ThemeCrowds: Multiresolution Summaries of Twitter Usage

Daniel Archambault, Derek Greene, Pádraig Cunningham, Neil Hurley
Clique Research Cluster
School of Computer Science & Informatics
University College Dublin, Ireland
{daniel.archambault, derek.greene, padraig.cunningham, neil.hurley}@ucd.ie

ABSTRACT

Users of social media sites, such as Twitter, rapidly generate large volumes of text content on a daily basis. Visual summaries are needed to understand what groups of people are saying collectively in this unstructured text data. Users will typically discuss a wide variety of topics, where the number of authors talking about a specific topic can quickly grow or diminish over time, and what the collective is saying about the subject can shift as a situation develops. In this paper, we present a technique that summarises what collections of Twitter users are saying about certain topics over time. As the correct resolution for inspecting the data is unknown in advance, the users are clustered hierarchically over a fixed time interval based on the similarity of their posts. The visualisation technique takes this data structure as its input. Given a topic, it finds the correct resolution of users at each time interval and provides tags to summarise what the collective is discussing. The technique is tested on a large microblogging corpus, consisting of millions of tweets and over a million users.

Categories and Subject Descriptors

H.5.0 [**Information Interfaces and Presentation**]: General; H.2.8 [**Database Management**]: Data Mining

General Terms

Algorithms

1. INTRODUCTION

With the advent of social media networks such as Twitter, users are able to generate large volumes of text data. There is great interest in tracking the trajectory of topics as new items emerge and the commentary on topics evolves over time [18]. However, visually summarising the scale and topics discussed by groups of users, or **crowds**, has received little attention. Tools that are able to present these summaries at an appropriate level of granularity would not only be able to convey the scale of discussion about a given topic but also reveal some context for how the topic is discussed in the crowd.

SMUC'11, October 28, 2011, Glasgow, Scotland, UK.

An example usage scenario is shown in Fig. 1 on some synthetic data. We aim at answering the question: *Are there several small groups of users discussing different aspects of the topic or a single, large group of users with a common voice?* In this example, a search for the term "obama" in March of 2011 reveals crowds talking about the Libya situation and the 2012 presidential election. The topics discussed in the presidential election clusters shift from Sarah Palin to Mike Huckabee. If the user is only interested in the discussions around Libya, a cluster can be selected, as indicated by the red box in Fig. 1(b), and tracked across the time series. Crowds, which speak substantially more about Libya than Obama, are revealed and the topics they are discussing are clarified by the frequent tags around them. As a situation develops, both the tags and the **crowd resolution**, or appropriate level of granularity, can change. Thus, tools that are able to find both the appropriate crowd resolution and present a "summary" of the types of topics discussed in these crowds are needed so as to better understand the subjects discussed in large volumes of Twitter data.

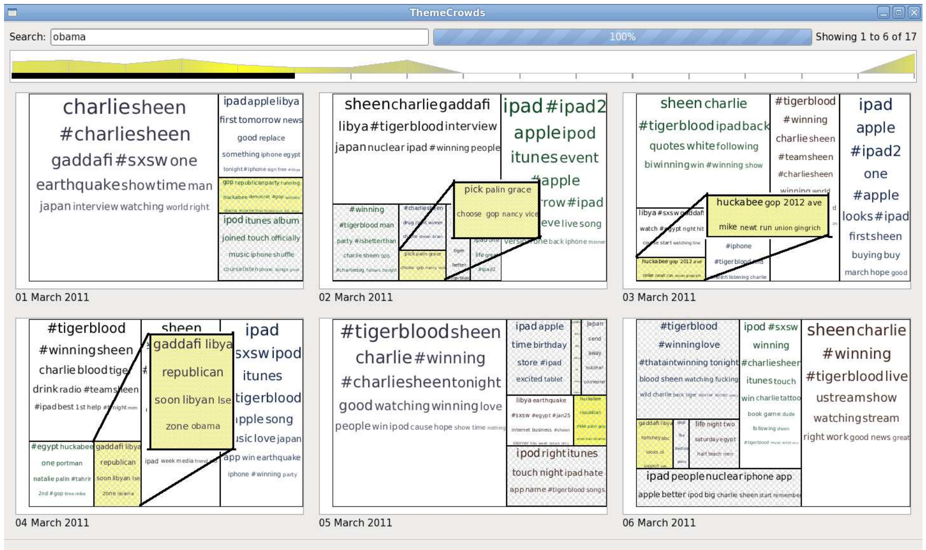
Here we present *ThemeCrowds*, a visualisation system that is able to discover trends, in terms of topics being discussed by clusters of Twitter users, and show how these trends evolve over time. The technique, at each time step, is able to select and present the most appropriate level of resolution based on a novel extension of tag clouds to the multilevel environment. We discuss the application of ThemeCrowds to a large collection of microblogging data. Our results show that it scales beyond the current state-of-the-art visualisation techniques [13, 22, 7], to millions of tweets. Note that an extended version of this paper with further case studies is available as a technical report with the same title [1][1].

