Electronics and Computer Science
Faculty of Physical Sciences and Engineering
University of Southampton

Jamie Davies
9th December 2013

Enhanced Content Analysis and Information Retrieval Using Twitter Hashtags

A project report submitted for the award of
MEng Computer Science

Supervisor:
Dr. Nick Gibbins

Examiner:
Dr. Klaus-Peter Zauner

Abstract

One of the key characteristics of Twitter and other microblogging platforms is the use of 'hashtags' — topical/categorical annotations provided by the authors of the posts (tweets) themselves. This flexible system was designed for the effective organisation and searching of tweets, but with Twitter facing an ever-increasing number of users and tweets it is hard for users to keep track of the vast number of hashtags in popular use. This results in data from the hashtags being fragmented and inaccurate due to the users making poor or uninformed hashtag choices.

If users are presented with a choice of relevant hashtags when writing a tweet, they are more likely to publish tweets with accurate tag data. This project aims to create an intelligent hashtag recommendation tool to raise the information gain from hashtags. However, whilst such a system could improve the quality of the hashtag data for future tweets, tweets that have already been published will remain untouched by the system. Thus, the system will be extended to also retrofit hashtags to published tweets — allowing for tweets to appear in search results for a particular hashtag even if they don't actually contain the hashtag in question.

Contents

1 Introduction

Some introduction text.

Through the creation of the system, new insights and understanding of how people use Twitter could come to light. This provides the possibility to attempt to answer optional interesting questions, such as

- Are certain types of tweet/hashtag easier to classify than others?
- Is it possible to make relevant hashtag suggestions using just the tweet text itself, or is other metadata needed to make the recommendations useful?

1.1 Project Goals

This project will create a system that aims to support and enrich the information provided by hashtags on Twitter[1]. It will use a combination of different machine-learning techniques to examine and classify the topics and concepts behind the hashtags and in doing so, be able to suggest suitable hashtags for tweets that are relevant to their content. This will allow users to make a better choice of hashtag when writing tweets, and therefore refine the information that they provide.

However, as suggesting better hashtags will only improve the information gain from future tweets, the system will be extended to provide a context-aware tweet search facility. This will enable users to search for a particular hashtag, and instead of only returning tweets containing that hashtag (as current systems do), it will also provide tweets that are contextually relevant to the search term but do not contain that given hashtag.

[1] `www.twitter.com`

2 Background and Literature Review

The main design goals behind hashtags are to categorise tweets and allow them to show up more easily in searches[2]. Whilst the task that this project is aiming to complete is novel and fairly unexplored, it is well connected with other experiments, systems and projects within the research community.

2.1 Recommendation Systems

Traditional recommendation systems are in place all over the web today. From music discovery services (such as Last.fm[3]) to suggested purchases on retail sites (like that in place at Amazon[4], these systems are all personalised recommendation engines that take an individual user's preferences and use them to provide suggestions tailored to that user.

2.1.1 Collaborative Filtering

Most personalised recommendation systems employ a set of techniques known as collaborative filtering. These techniques were first coined by Goldberg et al. (1992), where a system named *Tapestry* was created that allowed people to attach annotations to documents, and then use that information to filter the documents for other users.

One common implementation of collaborative filtering is the so-called "user-to-user" approach. "User-to-user" collaborative filtering works by taking the preferences of a user A, and finding a small subset of other users in the system that have similar preferences. For each user B in the subset any items that B has adopted that A hasn't are added to a ranked list of suggestions. A is now more likely to adopt items in the list than the items of another random person (Schafer et al., 2001).

2.1.2 Content-Based Recommendation

Another approach to provide relevant recommendations to a user is the use of content-based recommendation systems. This is a type of system that recommends items relevant to other items by comparing the details and descriptions of the items themselves. This can be extended to suggest items for a user by comparing a content-based description of the user's preferences with the descriptions of the items (Pazzani and Billsus, 2007).

