Using Recommendations in Content Curation

Zurina Saaya, MSc

A thesis is submitted to University College Dublin in fulfilment of the requirements for the degree of Doctor of Philosophy in

School of Computer Science and Informatics
College of Science

Head of School: Mr. John Dunnion
Principal Supervisor: Prof. Barry Smyth
Mentor: Dr. Rachael Rafter

October 2014
Contents

List of Figures v
List of Tables viii
Declaration of Authorship ix
Abstract xi
Publications xiii
Acknowledgements xiv
Abbreviations xv

1 Introduction 1
1.1 Towards the Curated Web 2
1.1.1 From Curation 1.0 to the Age of Search 3
1.1.2 Curation 2.0 9
1.1.3 Beyond Curation 2.0 12
1.2 Thesis Motivations 12
1.2.1 Using Curation as Platform for Recommender System 13
1.2.2 Collection Discovery 16
1.3 Thesis Contributions 17
1.3.1 New Setting for Recommender Systems 18
1.3.2 Intrinsic and Extrinsic Profile 19
1.3.3 Recommending Collections of Items 20
1.3.4 Analysis on Live-user Data ........................................ 21
1.4 Thesis Outline .................................................. 21

2 Background Research .............................................. 23

2.1 Defining Content Curation ........................................ 24
   2.1.1 A Model of Content Curation .................................. 25
   2.1.2 Examples of Content Curation ................................ 27

2.2 Content Curation Challenges .................................... 32
   2.2.1 Content Classification and Clustering ...................... 33
   2.2.2 Labelling and Tagging .................................... 35
   2.2.3 Information Discovery .................................... 36

2.3 Curation Case Studies ........................................... 37
   2.3.1 Case Study I: Scoop.it ....................................... 38
   2.3.2 Case Study II: HeyStaks ..................................... 43
   2.3.3 Summary .................................................. 50

2.4 A Review of Recommender Systems ............................... 50
   2.4.1 Item Information ........................................... 53
   2.4.2 User Information .......................................... 53
   2.4.3 Transactional Information .................................. 54
   2.4.4 Collaborative Recommender Systems ....................... 54
   2.4.5 Content-based Recommender Systems ....................... 58
   2.4.6 Recommending Collections of Items ......................... 65

2.5 Conclusions ..................................................... 66

3 Using Recommendation for Collection Assignment ............... 68

3.1 Introduction .................................................... 68

3.2 Related Work .................................................. 71
   3.2.1 Profile and Item Representation ............................. 71
   3.2.2 External Knowledge for Textual Features .................. 73
   3.2.3 Tag Recommendation ....................................... 75

3.3 The Assignment Task in Scoop.it ................................ 77
   3.3.1 Data Collection ........................................... 78

3.4 Defining the Assignment Task in Scoop.it ....................... 79
   3.4.1 Representing Collections ................................... 81
   3.4.2 Intrinsic Features .......................................... 83
   3.4.3 Extrinsic Features ......................................... 85
   3.4.4 A Term-based Recommendation Approach .................. 86
   3.4.5 A Classification Based Recommendation Approach .......... 89

3.5 Evaluation ..................................................... 91
3.5.1 Experiment Setup ........................................ 92
3.5.2 Test Conditions .......................................... 93
3.5.3 Accuracy by Test Conditions .......................... 94
3.5.4 Accuracy by Collection Maturity ...................... 97
3.5.5 Accuracy by Collection Count .......................... 99
3.6 Conclusions ............................................... 101

4 Using Recommendation for Content Discovery ........................................... 103
4.1 Introduction ................................................. 103
4.2 The Collection Discovery Challenge ....................... 104
4.3 Related Work .............................................. 107
4.4 Representing Collections ..................................... 110
    4.4.1 Basic Collection Representation ....................... 114
    4.4.2 Latent Semantic Indexing (LSI) Approach ........... 116
    4.4.3 Latent Dirichlet Allocation (LDA) Approach ........ 119
4.5 Recommending Collections ................................... 124
4.6 Evaluation .................................................. 126
    4.6.1 Experiment Setup ....................................... 127
    4.6.2 Recommendation Accuracy ............................. 129
    4.6.3 Collection Novelty ...................................... 132
4.7 Conclusion .................................................. 135

5 Recommending Search Experiences ....................................................... 137
5.1 Introduction ................................................. 137
5.2 The Assignment Task in HeyStaks .......................... 139
5.3 A Recommendation Approach to Stak Selection .............. 143
    5.3.1 Representing Search Context ........................... 143
    5.3.2 Recommending Staks .................................... 145
    5.3.3 Term-based approach .................................... 147
    5.3.4 Classification-based Approach ......................... 150
5.4 Evaluation I: Singleton Sessions ........................... 152
    5.4.1 Data Set ............................................... 152
    5.4.2 Success Rates by Recommendation Strategy .......... 154
    5.4.3 Success Rates by Stak Size ............................ 156
5.5 Extended Sessions ........................................... 157
    5.5.1 Harnessing Extended Sessions .......................... 159
    5.5.2 Session-based Term Weighting ......................... 161
5.6 Evaluation II: Extended Sessions Data ....................... 163
    5.6.1 Data Set ............................................... 163
5.6.2 Session Analysis ........................................ 165
5.6.3 Success Rate vs. Session Length ......................... 167
5.6.4 Improvement by Term Weighting ......................... 169
5.6.5 Influence of Stak Size .................................... 170
5.7 Conclusions .............................................. 172

6 Conclusions .................................................. 174
  6.1 Contributions, Implications and Limitations ............... 175
    6.1.1 Recommender Systems in Content Curation ............. 175
    6.1.2 Profiles and Item Information Enrichment .............. 178
    6.1.3 Representing Collections of Items ..................... 179
  6.2 Future Work ........................................... 180
    6.2.1 Maintaining the Collection .......................... 180
    6.2.2 Collaborative Curation ............................... 180
    6.2.3 Temporal Analysis of User Interest .................... 181
    6.2.4 Mining Domain Expertise .............................. 181
    6.2.5 Content Curation in Other Applications ............... 182
  6.3 Closing Comments ....................................... 183