2. RELATED WORK

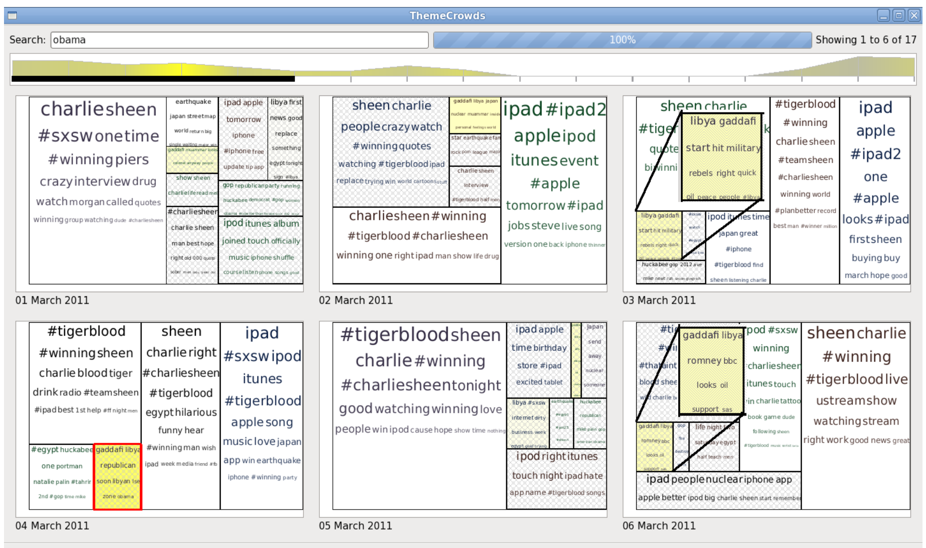
2.1 Dynamic Text Visualisation

A number of systems have looked at how to represent dynamically evolving textual data, often in the context of news stories or social media networks. ThemeRiver [11] encodes the frequency of terms as horizontal streams than grow and shrink over time. Dubinko *et al.* [8] present a method for depicting the evolution of tag clouds, using animation. The tags selected for animation have high "interestingness": a value computed based on tag frequency and variability. Lee *et al.* [17] presented a method that characterises tags and their evolution in terms of frequency, by overlaying spark lines on each tag. Dörk *et al.* [7] visualise conversations in Twitter data using "topic streams" that visually represented as stacked graphs. Their system scales to data sets of over a million tweets and successfully identified conversations in the data.

[1] `http://www.csi.ucd.ie/files/ucd-csi-2011-07.eps`



(a) Search for "Obama"



(b) Tracking the "Libya" crowd

Figure 1: Overview of usage of ThemeCrowds. (a) A general search for the query term "obama". The search reveals many topics across time with some clusters of users, or crowds, pertaining to the uprising in Libya and others pertaining to the 2012 Presidential Election. These distinct topics are uncovered by reading frequently used terms around Obama. (b) The user selects a crowd on 4 March 2011 pertaining to Libya and finds the best match to it over time. Crowds matching the presidential election are filtered out and crowds pertaining to Libya become more prominent. The time step dated 3 March 2011 finds a cluster that may be less focused on Obama but contains enough discussion about Libya to be relevant.

These techniques illustrate the evolution of tags in dynamic text data. However, in our problem, we need to illustrate the dynamic evolution of clusters of Twitter authors at the correct resolution. This problem requires the simultaneous visualisation of cluster content and tag frequency which these systems do not directly support.

2.2 Clustered Text Visualisation

A number of systems have looked at the problem of visualising clusters of documents in order to better understand the content of clustered document collections. IN-SPIRE [27] creates landscapes of documents using dimensionality reduction based on document statistics. FacetAtlas [4] describes a technique to encode entities, their relationships, and classes or clusters of entities. The approach is multilevel and allows users of the system to understand complex correlations across groups of documents at various levels of resolution.

Hetzler *et al.* [12] use animation to depict dynamically evolving document clusters and has facilities to take snapshots of the data over time. Rose *et al.* [21] summarize news stories by clustering them in their most highly associated theme and depicts keywords and stories as they evolve over time. Shi *et al.* [23] combine trend graphs with tag clouds to visualise cluster content and size as it evolves over time.