[2] `https://support.twitter.com/articles/49309-using-hashtags-on-twitter`
[3] `www.last.fm`
[4] `www.amazon.co.uk`

A key issue with content-based filtering is that the recommendations can only be as accurate as the algorithm used to derive a user's profile. There are a number of algorithms available to build user profiles, depending upon the context, but essentially a content-based profile is created using a weighted vector of item features. The weights mark the importance of each feature to the user, and can be computed from individually rated content vectors.

Cantador et al. (2010) studied and evaluated a number of content-based recommendation models based upon the premise of user and item profiles being described in terms of weighted lists and tags. Through their experiments they found that models that focused on user profiles outperformed the models oriented towards item profiles in nearly every test. They go on to suggest that a better way of profiling users would be through the use of tag clustering.

2.1.3 Relevance Feedback

Relevance feedback is a process that was originally designed for information retrieval, and works on the assumption that a user can not always correctly encapsulate into a query what it is they are searching for. It works by allowing a user to create an initial query to which an initial set of results is returned. Out of these initial results, the user can then mark certain results as relevant or irrelevant, and this information is then submitted and used to refine the original query and return more relevant results to the user (Salton and Buckley, 1990).

Instead of limiting recommendation systems to the accuracy of their classifiers, a common approach is incorporate relevance feedback techniques. Utiyama and Yamamoto (2006) showed that it is possible to combine collaborative filtering, content-based filtering and relevance feedback techniques into one system to provide better recommendations.

2.2 Hashtag Recommendation Research

Even though providing hashtag recommendations and suggestions is still a new and largely unexplored field, there have been several efforts to improve the hashtag experience for Twitter users.

2.2.1 Current Twitter Hashtag Implementation

The current hashtag system on Twitter (Figure 1) uses a non-personalised auto-complete tool to provide suggestions to the user. Whenever a hash symbol (#) is typed in the tweet composer, the system simply suggests hashtags starting with the letters that the user has typed so far. These suggestions are

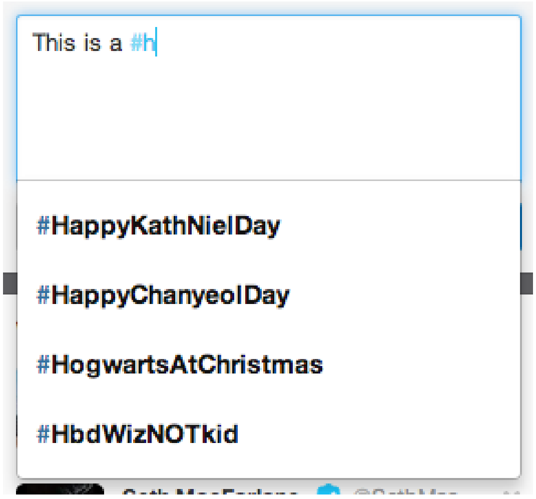


Figure 1: Twitter's current hashtag suggestion system.

chosen from a tiny subset of hashtags, taken from a mixture of the currently trending[5] and from the user's history. Whilst better than not having suggestions at all, this system is only truly useful in a specific use case: when the user knows the starting letters of a trending hashtag they want to use, or are trying to recall a hashtag from they have previously used. This system does not help the user choose the correct hashtag for their tweet.

2.2.2 Using Tweet Content to Suggest Hashtag Probabilities

Mazzia and Juett (2011) used an application of a naive Bayes model to suggest hashtags by using just the content of the tweet as input. The Bayes model allows the system to calculate the probabilities of the tweet using different hashtags. This gave them suggestion correctness rates of up to 72%, using various cross-validation data sets. They remark that a much larger training set would be required for real-world use, however.

