of our test users comprises of two parts. Firstly we explore a test user’s friends’ friends lists removing any friend targets that are already friends of that test user and add these users to a FoF list (see Figure B.1). That FoF list is populated until it contains 5000 unique FoF, then this list is ranked based on the co-occurrence of a friend across these friends lists, this ranked list is used as
the test user’s FoF’s target list instead of their own friends connections list. The 5000 FoF limit was imposed as we didn’t want to add an undesired bias to this expansion. For example if we take the several degrees of separation principal, by continuously expanding FoF we would end up with a list containing potentially all known users. Then the task of calculating precision scores becomes defunct as any suggestion will be a member of the FoF list.

Those precision scores shown in Figures B.2 & B.3 (a to d) are calculated by using the recommendations produced from our previous experiments discussed in Section 5.4 but now we determine the quality of a recommendation list using a test user’s FoF target list. Figures B.2 & B.3 (a to d) shows the precision scores for each strategy run over each of the four indexes of core, core+secondary, tertiary and everything profile interests.

Figure B.2 (a & b) has precision scores peaking at a max of 30% with the best performing strategy $F_4$ tags (friends and followers query), when users are profiled on their core winning out. When we explore this result list-tags are nearly double that of any other strategy. This is interesting because this results is not seen in our traditional approach in Figure 5.7 (a & b). Perhaps this points to there being latent potential for list-tag approaches at recommending friends using facet $F_4$ and user indexed on their core interests. Potentially this approach does identify relevant users but those users are currently not being followed by our test user. Neither of these conclusions can be affirmed but, this result does warrant further analysis.

Figure B.3 (c & d) almost mirror the precision seen in Figure 5.8 (c and d), again the tertiary facets $F_4$, $F_5$ and $F_7$ perform the best, this further backs up the utility in not only focusing on the user, but also their extended social graph for recommendation information. Users profiled on user-tags and facet $F_4$ return precision in excess of 20%, remember again these are
Figure B.2: Friend of a Friend Precision Results for Query Facets Evaluated on Users’ Indexed on their (a) Core Interests, (b) Core + Secondary Interests Regions.
**Figure B.3:** Friend of a Friend Precision Results for Query Facets Evaluated on Users’ Indexed on their (c) Tertiary Interests, (d) Everything/All Interests Regions.
not currently friends of our test user, but of their friends. Although Figure B.2 has lower precision scores than its Figure 5.8 (c & d) counterpart.

From a friend finding standpoint the results shown in Figures B.2 & B.3 are encouraging, the correctly identified targets represent users within one degree of separation to our test users. Expanding one’s social graph is done in most cases very carefully, choosing a user which a current friend has validated aids in the filtering process when selecting users to follow. In most cases the first place to select new friends will be via your current friend’s connections or discovery will be through Retweets of a friend’s friend content. These results bode well for using FoF to identify latent precision potential within a recommendation list, coupling FoF with traditional friend precision could increase relevant recommendation precision to over 50% but, this can’t be fully verified without a live user trial and as such will become part of future work.

**B.3 Discussion & Conclusions**

In summary, this alternative precision metric has revealed some interesting performance differences that were not visible using a more conservative precision metric. Of course this does not mean that the new metric is measuring something useful – it could be that we are no more likely to follow FoF users than any randomly chosen user, but this seems unlikely – and certainly further research is necessary to compare this and our more conservative precision metric to a gold standard such as a live user trial.
Bibliography


Bibliography


**Web search and data mining**, WSDM ’12, New York, NY, USA, pp. 153–162. ACM.


Bibliography


Guy, I., I. Ronen, and E. Wilcox (2009). Do you know?: recommending people to invite into your social network. In IUI ’09: Proceedings of the 13th international conference on Intelligent user interfaces, New York, NY, USA, pp. 77–86. ACM.


Signorini, A., A. M. Segre, and P. M. Polgreen (2011). The use of twitter to track levels of disease activity and public concern in the us during the influenza a h1n1 pandemic. PloS one 6(5), e19467.