Although many of these systems handle the evolution of temporal clusters of documents, they do not support the visualisation of topics at multiple levels of resolution. FacetAtlas [4] does support this sort of visualisation, but it is unclear how to extend it to depict the evolution of topics over time. In our problem, we are concerned with summarizing the tweets of groups of users at an appropriate resolution, along with the evolution of their content over time.

2.3 Social Media Analysis

Many researchers have become interested in content diffusion and network structure within the Twitter microblogging services, given the potential for Twitter to facilitate the rapid spread of information. Java *et al.* [13] provided some evidence of Twitter user communities, where the members share common interests as reflected by the terms appearing in their tweets. Kwak *et al.* [16] studied a sample of 41.7 million users and 106 million tweets. The authors studied aspects such as: identifying influential users, information diffusion, and trending topics. Shamma *et al.* [22] performed an analysis on microblogging activity during the 2008 US Presidential Debates. The authors demonstrated that frequent terms reflected the topics being discussed, but informal vocabulary complicated topic identification. Social media analysis has extensively studied large scale Twitter data along with the trends and topics that such data sets contain. However, these works do not investigate visualisation methods to support the exploration of such large volumes of data. With ThemeCrowds, we build a visualisation system to support the display and visual analysis of the most relevant resolution of topic clusters and how they evolve over time.

3. INTERFACE

ThemeCrowds is aimed at depicting the most relevant clusters of users relating to a particular topic, providing an overview of those clusters, in terms of both size and textual content, and depicting how they evolve over time. The technique must scale to millions of tweets, and therefore, some form of summarisation is needed. We choose to perform a multilevel clustering based on the similarity of user tweet profiles for each day.

The proposed visualisation interface for this technique is shown in Fig. 2. The user enters a query term in the search box at the top of the screen. Based on the term at each time step, an appropriate resolution is found and clusters enriched in the term are highlighted in yellow. The results are depicted in the multiples view of six **multilevel tag clouds**. **Small multiples** [24] places many individual time steps on the screen at the same time with each in its own window. Recent experiments [20] seem to suggest that small multiples can be effective for the display of dynamic data. A **scented scroll bar widget** shows the volume and relevance of the tweets that match a given topic.

3.1 Multilevel Tag Cloud

The multilevel tag cloud, at each time step, depicts an appropriate crowd resolution. The main purpose of the widget is to convey crowd size and content compactly for a given time step.

An intuitive method for depicting cluster size and content simultaneously is through a treemap[2][3] [14]. In our context, each crowd is represented as a node in the hierarchy and its size is set proportional to the number of users present in the cluster. In order to effectively use space, we embed tag clouds inside the nodes of the

[2] http://www.smartmoney.com/map-of-the-market

[3] http://newsmap.jp

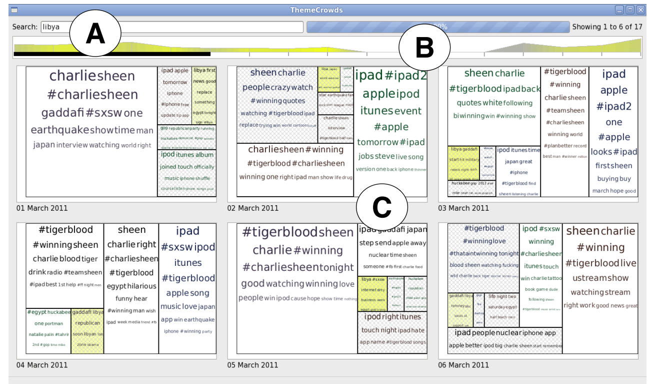


Figure 2: Main components of the ThemeCrowds interface: (A) Search box for entering a query term. (B) Scroll bar scented widget that depicts the relevance and scale of the topic over time. (C) Small multiples matrix of multilevel tag clouds.

hierarchy. The font size depends both on the term importance in the cluster and the size of the node to maximise readability. Informed by Rivadeneira *et al.* [19], tags are ordered by frequency as there is evidence that this ordering is effective for discerning the topics discussed by the cloud.

The user can interact with the multilevel tag cloud in many ways. Clicking on a crowd navigates to deeper levels of the hierarchy. Shift clicking coarsens the resolution to the parent node. The user can select crowds and track their best matches over time. Details views can be brought up for each individual time step in order to examine or compare time steps that are far apart in the series.

3.2 Scroll Bar Scented Widget

This scented widget [26] is a line graph and a scroll bar simultaneously. The line graph encodes time along the x-axis and the volume of tweets about the given topic along the y-axis. The colour at a given time encodes the relevance with respect to the topic discussed with more saturated yellow indicating greater relevance.