2.2.3 Comparing Tweets To Other Tweets

By assuming that the primary purpose of hashtags is to categorise tweets and improve searching (as Twitter envisioned), Zangerle et al. (2011) created a system that recommends hashtags for a tweet by taking tweets from other tweets that are textually similar to the query. The similarity between tweets is calculated with the TF-IDF (term frequency – inverse document frequency) model. The hashtags are then extracted from the similar tweets, ranked according to how similar the tweets were to the original query, and returned as a list of

[5] Trending hashtags are those with the highest rise in usage within a given time period.

suggestions to the user. A number of different ranking algorithms were tested, but this was found to be the most successful.

2.2.4 Creating Personalised Recommendations

After studying the advantages of providing personalised recommendations in retail situations on a per-user basis, Kywe et al. (2012) realised that a similar approach towards hashtags could prove fruitful. Hashtag use varies from user to user, with some users using the latest trending hashtags, other uses only using a specific set or type of tag, and with some users barely using them at all. They proposed a personalised hashtag recommendation system that considers both user preferences and the query tweet content: the system creates a ranked list of hashtags from both the most similar users and most similar tweets. This gave promising results, although it was noted that this may not be the best recommendation system for all types of tweets and hashtags.

Shepitsen et al. (2008) used a hierarchical agglomerative clustering algorithm to profile users and provide personalised recommendations in collaborative tagging systems. They found that clusters of tags can be effectively used to ascertain a user's interests, which could then be used in a traditional content-based recommendation approach. This technique worked well, particularly for dense folksonomies such as Last.fm.

2.2.5 Overcoming Hashtag Duality

Observers of social media have realised that hashtags play a dual role within the online microblogging communities, such as Twitter. On one hand, hashtags fulfil the design goals that Twitter created them to accomplish (bookmarking and improving search); on the other hand, however, they serve as a badge of community membership, connecting users together. Yang et al. (2012) took this duality into account when attempting to create a hashtag recommendation system by training a SVM (support vector machine) classifier with a variety of features taken from the tweet metadata to overcome the duality and suggest relevant hashtags.

2.3 Broader Classification in Twitter

Twitter is a thriving[6] metropolis of users expressing themselves on a daily (and often more frequent) basis, and has grown exponentially in size since its

[6] `www.bloomberg.com/news/2013-10-15/twitter-revenue-more-than-doubles-in-third-quarter.html`

inception in 2006. Due to this, the data that it contains has caught the attention of researchers throughout computer science and even other disciplines. Whilst the concept of recommending hashtags is relatively unexplored, there have been many other classification experiments run with Twitter data.

2.3.1 Categorising Tweets

Sriram et al. (2010) proposed an approach to classify tweets into 5 general categories: *news*, *opinions*, *deals*, *events* and *private messages*. This was achieved by using a small set of specific features from each tweet, instead of using the traditional "Bag-Of-Words" (BOW) text classification method. The BOW approach is centred around counting occurrences of words in the text, but in the case of Twitter and its 140 character limit, it is very rare that words are actually repeated in a tweet.

2.3.2 Categorising Users

Another approach to deciphering the vast quantity of data on Twitter is to classify the users themselves. Twitter has become a powerful platform for people posting content about events, and as such it would be useful to automatically establish *who* is participating in these events. By taking a number of features from each user account and passing them through a K-Nearest Neighbours (KNN) algorithm, De Choudhury et al. (2012) developed a system that would classify a user's behaviour into one of three categories: organisations, journalists/media bloggers and ordinary individuals.

3 Design

Aside from background reading and research, the work completed towards the project so far has been heavily grounded in data collection and analysis.

3.1 Requirements

There are two main requirements the system in this project is aiming to fulfil. It must:

- Allow users to compose and publish tweets whilst suggesting hashtags relevant to the content of their tweets.
- Allow users to search for a hashtag and view related tweets, including those that don't contain that hashtag.