Appendices .................................................... 184
  A Digital Curation Centre (DCC) Model ..................... 185

Bibliography .................................................. 190
List of Figures

1.1 Yahoo Directory in 1994 .............................................. 3
1.2 Term-document matrix ............................................... 5
1.3 A simplified illustration of PageRank score calculation .......... 7
1.4 Google +1 in search result ......................................... 9
1.5 Bing with Facebook integration .................................. 9
1.6 The early version of Delicious (formerly known as Del.icio.us) main interface .................................. 10
1.7 BagTheWeb main interface ......................................... 11
1.8 Example of an off-topic page ....................................... 15
1.9 The Pinterest boards ................................................ 17
1.10 Analogy Diagram ................................................... 19

2.1 The bag in BagTheWeb ............................................... 29
2.2 Story in Storify ....................................................... 30
2.3 Pinterest board content ............................................. 31
2.4 Example of recommended tags in Delicious ....................... 36
2.5 Scoop.it interface ..................................................... 39
2.6 Scoop.it topic page ................................................... 41
2.7 Scoop.it suggested content ......................................... 42
2.8 Add new page ........................................................ 42
2.9 Scoop.it topics dashboard .......................................... 43
2.10 The HeyStaks system architecture and outline recommendation model ......................................... 45
2.11 Heystaks browser plugin in Mozilla Firefox ....................... 46
2.12 HeyStaks in action ................................................... 47
2.13 Example of HeyStaks recommendations ........................... 49
2.14 An example of Amazon recommendations .......................... 52
2.15 Example of Amazon rating .......................................... 55
2.16 User-based collaborative filtering .................................. 56
2.17 Item-based collaborative filtering .................................. 56
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.18</td>
<td>The basic form of case-based recommendation</td>
</tr>
<tr>
<td>2.19</td>
<td>Two different similarity profiles</td>
</tr>
<tr>
<td>2.20</td>
<td>Unstructured content</td>
</tr>
<tr>
<td>2.21</td>
<td>Term-document matrix</td>
</tr>
<tr>
<td>2.22</td>
<td>Recommending a collection of items in Amazon</td>
</tr>
<tr>
<td>3.1</td>
<td>Assigning content to a collection in Storify</td>
</tr>
<tr>
<td>3.2</td>
<td>Flow of information in AutoTag</td>
</tr>
<tr>
<td>3.3</td>
<td>Example of an off-topic page</td>
</tr>
<tr>
<td>3.4</td>
<td>Scoop.it database schema</td>
</tr>
<tr>
<td>3.5</td>
<td>Sample Scoop.it data for user, collection and page</td>
</tr>
<tr>
<td>3.6</td>
<td>Summary dataset</td>
</tr>
<tr>
<td>3.7</td>
<td>System diagram</td>
</tr>
<tr>
<td>3.8</td>
<td>Sample BOW representation for a page</td>
</tr>
<tr>
<td>3.9</td>
<td>Result-list from Scoop.it search engine</td>
</tr>
<tr>
<td>3.10</td>
<td>Sample intrinsic and extrinsic information</td>
</tr>
<tr>
<td>3.11</td>
<td>Collection summary index</td>
</tr>
<tr>
<td>3.12</td>
<td>Sample training instances in classification-based approach</td>
</tr>
<tr>
<td>3.13</td>
<td>Accuracy of collection recommendations for different test conditions (I-I, X-I, I-X and X-X)</td>
</tr>
<tr>
<td>3.14</td>
<td>Average accuracy of each combinations of training and test; $k = 1$ and $k = 3$</td>
</tr>
<tr>
<td>3.15</td>
<td>Sizes of Training Set</td>
</tr>
<tr>
<td>3.16</td>
<td>Average accuracy for different collection size when $k = 3$</td>
</tr>
<tr>
<td>3.17</td>
<td>Accuracy vs. number of collections per user when $k = 3$</td>
</tr>
<tr>
<td>4.1</td>
<td>Screenshot of Scoop.it emphHome dashboard</td>
</tr>
<tr>
<td>4.2</td>
<td>Conceptual Model for URL recommendations in Twitter</td>
</tr>
<tr>
<td>4.3</td>
<td>Conceptual model for collection recommendations</td>
</tr>
<tr>
<td>4.4</td>
<td>Collection information at different levels of granularity</td>
</tr>
<tr>
<td>4.5</td>
<td>Collection basic information</td>
</tr>
<tr>
<td>4.6</td>
<td>Basic collection representation</td>
</tr>
<tr>
<td>4.7</td>
<td>Singular Value Decomposition</td>
</tr>
<tr>
<td>4.8</td>
<td>LSI sample output</td>
</tr>
<tr>
<td>4.9</td>
<td>Top-20 terms per topic and its probability score</td>
</tr>
<tr>
<td>4.10</td>
<td>Topic distribution for each document in the dataset and sample LDA output</td>
</tr>
<tr>
<td>4.11</td>
<td>Collection index</td>
</tr>
<tr>
<td>4.12</td>
<td>User representation</td>
</tr>
</tbody>
</table>
List of Figures

4.13 Summary recommendation combination .......................... 127
4.14 Distribution of number of followed collections by users ...... 128
4.15 Average precision for different user profile ....................... 130
4.16 Average precision for different collection indexes ............... 132
4.17 Average precision for different profile configurations .......... 132
4.18 Average novelty score for different user profiles ............... 134
4.19 Average precision and novelty score for different types of collection index ........................................... 135
4.20 Average precision and novelty score for different types of user profiles ........................................... 135