4. IMPLEMENTATION

ThemeCrowds takes a time series of multilevel clusterings of Twitter users as its input. Three files are associated with each node of the cluster tree: a tag cloud file with a list of tags and their weights, a file which stores a complete or representative list of data items (*e.g.* individual or aggregated sets of tweets), and a file which lists the size of each cluster in the hierarchy in terms of the number of users.

4.1 Clustering

To generate the actual cluster trees, a variety of methods could be used, such as standard agglomerative or divisive hierarchical clustering algorithms or manual construction. Due to the size of the data sets used in the results presented in this paper, we employ a scalable version of min-max linkage agglomerative clustering [6] for the experiments described in section 5. The approach makes use of a problem decomposition strategy to split large data sets for initial clustering as originally proposed in [5]. This algorithm allows us to generate cluster trees for sequences of data sets containing hundreds of thousands of items. A complete description of the algorithm is provided in the associated technical report [1].

Unlike in most text mining tasks, tweets are limited to 140 characters. In addition, the informal vocabulary used on Twitter makes

(a) hierarchy (b) antichain A (c) antichain B (d) antichain C

Figure 3: Tree map metaphor and antichain selection. (a) Hierarchy that this hierarchical tag cloud represents. (b) Antichain A: the antichain consists of the tag cloud associated with the root of the hierarchy. (c) Antichain B: the antichain after the root has been opened. (d) Antichain C: the antichain after the black node in antichain B has been opened. Nodes can be opened by clicking and closed by shift clicking.

the identification of topics difficult [22]. In order to cluster users based on the content of their tweets, we follow the user-centric approach of [10]: for each user, we create a single **user profile** document which is the aggregation of all their tweets for that time step. Therefore, for the evaluations described later in section 5, the scalable clustering algorithm is applied to the full set of user profiles at each time step to generate the cluster trees.

4.2 Multilevel Tag Cloud

For the remainder of this section, we focus on the visual representation and interaction techniques associated with the tool. We use an implementation [2] of the squarified treemap algorithm [3] to implement the multilevel tag cloud. In order to precisely define how we select the appropriate crowd resolution, we need to first introduce some terminology.

A **maximal antichain** of a hierarchy is a set of nodes that cuts all paths to the root of the hierarchy exactly once. As we deal only with maximal antichains in this paper, we refer to them as **antichains**. Antichains have been used extensively for the purposes of information visualisation and graph visualisation [25, 9] to show or hide details. Since the antichain contains only one node for every path, details are shown for nodes above the antichain and hidden for nodes below it - see Fig. 3. In this case, we show or hide antichains in a similar way to the DagMap [15], but instead of navigating on a level-per-level basis, we allow nodes at different levels to be shown (Fig. 3(d)).

Antichains are used to specify the crowd resolution. When a node is on the antichain, it is opaque and displays a tag cloud with the number of terms displayed proportional to the space available. A shift in resolution corresponds to a shift in antichain. When passing to a finer resolution, the node is shifted above the antichain and all its children are shifted onto it. When passing to a coarser resolution, the parent of a node is placed on the antichain and all the parent's children shift below it. **Leaves** of the hierarchy are indicated using a grey chain-link pattern as shown in Fig. 4. They have no finer resolutions.

We need to distribute tags inside nodes appearing on an antichain and make a simplifying assumption that tag size does not vary too greatly. Given n tags and a node of the treemap with width w and height h, an average tag width and height of w_a and h_a, and aspect ratio $a = \frac{w}{h}$, we assume $n \propto wh$, or a uniform distribution of tags across the rectangle, to scale the size of each tag up by a factor of:

$$\min\left(\frac{w}{w_a\sqrt{na}}, \frac{h}{h_a\sqrt{\frac{n}{a}}}\right)$$

(a) leaf node (b) internal node

Figure 4: Encoding for a leaf node of the graph hierarchy. (a) Chain-link pattern in the background indicates that this node of the tree map is a leaf of the hierarchy. (b) A node that is not a leaf of the hierarchy with no chain-link pattern.

We place tags from top left to bottom right in frequency order, scaling if the word does not fit the required area.

4.3 Automatic Antichain Selection

After a search term is entered or a crowd is selected, the user can find crowds that are enriched in that term. However, these interesting nodes may be buried deep inside the hierarchy at various levels. Our approach for automatic antichain selection adjusts the antichain to display the most relevant matching resolution.

We place a node on the antichain if it is of shallowest possible depth that roots a subtree whereby it has the best match score of all of its descendants. After a term is entered or a cluster is selected, the approach begins by performing a depth first search of all nodes in all hierarchies and computes a match score $[0,1]$ for each node. If the match score is based around a selection, cosine similarity compares the tag frequencies of the selected node to the internal nodes in each hierarchy. If a search term is entered, the score is the ratio of the frequency of the term in the internal node compared to the maximum frequency for that term in the data set.