3.1.1 Functional Requirements

1. The system must allow the user to log in and publish tweets to their Twitter account.
2. The system must provide hashtag recommendations as the user is creating a tweet.
3. The system must perform a hashtag search through a large dataset of tweets and return all relevant tweets, including those that do not contain the search query.
4. The system must use information from a large dataset of tweets to generate a model representing each hashtag.
5. The system must be able to compare tweets against its representational hashtag models.
6. *Optional:* The system must be able to update its classification models using information from the live Twitter stream.
7. *Optional:* The system must provide probabilities for how likely a hashtag is to be related to a tweet.

3.1.2 Non-Functional Requirements

1. The system must be accessible via a web interface.
2. The system must be responsive and easy to use.
3. The system must be able to perform searches quickly.

4. The system must be able to make hashtag recommendations quickly.
5. The system must be able to produce visualisations to provide an easy way to interpret the hashtag recommendations/assignments.
6. *Optional:* The system must be accessible via mobile web browsers.

3.2 Scraping Tweets

Twitter offers two APIs with which to access users' tweets: a REST API[7] and a Streaming API[8]. The REST API offers query-based access to tweets, such as tweets by a specific user, and the Streaming API offers a live-stream of the latest tweets being published on Twitter. To scrape data for the initial tests and analysis, tweets were collecting from the Twitter sample public stream. This is a stream containing a small random sample of all public tweets being made, which makes it possible to get a good sample of the tweets being posted on Twitter on a machine with limited resources.

By using the excellent Twython[9] library, which is a collection of pure Python wrapper functions around the Twitter API calls, 500,000 tweets were collected from the sample Twitter stream over a time period of approximately 4.5 days. The tweets collected were filtered to ensure that they were in English, and contained at least one hashtag. The tweets were stored in a CSV file, with numerous pieces of information about each tweet being stored:

- The tweet ID
- The timestamp of when the tweet was created
- A list of the hashtags contained within the tweet
- Whether the tweet is a retweet of another user's tweet
- The geocoordinates of where the tweet was posted from (if available)
- The ID of the user that posted the tweet
- The location of the user, as declared by the user on their profile
- The timezone the user has set on their profile
- The actual text of the tweet

[7] `https://dev.twitter.com/docs/api/1.1`
[8] `https://dev.twitter.com/docs/streaming-apis`
[9] `https://github.com/ryanmcgrath/twython`

3.3 Analysing the Data

After collecting the data, several small scripts were written to evaluate the tweets and gather some insight into the range and quality of information available across Twitter.

4 Project Management

4.1 Risk Analysis

Every large-scale project has to face a number of uncertain factors that could impact the progress or functionality of the final result. Listed below are some of the issues that may occur during the project.

4.1.1 Health Issues

It is possible that throughout the duration of the project serious illness or health problems may arise, meaning no work can be done on the project for an extended period of time. This risk is unpredictable, and it's difficult to take measures to prevent it from happening. If this situation was to arise, then the only solution would be to catch up on the work missed after recovering from the problem. However to reduce the impact of the illness, the project schedule will be followed closely, and therefore minimising the amount of work to catch-up on.

4.1.2 Technical Issues

The project will be completed using an arsenal of digital tools and programming languages, all of which can produce unexpected technical issues. To protect against this, all work will be stored in a cloud-based git repository, hosted on the GitHub[10] service. This will provide both a cloud-based back-up solution, as well as enabling the facility to roll back through past modifications to the project, if an issue should occur.

4.1.3 Scheduling Issues

Throughout the duration of the project, other University modules will be taking place and setting their own courseworks, exams and deadlines. This may place additional strain upon the project, and result in falling behind on the project schedule. To combat this, the project schedule has been carefully planned to balance the workload between the project and other modules throughout the academic year. As the project progresses, the Gantt chart will be updated frequently to monitor the progress and ensure the project stays on track.