5.1 The stak in HeyStaks .................................................. 141
5.2 Sample stak page ...................................................... 142
5.3 Sample result list ...................................................... 146
5.4 Staks Recommendation .............................................. 147
5.5 Stak Summary Index .................................................. 148
5.6 Training instances for classification-based approach .......... 151
5.7 Summary user data .................................................... 153
5.8 Comparison of success rates ......................................... 154
5.9 An analysis of recommendation success rate based on stak membership. ........................................... 156
5.10 Average success rates based on stak size, for each context information ........................................... 158
5.11 A sequence of queries submitted by one user ..................... 161
5.12 A sequence of queries submitted by one user in an extended search session ........................................... 162
5.13 Summary user data .................................................... 164
5.14 Summary of session length (number of queries per session) .. 164
5.15 Query modification classes .......................................... 166
5.16 Average unique URLs, query terms and snippet terms across different session length ........................................... 166
5.17 Success rate for different session length without term weighting . 168
5.18 Success rate for different session length with term weighting . 168
5.19 Difference of success rate between standard session-based and term weighting ........................................... 169
5.20 Large vs. Small Search Staks, no term weighting, k=1 ......... 171
5.21 Large vs. Small Search Staks, with term weighting, k=1 ....... 171

6.1 Key contributions and results ....................................... 176

A.1 Main elements in DCC Model ....................................... 186
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Sequential actions in DCC curation lifecycle model</td>
<td>26</td>
</tr>
<tr>
<td>2.2</td>
<td>Movie database</td>
<td>59</td>
</tr>
<tr>
<td>3.1</td>
<td>Summary Scoop.it dataset</td>
<td>79</td>
</tr>
<tr>
<td>3.2</td>
<td>Summary Scoop.it dataset for experiment</td>
<td>93</td>
</tr>
<tr>
<td>4.1</td>
<td>Summary Scoop.it dataset</td>
<td>128</td>
</tr>
<tr>
<td>5.1</td>
<td><em>Stak query</em> (<em>s_Q</em>) representation</td>
<td>145</td>
</tr>
<tr>
<td>5.2</td>
<td>Staks Categories</td>
<td>153</td>
</tr>
</tbody>
</table>
Declaration of Authorship

I hereby certify that the submitted work is my own work, was completed while registered as a candidate for the degree stated on the Title Page, and I have not obtained a degree elsewhere on the basis of the research presented in this submitted work.

Signed: ____________________________________________________________

Date: __________________________________________________________________
'If the water of the sea were ink for a pen to write down the words, wisdom and signs of Allah, the sea would run dry before it all could be written down...'.
Quran 18:109
Abstract

The evolution of information sharing on the web has introduced a new chapter for information discovery. With all of the information and all of the people together in one place, there are more opportunities for creating, sharing, and discovering information. Now, many individuals are using their Twitter and Facebook accounts to share interesting pieces of content they locate. To some extent this is known as content curation, which involves users as curators who search, filter, organise and share the information they find. There is also a specific sites for content curation (e.g. Storify, Pinterest, Scoop.it, BagTheWeb) which provide users with a set of tools to manually collect, manage topical collections of content and share the content with others.

As it stands though curation is very much a manual process with the user solely responsible for performing each of the aforementioned steps in curating collections of content. We believe that we can alleviate some of this burden on the user by providing intelligent assistance at different stages of the curation cycle. In particular we focus on the search and organisation stages and identify two key tasks, assignment and discovery.

The assignment task involves situating new content within a collection of other related content. In this thesis we endeavour to automate this process and identify the correct collection for incoming content as it is discovered by the user, thus making the process both simpler and more efficient. We investigate recommender systems approaches and evaluate their efficacy for two different types of curation systems. The first, Scoop.it, is a traditional online curation service where users can both curate their own collections of content, and follow the collections of others. The second, HeyStaks, is a social search platform in which curation is directly integrated within everyday search. In HeyStaks communities of like-minded searchers can share curated repositories of search experiences.

The discovery task involves identifying new and interesting content for a user to curate. We examine this task within the context of Scoop.it. In particular, we exploit the information of collections that users have both curated and followed in order to establish their interests and recommend new collections for them to follow.
By improving the manner in which content is organised and discovered we believe this research will help existing curators, encourage new curators, and improve the quality of content collections in general. An increase in both the quantity and quality of curated collections should in turn benefit information seekers for whom search is increasingly not stringent enough in terms of seeking out the very best information.
Publications


xiii
Acknowledgements

I would like to take this opportunity to express my sincere gratitude toward my supervisor Barry Smyth for his invaluable guidance, and support. This thesis was made possible due to his constant encouragement and belief in me. I am extremely thankful to Rachael Rafter for being a very valuable mentor.

My thanks are also due to my colleagues in Clarity and HeyStaks: Ruihai Dong, Sandra Garcia Esparza, Stephen Bourke, Owen Phelan, Kevin McNally, John Hannon, Markus Schaal, Kevin McCarthy, Michael O’Mahony, Humberto Corona, Maurice Coyle and Peter Briggs with whom I enjoyed intellectually stimulating discussions. All of these people have broadened my perspective on research. I also wish to thank my family for their invaluable support, both emotional as well as intellectual.