After a match score has been assigned to each node, the antichain is moved so that it highlights the best possible matches. Fig. 5 provides pseudo-code for this procedure and Fig. 6 gives of an example of an antichain computed on a hierarchy where all nodes have been assigned match scores. The algorithm examines the match scores of each node bottom-up from the leaves of the hierarchy. During this traversal, a node r, with match score r_v, subtends a subtree. If

```
double findMaxAntichain (r)
  m_v ← −1
  for ∀c ∈ children of r do
    c_v ← findMaxAntichain (c)
    if (c_v > m_v) then
      m_v ← c_v
    end if
  end for
  if (m_v < θ and r_v < θ) or (r_v > m_v) then
    coarsen antichain to r
    return r_v
  else
    return m_v
  end if
```

Figure 5: Algorithm to find the best matching antichain. The match score for each node in the tree is computed beforehand and is supplied as input. The current root of the subtree is r and its match score is r_v. The maximum match score for any node in the subtree rooted at r is m_v. The value θ is the match threshold (everything below θ is considered as zero). All nodes present on the antichain are the crowds of coarsest resolution that have maximal match scores when compared to all nodes in the subtrees they subtend.

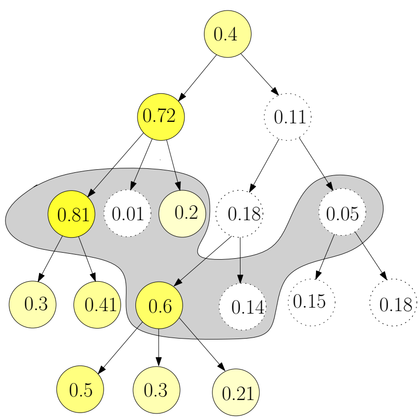


Figure 6: Method for automatic maximal antichain selection with a threshold of 0.2. Nodes with matches above the threshold are coloured yellow with saturation proportional to degree of match. Nodes below the threshold are white with dotted borders. After a score has been assigned to each node, the antichain is lowered automatically to reveal the best matching antichain. A node is on the antichain if it is a node of shallowest possible depth that roots a subtree whereby it has the best match score of all of its descendants.

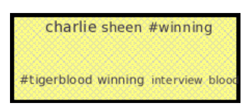
r_v is larger than all of the match scores of the nodes in the subtree it subtends, r is placed on the antichain. The node r can also be placed on the antichain if r_v is below a match threshold of θ and if all of the nodes in the subtree it subtends also have a match score less than θ. In both cases, the value r_v is returned for this subtree. The first condition ensures the closest match is placed on the antichain while the second condition ensures that if there is no match, the coarsest resolution is placed on the antichain. If neither condition is met, r is not placed on the antichain and the value m_v is returned for the subtree. In the current implementation of our technique, the default parameter value is $\theta = 0.20$, which was determined empirically after trials on several Twitter subsets.

5. CASE STUDY

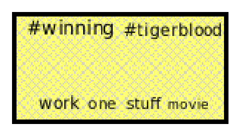
As a case study, ThemeCrowds is applied to a microblogging corpus with the goal of identifying groups of users within a large geographical area, who discuss similar topics over time. We make no prior assumptions on what users might be discussing, and do not filter or constrain the data beyond broad geographical and linguistic limits. The corpus was collected during March 1–17 2011, by retrieving all tweets available from the Twitter streaming API produced by users located in eight US cities: Boston, Chicago, Houston, Los Angeles, Miami, New York, San Francisco, and Philadelphia. Tweets marked as English language were kept, although we observed that the language classification was often inaccurate. We also removed Twitter usernames and URLs. No further filtering was performed on the data. This resulted in 2,200,138 tweets produced from 135,032 unique users over the 17 day period. We applied the scalable min-max agglomerative clustering algorithm to the resulting user profile documents for each 24 hour time step. The data was randomly divided into $p = 5$ fractions to seed the algorithm, and the maximum number of leaf nodes in the hierarchy was set to 50. The clustering process took an average of 438 seconds per time step.

Exploring the small multiples matrix at a high level also reveals the presence of several frequently-appearing hashtags, whose meaning may not be immediately apparent. An example is the cryptic hashtag "#tigerblood", which appears in the data on 2 March 2011. Inspecting the terms in clusters containing this hashtag (see Fig. 7) indicate that it signifies Twitter users discussing actor Charlie Sheen, who joined Twitter on March 1st after a television interview, and had gained one million followers within 24 hours (the fastest in Twitter's history). By tracking the first cluster containing this hashtag, we see many crowds that continue to use this and co-occurring hashtags (*e.g.* "#winning") in their tweets for a number of days after its initial emergence.