[10] `www.github.com`

4.1.4 Unrealistic Goals

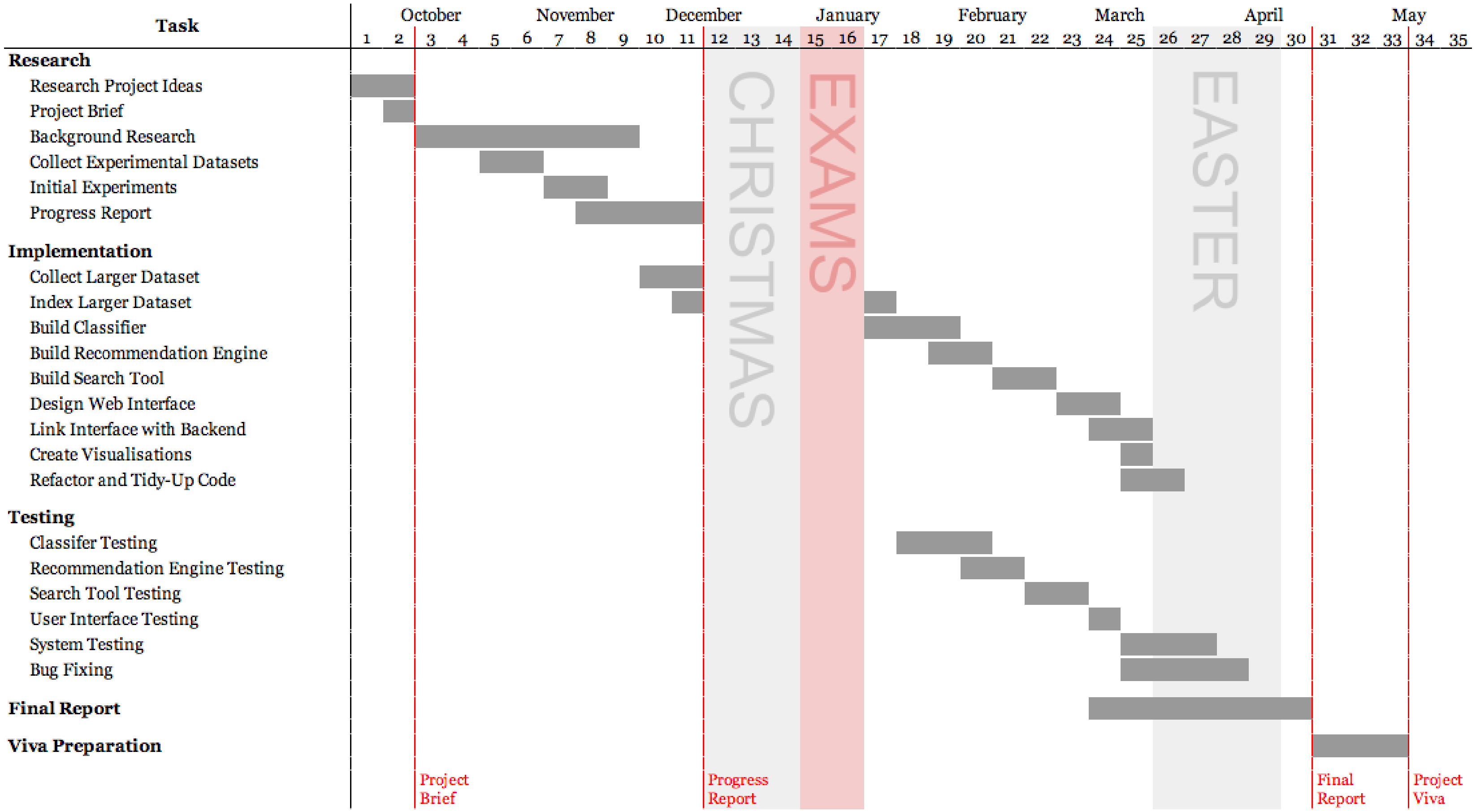
Through the long-term planning and foresight required to plan a large-scale project, it is possible that the goals and functionality of the project may be more difficult to achieve than originally thought. In this case, the functionality of the project will be simplified to ensure that a core subset of the goals and features can still be achieved, in coordination with the project supervisor.