Last but not least, I sincerely acknowledge the PhD scholarship from the Ministry of Education of Malaysia and Universiti Teknikal Malaysia Melaka, and also the funding from Science Foundation Ireland.
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOW</td>
<td>Bag-of-words</td>
</tr>
<tr>
<td>CSI</td>
<td>Collection Summary Index</td>
</tr>
<tr>
<td>DCC</td>
<td>Digital Curation Centre</td>
</tr>
<tr>
<td>DT</td>
<td>Decision Tree</td>
</tr>
<tr>
<td>HTML</td>
<td>HyperText Markup Language</td>
</tr>
<tr>
<td>IR</td>
<td>Information Retrieval</td>
</tr>
<tr>
<td>NN</td>
<td>Nearest Neighbors</td>
</tr>
<tr>
<td>LDA</td>
<td>Latent Dirichlet Allocation</td>
</tr>
<tr>
<td>LSA</td>
<td>Latent Semantic Analysis</td>
</tr>
<tr>
<td>LSI</td>
<td>Latent Semantic Index</td>
</tr>
<tr>
<td>NBM</td>
<td>Naive Bayes Multinomial</td>
</tr>
<tr>
<td>NLP</td>
<td>Natural Language Processing</td>
</tr>
<tr>
<td>ODP</td>
<td>Open Directory Project</td>
</tr>
<tr>
<td>RSS</td>
<td>Rich Site Summary</td>
</tr>
<tr>
<td>SEO</td>
<td>Search Engine Optimisation</td>
</tr>
<tr>
<td>SSI</td>
<td>Stak Summary Index</td>
</tr>
<tr>
<td>STS</td>
<td>Social Tagging System</td>
</tr>
<tr>
<td>SVD</td>
<td>Singular Value Decomposition</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machines</td>
</tr>
<tr>
<td>TB</td>
<td>Term-based</td>
</tr>
<tr>
<td>TF-IDF</td>
<td>Term Frequency Inverse Document Frequency</td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

The information overload problem on the web seems to be the problem that just keeps on giving in terms of the interesting challenges it presents for researchers [1, 2]. For a long while, mainstream search engines like Google, Bing and Yahoo have provided an almost perfect fit for finding information online, developing elegant solutions like Google’s PageRank algorithm that could search and rank millions of webpages in a useful manner [3]. However, today the web continues to grow and given the popularity of Web 2.0 technologies, the information overload problem remains a serious challenge. Traditional search engines are now simply indexing and retrieving too many results for the end user to digest. In fact, it is increasingly accepted that search is not always the best approach to begin with, as users often are unsure of what they are actually seeking [4]. Therefore information discovery has become more important, allowing users to find (discover) interesting information without having to search for it specifically or even be aware it exists [5].

The recent information sharing and discovery trend has introduced a new generation of curation services (e.g. Delicious\textsuperscript{1}, Storify\textsuperscript{2}, Pinterest\textsuperscript{3}, Scoop.it\textsuperscript{4}) which

\textsuperscript{1}http://www.delicious.com
\textsuperscript{2}http://www.storify.com
\textsuperscript{3}http://www.pinterest.com
\textsuperscript{4}http://www.scoop.it
provide users with a set of tools to manually curate and manage topical collections of content. In particular, content curation involves humans who collect and organise information relevant to a particular topic or area of interest. It is a future-oriented activity, that is designed to facilitate future discovery and consumption [6]. There are many examples of people collecting information for future use, such as organising and managing URLs or bookmarks [7]. In fact, people do not simply passively keep this information; they make extra effort to organise it in ways that will promote future retrieval. For example, a survey by Pew Internet & American Life in 2012 shows that 41% of online adults assemble collections photos and videos on sites specifically designed for collecting and sharing [8]. One example of such site is Flickr\(^5\), which allows users to organise, annotate and share photos. This site contains hundreds of millions of photos that are uploaded, tagged and organised by millions of their registered users.

Certainly, curation services benefit users who are looking for quality content since the content is contextualised, arranged and organised by other users in a way that is easy to find and simple to understand. In this thesis we will focus on curation as a platform and opportunity for recommender systems. In particular we will explore recommender systems solutions to some important curation tasks.

### 1.1 Towards the Curated Web

The problem of information overload has pointed to a need for new ways of content discovery and organisation. In this section we explain how the web has changed in terms of the way information is being discovered, organised and shared, through the development and adoption of *curation* systems.

\(^5\)http://www.flickr.com
1.1.1 From Curation 1.0 to the Age of Search

In the early 1990s there were no search engines as we would know them today. If a user wanted to get to a particular web page then they either provided the URL directly to the browser (if they knew the address) or used a portal like Yahoo to browse. Using an early version of Yahoo web users could navigate directly to content by traversing a hierarchical tree of category labels. Basically the Yahoo portal consisted of links to web pages which were manually organised into hierarchical tree of categories (taxonomies) such as Art, Business, Computer and Economy see for example in Figure 1.1. This can be seen as the era of

Yahoo


[ Yahoo | Up | Search | Suggest | Add | Help ]

- Art (819) [new]
- Business (8346) [new]
- Computers (3266) [new]
- Economy (898) [new]
- Education (1839) [new]
- Entertainment (8814) [new]
- Environment and Nature (268) [new]
- Events (64) [new]
- Government (1226) [new]
- Health (548) [new]
- Humanities (226) [new]
- Law (221) [new]
- News (301) [new]
- Politics (134) [new]
- Reference (49.5) [new]
- Regional Information (4507) [new]
- Science (3289) [new]
- Social Science (115) [new]
- Society and Culture (933) [new]

There are currently 31897 entries in the Yahoo database

Figure 1.1: Yahoo Directory in 1994
Chapter 1: Introduction

Curation 1.0, in which the curation was done by the dedicated professionals (e.g. Yahoo employees) whose job was to organise content and the end users acted only as the consumers of content.

The hierarchical structure in Yahoo directories allowed information seekers to browse but not search for relevant content. While this approach provided a convenient platform for seeking web pages, it had a number of disadvantages [9]. First, accurately and consistently classifying web pages into categories was, for the most part, a manual curation process and therefore did not scale well with the size of the web. Second, in order for a user to effectively discover web pages in this manner, the user’s knowledge of what sub-tree(s) to browse for a particular topic had to correspond to the classification scheme. This quickly became difficult as the size of the categories grew. Also the relevance of pages could change over time, for example pages organised under category X might need to be re-categorised under category Y. Given these problems, the popularity of taxonomies had declined over time [3]. At the same time search engines came to offer a working alternative.