A second topic which trended on Twitter in early March was the Rebecca Black Internet meme. Fig. 8 shows two example clusters of users discussing the singer. In the March 14th cluster, a number of users seem to be speaking both about this Internet meme and the Charlie Sheen situation – possibly comparing them. On March

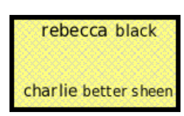


(a) 2 March



(b) 5 March

Figure 7: Excerpts of multilevel tag clouds for two time steps, representing clusters of users using the hashtag "#tigerblood" in their tweets.



(a) 14 March

(b) 17 March

Figure 8: Excerpts of multilevel tag clouds for two time steps, representing clusters of users "rebecca" in their tweets. These groups of users seem to be talking about the Rebecca Black Internet meme.

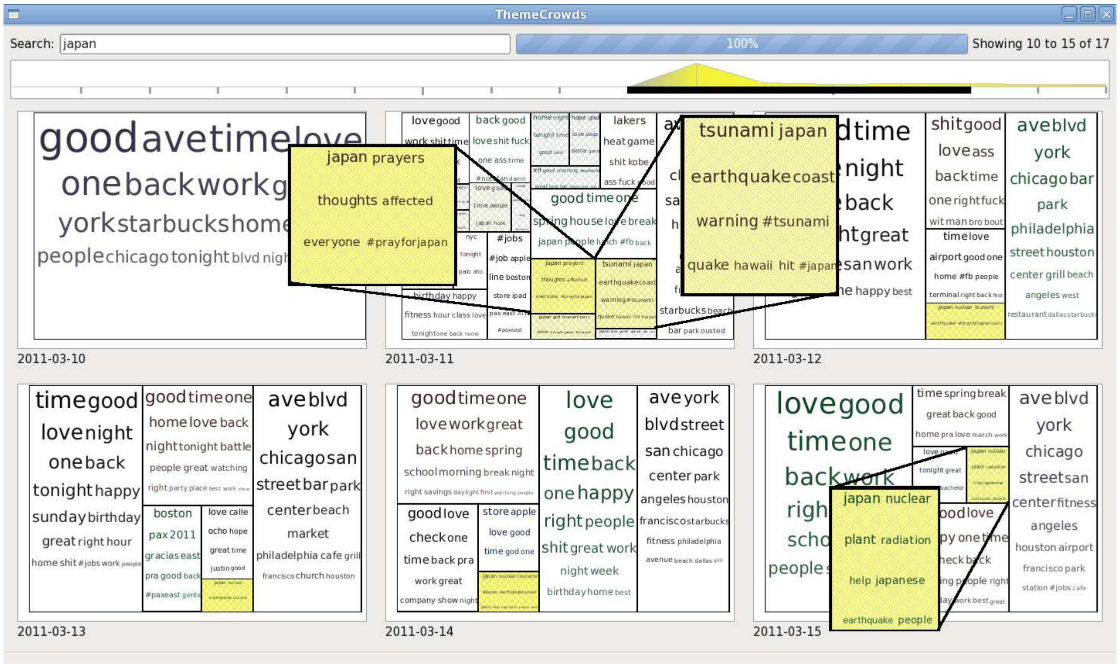
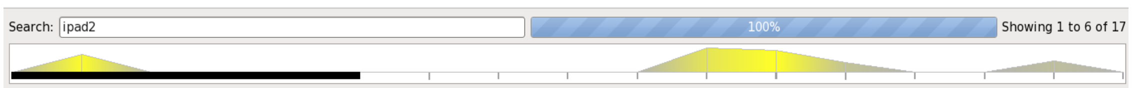


Figure 9: Overlay showing the development of discussions on Twitter after the earthquake and tsunami in Japan on 11 March 2011.



(a) Scented widget for query '#ipad2'

(b) Details view for 2 March 2011

(c) Details view for 11 March 2011

Figure 10: When searching for instances of the hashtag "#ipad2", (a) shows the relevant scented widget showing two distinct periods of activity, with (b) and (c) showing the details views for the corresponding two time steps.

17th, there is a larger group of users who are discussing what they think about the meme. Through ThemeCrowds, the types of language used by groups of users when discussing various memes can be better understood.