4.2 Project Planning

- Talk about agile development, testing and programming at the same time
- Talk about conflicting deadlines/exams, no work scheduled for Easter
- Lighter workload in Semester 1 due to high volume of coursework (IA, Graphics, Management, Scripting)
- Easter is just bug fixing and report stuff, due to revision for summer exams.

A Project Gantt Chart

This is a Gantt chart showing the scheduled progression through the different aspects of the project.



References

Cantador, I., Bellogín, A. and Vallet, D. (2010). 'Content-based Recommendation in Social Tagging Systems'. *Proceedings of the Fourth ACM Conference on Recommender Systems*. RecSys '10. Barcelona, Spain: ACM, pp. 237–240.

De Choudhury, M., Diakopoulos, N. and Naaman, M. (2012). 'Unfolding the Event Landscape on Twitter: Classification and Exploration of User Categories'. *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*. CSCW '12. Seattle, Washington, USA: ACM, pp. 241–244.

Goldberg, D., Nichols, D., Oki, B. M. and Terry, D. (1992). 'Using Collaborative Filtering to Weave an Information Tapestry'. *Communications of the ACM* 35.12, pp. 61–70.

Kywe, S. M., Hoang, T.-A., Lim, E.-P. and Zhu, F. (2012). 'On Recommending Hashtags in Twitter Networks'. *Proceedings of the 4th International Conference on Social Informatics*. SocInfo '12. Lausanne, Switzerland: Springer-Verlag, pp. 337–350.

Mazzia, A. and Juett, J. (2011). 'Suggesting Hashtags on Twitter'. Computer Science & Engineering, University of Michigan. URL: `http://www-personal.umich.edu/~amazzia/pubs/545-final.pdf`.

Pazzani, M. J. and Billsus, D. (2007). 'The Adaptive Web'. Ed. by P. Brusilovsky, A. Kobsa and W. Nejdl. Vol. 4321. Berlin, Heidelberg: Springer-Verlag. Chap. Content-based Recommendation Systems, pp. 325–341. ISBN: 978-3-540-72078-2.

Salton, G. and Buckley, C. (1990). 'Improving Retrieval Performance by Relevance Feedback'. *Journal of the American Society for Information Science* 41.4, pp. 288–297.

Schafer, J. B., Konstan, J. A. and Riedl, J. (2001). 'E-Commerce Recommendation Applications'. *Data Mining and Knowledge Discovery* 5.1-2, pp. 115–153.

Shepitsen, A., Gemmell, J., Mobasher, B. and Burke, R. (2008). 'Personalized Recommendation in Social Tagging Systems Using Hierarchical Clustering'. *Proceedings of the 2008 ACM Conference on Recommender Systems*. RecSys '08. Lausanne, Switzerland: ACM, pp. 259–266.

Sriram, B., Fuhry, D., Demir, E., Ferhatosmanoglu, H. and Demirbas, M. (2010). 'Short Text Classification in Twitter to Improve Information Filtering'. *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '10. Geneva, Switzerland: ACM, pp. 841–842.

Utiyama, M. and Yamamoto, M. (2006). 'Relevance Feedback Models for Recommendation'. *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*. EMNLP '06. Sydney, Australia: Association for Computational Linguistics, pp. 449–456.

Yang, L., Sun, T., Zhang, M. and Mei, Q. (2012). 'We Know What @You #Tag: Does the Dual Role Affect Hashtag Adoption?' *Proceedings of the 21st International Conference on World Wide Web*. WWW '12. Lyon, France: ACM, pp. 261–270.

Zangerle, E., Gassler, W. and Specht, G. (2011). 'Recommending #-Tags in Twitter'. *Proceedings of the Workshop on Semantic Adaptive Social Web*. SASWeb '11. Girona, Spain: CEUR Workshop Proceedings, pp. 67–78.