Web Search Engines

Early search engines such as Infoseek and AltaVista popularised the notion of automatically building indexes of searchable content. These search engines were built based on information retrieval (IR) technologies [10, 11]. Early search engines constructed their own index of the web using crawler programs. Basically, they used crawler to collect web pages, then analysed the content of each page by recording the words, and their frequencies contained in each page. Then, in order to respond to a search query, the search engine retrieved and ranked pages that contained query terms.

A Boolean model was used by earlier search engines as their retrieval mechanism [11]. It provided a simple boolean term matching, and retrieved documents based on whether or not they contained the query terms. As an example of a Boolean retrieval, consider the following query for a search engine that has indexed a collection of documents. The simple query ”java” would retrieve a large
number of documents that mention "java" in the content such as Java island in Indonesia and Java programming language. All of these documents would be equivalent in terms of ranking in the Boolean retrieval model. The user may attempt to narrow the scope of search with the following query, "java AND programming". This query will retrieve a set of documents that contain both words occurring anywhere in the document. User may also eliminate or add documents by using NOT or OR operator in their queries, for example "java AND programming AND NOT (oracle OR sun)". The main drawback of the Boolean model was that, it only retrieved the documents that exactly matched the query which led to either too few or too many documents being retrieved with no ranking ability possible.

The vector space model [12] is later adopted by many web search engines as it provides a mechanism with which documents can be ranked according to a finer grained relevance measure than binary. Using this approach, documents in the search engine’s index are represented as a \emph{bag-of-words} (BOW), in other words as an $m$-dimensional vector $(w_1, \ldots, w_m)$ where $w_i$ represent a weight for the $i$th unique index term in a given document, see Figure 1.2. The weights

\begin{figure}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
 & Life of Pi & Spider-Man & Inception & Rio & Godzilla \\
\hline
book & 1 & 0 & 0 & 0 & 0 \\
comic & 0 & 1 & 0 & 1 & 0 \\
animation & 1 & 0 & 0 & 1 & 0 \\
adventure & 1 & 1 & 1 & 1 & 1 \\
child & 1 & 1 & 0 & 1 & 0 \\
action & 0 & 1 & 1 & 0 & 1 \\
monster & 0 & 0 & 0 & 0 & 1 \\
\hline
\end{tabular}
\caption{Term-document matrix for a collection of five documents}
\end{figure}

indicate how representative the term is of the document, with term frequency
being a popular way to measure this. Furthermore, the frequency of the word in a document can be offset by the frequency of the word in whole collection of documents, thus differentiating between truly representative words and simply common words. This is known as TF-IDF weighting. Details for computing of the term weighting is explained in Section 2.4.5.

Based on BOW document representation, the vector space approach is used to measure the similarity between a search query and the documents in a collection index. Similar to the document, the query could also be represented as a unit vector. The standard way to quantify the similarity between the query and the document is to compute the cosine similarity between their vector representation [9]. Then the documents may be ranked in order of decreasing cosine similarity values so that the most relevant documents appear at the beginning of the list. Although term matching and weighting information go some of the way towards providing quality search engines, however, on their own there are insufficient, as vital information about the quality and relevance of web pages is not incorporated into the retrieval mechanism [10].

The work of Page and Brin [3] improves the ranking process by taking the advantage of the connectedness of web pages, and use this information to evaluate the relative importance of individual pages. They introduce a new innovation in ranking strategies called PageRank and implement it as the main component of the Google search engine. PageRank takes into account the link structure of the web, seeking to use the links between on-line documents to identify the more influential and reliable sources of information. Essentially a web site is given a higher reputation of prominence (and thus more likely to appear higher in search results), if other reputable or prominent sites link to it. Specifically, as in Equation 1.1, the PageRank value for page \( u \) is dependent on the PageRank values for each page \( v \) contained in the set \( B_u \) (the set containing all pages linking to page \( u \)), divided by the number \( L(v) \) of links from page \( v \). For example, Figure 1.3 shows a simplified illustration of PageRank score calculation.

\[
PR(u) = c \sum_{v \in B_u} \frac{PR(v)}{L(v)} \tag{1.1}
\]
Chapter 1: Introduction

with

- $B_u$: the set of pages that point to page $u$.
- $L(v)$: the number of links from page $v$.
- $c$: a factor used for normalisation so that the total rank of all web pages is constant.

![Figure 1.3: A simplified illustration of PageRank score calculation [13]. Page C has a higher PageRank than Page E, even though there are fewer links to C; the one link to C comes from an important page and therefore it gets higher value.](image)

Problems in Web Search

Web search has become one of the most important means of information seeking, helping to fulfil the information needs for millions of searches everyday. However, if users do not know what they are looking for or have only a vague idea of what they are looking for, then they will usually either fail to find relevant information or waste a lot of time trawling through irrelevant result-lists [14]. The evidence suggests that up to 50% of search sessions fail to deliver relevant results, thus the search sessions fail to lead to result selections [15].
Chapter 1: Introduction

The recent culture of sophisticated search engine optimisation (SEO) techniques and so-called content farming strategies have led to a degradation of search result [16]. SEO is the practice of optimising web pages in order to improve their ranking in a "natural" or "organic" search result. Similarly, content farming strategies are designed to boost the rank of targeted results, often to the detriment of the individual searcher. In particular with these techniques users will receive result-lists with high rankings for low quality content such as the article that was written by authors with limited knowledge of subject they cover even though better quality (but less optimised) content is available. McCreadie et al. [16] find that, one in every two of the search queries, a content farm article made the top five ranked in the search results. Basically, the content farms aim to leverage search engines to drive more visitors to their pages and generate revenue from on-page adverts.