One notable application of ThemeCrowds is the identification of emerging topics and trends being discussed by communities on Twitter. On 11 March 2011, we observe at the root node level in the hierarchy that the term "japan" appears. After a search for this term, the scented widget of Fig. 9 reveals that it is not prominent in the data set prior to this date. In the multilevel tag clouds, ThemeCrowds reveals the development of discussions on Twitter surrounding the earthquake and tsunami. Initially on 11 March 2011, we see two distinct types of discussion on Twitter – one cluster consisting of an out-pouring of sentiment regarding the disaster (frequently accompanied by the "#prayforjapan" hashtag), while another cluster pertains to factual items, such as news reports and tsunami warnings. As the story develops, discussion around the topic shifts from “earthquake” and “tsunami” to “nuclear” and “radiation” which did not appear previously.

As well as finding emerging discussion around events, ThemeCrowds also allows users to identify groups of Twitter members discussing intermittent events. As an example, we observed that on 2 March 2011 the tag “#ipad2” was prominent at the upper levels of the hierarchy. After searching for this hashtag, Fig. 10(a) shows discussion activity around this hashtag in two distinct time periods. Fig. 10(b) shows the details view for 2 March 2011, where a homogeneous cluster is highlighted – the terms around the hashtag indicate that this group pertains to the announcement of the Apple iPad 2 by Steve Jobs. Later on 11 March 2011, Fig. 10(c) shows another highlighted cluster where users are discussing the iPad 2. However, here the terms around the hashtag suggest that tweets are related to people waiting in line to buy the product from the Apple Store.

ThemeCrowds can also be used to uncover the multiple different ways a topic is discussed on a given day. To illustrate this capability, we show the results for the search “patrick” on 17 March 2011. In this case, three distinct clusters emerge. One of these clusters contains users tweeting St. Patrick’s Day wishes to other users. A second speaks more about St. Patrick’s Day in New York City and a parade occurring there. Another cluster contains users tweeting about what they are wearing/doing on the day. Closer to the bottom of the display, a fourth cluster speaks about St. Patrick’s Day events in Boston.

6. CONCLUSIONS

The primary contribution of this work has been the development of techniques to visualise groups of Twitter users based on the topics they discuss and track their progression over time, through a range of interactive techniques. The algorithm introduces a novel method for automatic antichain selection and extends tag clouds to a multilevel setting in order to select the appropriate crowd resolution. ThemeCrowds was tested on a large Twitter corpus containing over two million tweets, where we employed the technique to identify discussions in the data that persisted over time at different levels of granularity. Although our primary use case was microblogging data, the technique is data-agnostic and can be applied to any time series data collection with a textual representation and a hierarchical categorisation.

Currently, ThemeCrowds is not able to visualise information relating to sentiment associated with topics being discussed on Twitter. In future work, we plan to experiment with using sentiment, rather than topic matching, to determine the appropriate level of resolution in multilevel clusterings of social media users. We also intend to integrate other kinds of metadata provided by Twitter into the visualisation system, such as geospatial information associated with tweets.

(a) Search results for “patrick”

Figure 11: Search results for “patrick” on 17 March 2011. Four distinct relevant clusters are visible: Happy St. Patrick’s day wishes, a parade in New York City, what people are doing/wearing, and events taking place in Boston.

7. ACKNOWLEDGEMENTS

This work is supported by Science Foundation Ireland Grant No. 08/SRC/I140. (Clique: Graph and Network Analysis Cluster).

8. REFERENCES

[1] D. Archambault, D. Greene, J. Hannon, P. Cunningham, and N. Hurley. ThemeCrowds: Multiresolution summaries of twitter usage. Technical Report UCD-CSI-2011-07, School of Computer Science & Informatics, UCD, Ireland, June 2011.

[2] D. Auber. Tulip : A huge graphs visualization framework. In P. Mutzel and M. Junger, editors, *Graph Drawing Software*, Mathematics and Visualization, pages 105–126. Springer-Verlag, 2003.

[3] D. M. Bruls, C. Huizing, and J. J. van Wijk. Squarified treemaps. In *Joint Eurographics IEEE TCVG Symp. on Visualization (VisSym 2000)*, pages 33–42, 2000.

[4] N. Cao, J. Sun, Y. Lin, D. Gotz, S. Liu, and H. Qu. FacetAtlas: Multifaceted visualization for rich text corpora. *IEEE Trans. on Visualization and Computer Graphics (InfoVis 2010)*, 16(6):1172–1181, 2010.

[5] D. R. Cutting, J. O. Pedersen, D. Karger, and J. W. Tukey. Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections. In *Proc. 15th Int. Conf. on Research and Development in Information Retrieval*, pages 318–329, 1992.

[6] C. Ding and X. He. Cluster merging and splitting in hierarchical clustering algorithms. In *Proc. IEEE International Conference on Data Mining (ICDM’02)*, pages 139–146, 2002.

[7] M. Dörk, D. Gruen, C. Williamson, and S. Carpendale. A visual backchannel for large-scale events. *IEEE Trans. on Visualization and Computer Graphics (InfoVis 2010)*, 16(6):1129–1138, 2010.