Social Search

Web search engines like Google and Bing have begun to address problems in web search by using ideas borrowed from social filtering and discovery research [17–19] which allow social networks users to influence result-lists. For example, Google has the +1 button which allows users to share and recommend results to their friends and contacts. If a user finds an interesting web page, he can click the +1 button and in turn his friends or contacts will see this +1 in any result list containing this page. In other words the pages that his friends have found to be interesting are highlighted during his own searches [20], see Figure 1.4.

The integration between Bing and Facebook, allows Bing to promote and highlight results in a similar way. In particular Bing integrates the Facebook like within its regular search results. For example, as in Figure 1.5, if a user searches for something that one of his friends has liked on Facebook, he will see this information displayed by Bing alongside the search results6. The goals of social discovery and filtering, then, is to aggregate and share the useful results of individual activity and knowledge. Specifically, it facilitates the dissemination

http://blog.facebook.com/blog.php?post=437112312130

6
Chapter 1: Introduction

1.1.2 Curation 2.0

Nowadays, there is renewed interest in the organising and sharing of web content. Previously, in Yahoo, the categories were curated manually by Yahoo’s
employees and the indexes of search engine were automatically built by crawlers.

Today, using the new generation of curation services, the users themselves have more opportunities to do the curation. We refer to this as Curation 2.0. This is due to the popularity of so-called Web 2.0 advances in web design, which means web sites are more dynamic and their visitors are encouraged to actively interact with the web pages such as write comments or share photo [21]. For example, by using social bookmarking websites such as Delicious or Bibsonomy\(^7\), users can annotate, organise and bookmark a web page for later retrieval using keywords or tags. In this way, tagging is as an alternative to strict categorising hierarchies (as in Yahoo). In other words, tagging is usually considered as user-defined categorisation. It is used to construct and maintain a navigational aid to the resources. In addition, tagging can be viewed as a means to accurately and precisely describe resources so that the tags are useful for later searching. Figure 1.6 shows a snapshot from Delicious in which the user is presented with the list of web pages and the tags that are associated with the page.

Figure 1.6: The early version of Delicious (formerly known as Del.icio.us) main interface

\(^7\)http://www.bibsonomy.org
Chapter 1: Introduction

The social bookmarking trend which started with Delicious a decade ago is today exemplified by a new type of curation services (e.g. Storify, Pinterest, Scoop.it, BagTheWeb⁸) which provide users with a set of tools to manually collect and manage topical collections of content. For example in BagTheWeb, users can see the collections (or ”bags”) of URLs which are shared by other users. Users also can create their own collections, see Figure 1.7. We will return to these services in Chapter 2.

⁸http://www.bagtheweb.com

Collections themselves become new types of media [22–25]; for example Storify allows users to narratively knit together collections of links and content to form unique stories. Users of this kind of application act both as producers and consumers of content, creating their own collections and following interesting collections of others. Importantly, by following other users’ collections, users are kept updated on what other experts are curating in their collections, and in turn, can leverage this information for their own information needs [26].
1.1.3 Beyond Curation 2.0

The web has seen an explosion of social tools that are empowering more people to create and share content on the web [27]. To a certain extent, the web has always been about curation; it has always been loosely about users collecting, organising and sharing links with each others.

In the past, curation and search have been viewed as two different information discovery approaches. But there are good reasons to look now at the intersection between these two approaches. Specifically how can we make search more social and collaborative and how can curated content usefully influence mainstream search engines. In particular, while searching for information using a search engine, there is an opportunity to store and share search activities and experiences such as the search queries and the useful links found for those search queries. In some cases users can also organise these experiences based on topics of interest.

HeyStaks\textsuperscript{9} is one of such system that could support curation in web search [28]. HeyStaks was not originally conceived as a curation service. It is primarily a social search service, which is designed to complement mainstream search engines by recommending relevant pages to users based on the past experiences of search communities [29]. But it is useful to consider it as a curation service and it is reasonable to do so as it combines ideas from web search, curation, and social networking to make recommendations to users, at search time, based on topics that matter to them. Furthermore, it allows searchers to organise and share their search experiences and to collaborate with others as they search. The details of HeyStaks are described in Chapter 2.

1.2 Thesis Motivations

Curation is still in its infancy, but there are many services which allow content curation by public internet users. Curation is not only for web content but

\textsuperscript{9}http://www.heystaks.com/
Chapter 1: Introduction

also includes other types of digital data such as specialised repositories of scientific dataset (UniProt\textsuperscript{10} and Protein Bank Data\textsuperscript{11}). These repositories have been curated by human effort through consultation, verification, and aggregation of existing sources, and interpretation of new raw data (experimentally obtained) \cite{30}. In this domain the curators are dealing with structured and consistent data which is easier to process compared to unstructured nature of the web content.

Curation of web content used to exist years ago where people curate collection of links (URLs). Now it has become more formalised and sophisticated in which it involves a number of activities such as content assignment and discovery. Of course content assignment and discovery not a major problem yet, but for anyone who involves in curation very quickly they reach a point at which it become problem or at least it add friction of workflow to them. Therefore anything that can reduces this friction is important. So the main motivation of this thesis is to look at a growing area of service or application namely curation and identify problems that soon will be major and try to develop solution. Specifically, we focus on various ways to form and organise a collection of content. We also look at how curated content can be harnessed and discovered.

1.2.1 Using Curation as Platform for Recommender System

Content curation right now can be a fairly manual process and in general it is to the benefit of users if we can identify ways to automate some of the classical content curation workflow. Currently many curation services allow users to host a browser bookmarklet to make curation a little easier as users can organise and store information of curated webpage without having to leave the original site. However we believe that it should be more deeply integrated with services like search so that curation can happen as a consequence of other discovery activities. For example the service provided by HeyStaks is helping users to

\textsuperscript{10}http://www.uniprot.org
\textsuperscript{11}http://www.rcsb.org/pdb
curate as they search (using HeyStaks and their favourite mainstream search engine) and the results that they select (or tag or share) will be associated with their active collection. The active collection has to be specified at the start of their search session. However, many users forget to do so, then search experiences can be mis-recorded, compromising collection quality and leading to poor organisation in the future. Without some automatic/semi-automatic collection recommendation or assignment some content will not get curated which means that the collections will not be as rich and interesting as they could be.

Collection assignment refers to the act of associating new content within a collection of other related content. The assignment task is also related to tagging or labelling as both are about supporting the process of organising the content to facilitate retrieval in the future. There are a number of studies in tag recommendation which propose approaches to automatically generate appropriate tags for a given resource [31–36]. The aim of tag recommendation to improve the tagging process from generation process to one that require users to only recognise appropriate tags which significantly reduces the cognitive burden and increases the overall performance [31]. Similarly in collection assignment, in which a user is about associate new content within a collection, if we can make the curation process easier, more people will be likely to curate content themselves. The increment in the number of people involved in curation will in turn increase the likeness of richer web content and benefit the entire web community particularly for information discovery.

While the advantages of well organised content are clear, it is important to be aware of some problems when dealing with human curated content, for example the misplaced content, that is, the misclassification of content in a collection [23, 37, 38]. One scenario of an ineffective collection organisation is illustrated in Figure 1.8. This figure shows a collection (or story) in Storify consists of a number of story items which suppose share the same theme. As we can see from this figure, the first item is an off topic or mis-filled content and ultimately this could affect the quality of the collection in term of providing contextualised web content. This situation will corrupt the collection as the poorly curated
**Chapter 1: Introduction**

Figure 1.8: Example of an off-topic page; The main theme of this collection is related to "JQuery" however there is one off topic content which is related to "MySQL PHP Tutorial".

Content can at best lead to irrelevant content that becomes isolated within a collection and at worst can render the collection useless.

Our motivation here is to improve curation process by helping users to assign content to their collections more efficiently. We will do this by profiling the target content and to use these profiles as the basis for recommending the evolving collections managed by a curator at assignment time. Basically we will
use recommender systems approaches in an effort to automate this assignment task.

### 1.2.2 Collection Discovery

The growing popularity of curation services, including their large user communities and ever-growing networks of user-generated content, has made them an attractive object of investigation for researchers from different disciplines like social network analysis [24, 26], data mining [39], information retrieval and knowledge discovery [40]. Curation sites connect millions of people to content which has been intentionally curated and organised and may therefore be more useful (or certainly different) to which mainstream search engines like Google or Bing might provide [25]. This is because they can follow (or subscribe) one another or follow specific collections in a Twitter-like following model, forming networks based on shared interests. For instance in Figure 1.9, an example from Pinterest, a user can follow other users’ boards (or collections); showing common interest between the creator and the followers. In particular, information consumers are able to discover new content using this site because the actions of curators caused the content to be stored by the curation site in the first place, through the action of selecting and organising the content as collections. In this case, discovering social interests shared by users is very important because it helps to connect people with common interests and encourages people to contribute and share more content [41, 42].

Currently many curation services provide a query-based search service and category navigation that allow users to look for content or collections of interest. However, as we can see in Figure 1.9, there are lots of collections available for users. Finding an interesting collection is difficult given that there are so many to choose from. Moreover, the number of collections is increasing every day. This produces overwhelming amount of content, much beyond what users have time to process as they need to spend significant level of time to identify high-quality content [25].
Our motivation here is to facilitate users to identify potentially interesting collections to follow (subscribe) and start receiving update information from these collections. These collections can be considered as information sources which can provide interesting content to users. We will do this by using a recommender system to automatically recommend relevant collections for a user among a large number of collections.

### 1.3 Thesis Contributions

The work that we present in this thesis consists of four main contributions in the area of recommender systems, specifically focusing on supporting the assignment and discovery tasks in content curation.
1.3.1 New Setting for Recommender Systems

Up until now most recommender systems have been about recommending items to users. For example, there are recommenders for recommending music [43], books [44, 45], movies [46] and products [47] to users. In the beginning these items were atomic, in the sense that they were simple items with no configurable parts; a book is just a book. More recently people have started to consider some of the challenges that go with recommending more complex items to users; for example a travel recommender may recommend a vacation which has different components such as flights, hotel, car hire etc. [48]. Each of these components can be independently configured which adds a lot to the recommendation complexity.

In another direction some recommender systems researchers have started to look at different user types and in particular moved from making recommendations to individual users to groups of users [49, 50]. This adds more complexity again because groups of users can have conflicting preferences for example and recommendations may need to be carefully chosen to balance the differing preferences that users have.

The above recommender systems work, new as it may be, still fits within the general framing of recommending items to people. Typically, recommender systems refers to a very general type of technology that is about making suggestions to a user without the need for an explicit query. As such there is generally some pseudo profile that can be associated with the user (often a user profile). The classic recommender system suggests concrete items to a user (movies, music, books etc.) but of course almost anything can play the role of an item.

One of the main contributions of this thesis is to identify new types of recommendation problems or new settings for recommender systems. Specifically, in adopting a recommendation approach to solving the assignment problem in curation, we are framing a recommender system that recommends collections to pages as such collections play the role of items and pages play the role of users. Figure 1.10 illustrates the new setting of recommender systems which is applied in this thesis. In particular, an "item" corresponds to a collection and the job
Chapter 1: Introduction

of the "recommender engine" is to suggest a collection for a given page. The "page" is playing the role of the user and the pseudo profile that is associated with the page includes things like the keywords in the page title, snippet text etc. This is an interesting variation of classical recommender systems in itself not the least because it also introduces new types of data that can be used to guide the recommendation process.

![Diagram](image)

Figure 1.10: Proposed system vs. classical recommender system.