[8] M. Dubinko, R. Kumar, J. Magnani, J. Novak, P. Raghavan, and A. Tomkins. Visualizing tags over time. *ACM Trans. on the Web*, 1(2):Article 7, 2007.

[9] N. Elmqvist and J.-D. Fekete. Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines. *IEEE Trans. on Visualization and Computer Graphics*, 16(3):439–454, 2010.

[10] J. Hannon, M. Bennett, and B. Smyth. Recommending twitter users to follow using content and collaborative filtering approaches. In *Proc. 4th ACM Conference on Recommender Systems (RecSys’10)*, pages 199–206, New York, NY, USA, September 2010. ACM.

[11] S. Havre, E. Hetzler, P. Whitney, and L. Nowell. ThemeRiver: Visualizing thematic changes in large

document collections. *IEEE Trans. on Visualization and Computer Graphics*, 8(1):9–20, 2002.

[12] E. G. Hetzler, V. L. Crow, D. A. Payne, and A. E. Turner. Turning the bucket of text into a pipe. In *IEEE Symp. on Information Visualization (InfoVis 2005)*, pages 89–94, 2005.

[13] A. Java, X. Song, T. Finin, and B. Tseng. Why we twitter: understanding microblogging usage and communities. In *Proc. 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 56–65. ACM, 2007.

[14] B. Johnson and B. Shneiderman. Treemaps: A space-filling approach to the visualization of hierarchical information structures. In *Proc. of IEEE Visualization (Vis 1991)*, pages 275–282, 1991.

[15] P. Y. Koening, G. Melançon, C. Bohan, and B. Gautier. Combining DagMaps and sugiyama layout for the navigation of hierarchical data. In *Proc. of the 11th International Conference on Information Visualization (IV 2007)*, pages 447–452, 2007.

[16] H. Kwak, C. Lee, H. Park, and S. Moon. What is Twitter, a social network or a news media? In *Proc. 19th international conference on World wide web*, pages 591–600. ACM, 2010.

[17] B. Lee, N. H. Riche, A. K. Karlson, and S. Carpendale. SparkClouds: Visualizing trends in tag clouds. *IEEE Trans. on Visualization and Computer Graphics (InfoVis 2010)*, 16(6):1182–1189, 2010.

[18] J. Leskovec, L. Backstrom, and J. Kleinberg. Meme-tracking and the dynamics of the news cycle. In *Proc. 15th International Conference on Knowledge Discovery and Data mining*, pages 497–506, 2009.

[19] A. W. Rivadeneira, D. M. Gruen, M. J. Muller, and D. R. Millen. Getting our head in the clouds: toward evaluation studies of tagclouds. In *Proc. SIGCHI conference on Human factors in computing systems (CHI '07)*, pages 995–998, 2007.

[20] G. Robertson, R. Fernandez, D. Fisher, B. Lee, and J. Stasko. Effectiveness of animation in trend visualization. *IEEE Trans. on Visualization and Computer Graphics (InfoVis '08)*, 14(6):1325–1332, 2008.

[21] S. Rose, S. Butner, W. Cowley, M. Gregory, and J. Walker. Describing story evolution from dynamic information streams. In *IEEE Symp. on Visual Analytics Science and Technology*, pages 99–106, 2009.

[22] D. Shamma, L. Kennedy, and E. Churchill. Tweet the debates: Understanding community annotation of uncollected sources. In *Proc. 1st SIGMM workshop on Social media*, pages 3–10. ACM, 2009.

[23] L. Shi, F. Wei, S. Liu, L. Tan, X. Lian, and M. X. Zhou. Understanding text corpora with multiple facets. In *IEEE Symp. on Visual Analytics Science and Technology*, pages 99–106, 2010.

[24] E. Tufte. *Envisioning Information*. Graphics Press, 1990.

[25] T. von Landesberger, A. Kuijper, T. Schreck, J. Kohlhammer, J. J. van Wijk, J.-D. Fekete, and D. W. Fellner. Visual analysis of large graphs. In *EG 2010 - State of the Art Reports*, pages 37–60, 2010.

[26] W. Willett, J. Heer, and M. Agrawala. Scented widgets: Improving navigation cues with embedded visualizations. *IEEE Trans. on Visualization and Computer Graphics*, 13(6):1129–1136, 2007.

[27] J. A. Wise, J. J. Thomas, K. Pennock, D. Lantrip, M. Pottier, A. Schur, and V. Crow. Visualizing the non-visual: Spatial analysis and interaction with information from text documents. In *Proc. of the IEEE Symp. on Information Visualization*, pages 51–58, 1995.