1.3.2 Intrinsic and Extrinsic Profile

Normally recommender systems are guided by intrinsic information that is directly associated with the user (their ratings, preferences, browsing histories). That said, it is interesting to consider extrinsic profile information as in the work of Hannon et al. [51] and Chen et al. [40]. To begin they start with an intrinsic profile made up of the user’s own tweets. But then go on to show that profiling a user based on their own tweets plus the tweets of their friends and followers serves to further improve recommendation quality. In a sense, the
intrinsic information form a basic profile and the addition of extrinsic information corresponds to a type of query expansion [52, 53], augmenting the limited intrinsic profile with additional relevant information.

In the case of curation, there are many different strategies that can be applied to enrich the intrinsic profile to guide the recommendation. Therefore, the second contribution of this thesis is the identification of the use of intrinsic and extrinsic profile information during recommendation. The key idea is the adoption of internal and external knowledge such as existing features in an item and world knowledge that is related to the item. For example, in the recommender system for collection assignment, the page profile that is used to guide recommendation can be augmented with extrinsic information from external knowledge in order to improve recommendation quality. In particular, the extrinsic profile is made up of the external information which are related to intrinsic profile in some way.

1.3.3 Recommending Collections of Items

 Recommending collections of items is a different task from recommending individual items. For example items in a collection are complementary so that the value of each item is increased when it is combined with other items (e.g. music playlist, meal plan). Many popular online systems facilitate the creation user-generated collections such as ”board” in Pinterest, ”story” in Storify, ”gallery” in Flickr and ”playlist” in Youtube. Most of these sites currently only support manual creation of collections, still missing out on the opportunity to recommend relevant items to collections or recommend collections to users.

To date most research in recommender systems has addressed the challenges in recommending individual items. These have been successful and useful in various domains. However only a small number of studies have explored how to effectively recommend collections of items. For example, the studies by Hansen and Golbeck [54, 55] are only focused on the design space for collection recommender systems and there is no development on a real application. The recent
work by Kamath et al. [56] explore content-based approach for recommending new boards (collection of images) to Pinterest users which is closely related to our work.

Thus, our next contribution is the development and evaluation strategies for recommending collections in the context of content curation particularly for collection assignment and discovery. More specifically, we identify a number of approaches to represent the collections of web pages and profile users’ interests in order get better recommendation.

1.3.4 Analysis on Live-user Data

For the purpose of this study we conduct a detailed analysis of our proposed designs using live-user data collected from the Scoop.it and HeyStaks. Scoop.it is an established curation service which provides a conventional approach for content curation while HeyStaks is a novel application for curation focusing on social search. We crawl Scoop.it and HeyStaks API in order to gather data about users, the collections they created and followed and the content associated with those collections. We also extract data from the HeyStaks usage logs in order to get data related to search activities such as search queries and click-throughs links.

1.4 Thesis Outline

This thesis is organised as follows. The next chapter presents the research background where we discuss the fundamental concepts for content curation and recommender systems which serve as common foundations for the research we explore in Chapters 3, 4, and 5. We also give an overview of two curation services, Scoop.it and HeyStaks that we use as case studies in this thesis. In Chapter 3, we describe an additional assistance to users in curation tasks, in particular when it comes to efficiently assigning content to collections. Specifically for the evaluations in this chapter we use the live-user data from Scoop.it.
In Chapter 4, we design and study a recommender for information discovery particularly in the surroundings of curated web where we evaluate our approach using dataset from Scoop.it. In Chapter 5, we examine on one particular approach of curation in web search using HeyStaks and recommendation strategies to support the assignment task in an integrated form of curation. Finally, in chapter 6, we summarise the context and contributions of this thesis and conclude with a discussion of the most promising areas for future work.
Chapter 2

Background Research

In this chapter we discuss the fundamental concepts of content curation and recommender systems which serve as the foundation of this thesis. Recommender systems focus on ways to make proactive suggestions to users based on their learned preferences and/or current context [57]. Such systems typically provide users with a list of recommended items they might be interested in and which are relevant to their current context. Recommender systems benefit users by reducing the amount of time and effort needed to search for information, and can help users find interesting information they were not aware of in the first place [58].

We will begin by giving an overview of content curation and develop a functional definition for it. We will examine curation from the perspective of real and popular curation services that are in use today and from a more academic perspective that will highlight recent advances in the state of the art of curation. Following this we will detail two curation services, Scoop.it and Heystaks which we use as case studies throughout this thesis.

We then move on to provide an overview of recommender systems focusing on two common approaches, content-based and collaborative recommender systems. We will conclude this chapter with a high level summary of how recommender system ideas can be used to solve some of the key challenges facing
modern curation systems. These challenges will form the basis of the remainder of this thesis.

2.1 Defining Content Curation

Curation: *The act of organising and maintaining a collection of artworks or artefacts.* -Dictionary.com

The traditional way of defining curation is with reference to the work of curators, the professionals who typically manage and take care of artifactual collections at culture heritage institutions and who organise exhibits in galleries [59]. For instance, curators of museum objects or physical samples are typically experts in their domain, and their role requires a particular level of content expertise [60].

Recently, a new type of curation involving digital data has emerged, known as *digital curation* [61]. Digital curation may also be referred to as *content curation* and increasingly it is used to refer to the actions needed to annotate and organise digital content for current and future use [24, 62]. It involves a number of activities such as maintaining, preserving and adding value to digital data throughout its lifecycle (see Section 2.1.1) and the curated content may be shared and discussed among the wider community.

Rotman et al. [60] define curation as the actions performed by individual users to identify, select, validate, organise, describe, maintain, and preserve existing content. Bhargava [63] suggests a definition of content curation as the act of finding, grouping, organising or sharing the best and most relevant content on a specific issue. According to Kanter [64] content curation is the process of filtering through the huge amounts of content on the web and presenting it in a meaningful and organised way around a specific theme. Based on these definitions, in the next section we synthesise a 4-stage model that will be used in the rest of this thesis.

Content curation has been heavily influenced by Web 2.0 trends and social media initiatives and tools [21, 23, 42]. For example, a number of curation