A Method for Building Burst-Annotated Co-Occurrence Networks for Analysing Trends in Textual Data

Yutaka Mitsuishi[1], **Vít Nováček**[2], **Pierre-Yves Vandenbussche**[1]
[1]Fujitsu (Ireland) Limited, [2]Insight @ NUI Galway (formerly known as DERI)
Firstname.Lastname@ie.fujitsu.com, Firstname.Lastname@deri.org, Firstname.Lastname@ie.fujitsu.com

Abstract

This paper presents a method for constructing a specific type of language resources that are conveniently applicable to analysis of trending topics in time-annotated textual data. More specifically, the method consists of building a co-occurrence network from the on-line content (such as New York Times articles) that conform to key words selected by users (e.g., 'Arab Spring'). Within the network, burstiness of the particular nodes (key words) and edges (co-occurrence relations) is computed. A service deployed on the network then facilitates exploration of the underlying text in order to identify trending topics. Using the graph structure of the network, one can assess also a broader context of the trending events. To limit the information overload of users, we filter the edges and nodes displayed by their burstiness scores to show only the presumably more important ones. The paper gives details on the proposed method, including a step-by-step walk through with plenty of real data examples. We report on a specific application of our method to the topic of 'Arab Spring' and make the language resource applied therein publicly available for experimentation. Last but not least, we outline a methodology of an ongoing evaluation of our method.

Keywords: Trend Analysis; Keyword Bursts; Co-Occurrence Networks

1. Introduction

The web has become a primary venue for sharing information in the last decade, which increasingly affects also on-line availability of all sorts of textual content. With the virtually instant and very cheap digital means for global dissemination of text, the problem of information overload has become paramount to any attempt for making sense out of all the information hypothetically available. Our paper focuses on one particular way of alleviating the information overload - analysis of prevalent trends in on-line textual content related to a specific topic (such as 'Arab Spring' to give a recent example of major global news).
Our approach consists of the following high-level steps:

1. automated download of articles (and corresponding meta-data like issue date) that correspond to a topic specified by the user;

2. extraction of co-occurrence networks from monthly snapshots of the downloaded articles, using the tool described in (Nováček and Burns, 2013);

3. computation of the burst scores of vertices (keywords) and edges (co-occurrence relations) in the co-occurrence networks, using a method combining the approaches of (Yao et al., 2012) and (Kleinberg, 2002);

4. presentation of the results via a service that allows for exploration of the context of trending phenomena in the articles by means of: (1) searching for trending keywords and edges, (2) browsing of their burst-annotated co-occurrence network context filtered only to the most important (i.e., bursty) vertices and edges, (3) retrieval of the articles related to the displayed trending content to provide a complete provenance context as well.

We believe that the key technical advantages of our method when compared to related state of the art are two-fold. Firstly, we use point-wise mutual information scores instead of mere frequencies when computing the burst scores, which is supposed to improve the accuracy of the method for the co-occurring phenomena. Secondly, we offer contextual information within the exploration of the bursty phenomena. This facilitates a mixed initiative approach where users have more information available in order to determine the actual reasons for the importance of the bursty phenomena.
The main contribution of the paper is a method for automated building of language resources (i.e., burst score-annotated co-occurrence networks) that can support analysis of trends in mass media content (Section 3.). Furthermore, Section 4.2. describes an application of the method to a specific scenario of exploring bursty phenomena in the news related to Arab Spring. The last contribution is the language resource (i.e., the burst-annotated co-occurrence network) supporting the Arab Spring application (described in Section 4.2. together with the link to the actual file).
In addition to the main contributions, we describe a publicly available service implementing our approach that is currently being developed (Section 4.1.) and will be ready for demonstration by May, 2014. We also present a methodology for an ongoing evaluation of the service (Section 4.3.). The work related to our approach is summarised in Section 2..

2. Related Work

In recent years, the area of event/topic/trend detection has been a hot topic. (Sayyadi et al., 2009) proposed an event detection algorithm using keyword co-occurrence, but did not consider time-varied burstiness. (Diao et al., 2012) adopted Latent Dirichlet Allocation (LDA) considering time-dependant topic distributions for detecting burst

topics, but did not explicitly treat keyword co-occurrence inside a topic. (Fung et al., 2005) grouped multiple bursty keywords extracted in advance as a single bursty event but didn't consider burst co-occurrences. As (Yao et al., 2012) pointed out about existing researches including them, they cannot detect a bursty co-occurrence with no bursty keyword involved. (Mathioudakis and Koudas, 2010) also compiled bursty keywords into groups but didn't take bursty co-occurrences into account.

We use the approach introduced in (Kleinberg, 2002) to compute the burst scores, however, replace the originally used relative frequencies by normalised point-wise mutual information scores (Manning et al., 2008) in case of co-occurrence relationship burstiness. This is expected to be more accurate than the original formula due to additional insight the information-theoretic score brings regarding significance of the co-occurrences. The most relevant work to this paper is (Yao et al., 2012), whose system detects bursty social tags and their co-occurrences in a collaboration tagging site, and constructs bursty tag events by clustering. This paper shows that his approach is also applicable to keyword co-occurrences in textual data.

3. Method Description

The outline of the method is presented in Figure 1. The input textual data is processed by two main modules: (i) preprocessing module and (ii) burst module. The particular steps executed within these modules are described in the following sections. Note that where applicable, we use running examples from a sample deployment of our approach to the domain of 'Arab Spring'.

3.1. Data Insertion

For the initial input, the minimal expectations are English textual data, titles of particular basic units of the data (e.g., news articles or scientific papers) and time stamps for each unit (e.g., publication date of an article). Other types of meta-data like authors and their affiliations, key-words, references, etc., are optional. The data and meta-data are converted to JSON format for representation of structured text-based information. Before the actual insertion, the textual content within the JSON elements corresponding to each data item is processed by a sentence and word tokenizers and the results are stored alongside the original texts for further processing. For the tokenization (as well as for other NLP tasks later on), we use the Natural Language Toolkit[1]. The resulting JSON data are then incorporated in a document store, MongoDB[2]. The document store offers efficient services for retrieving data items corresponding to certain key-words and time intervals, which is very useful in the consequent stages of processing.

Example 1 *The data set for the 'Arab Spring' topic was generated using the New York Times (NYT) API[3], which offers access to snippets from the majority of NYT articles together with their comprehensive meta-data, including the titles, URLs and publication dates. In the data insertion step, we downloaded all the NYT data from the last three years that have been made available via the API and stored it in a MongoDB instance.*

[1] See `http://nltk.org/` for details.

[2] See `http://www.mongodb.org/` for details.

[3] See `http://developer.nytimes.com/docs/read/article_search_api_v2` for details.

3.2. Data Selection

The next step (after the input data have been converted into JSON and stored) is selection of a topical data set on which users wants to run the trend analysis. At that stage, documents are selected from the document store using one or more keywords and a time period selected by the user. The results are split into chronological document sets by user-specified time intervals (such as monthly buckets).

Example 2 *The topic-oriented data set was extracted from the NYT MongoDB instance by searching for all articles relevant to the following keywords:* arab spring, algeria, bahrain, djibouti, egypt, iraq, jordan, kurd, kurdistan, kuwait, lebanon, libya, mauritania, morocco, oman, palestina, palestine, sahara occidental, saudi arabia, somalia, sudan, syria, tunisia, western sahara, yemen. *The publication date was filtered according to the period from January 2011 to June 2013 and the articles were split into buckets corresponding to one-month intervals. This resulted into 22,613 relevant articles that have been distributed in 30 one-month buckets.*

3.3. Keyword Extraction

After the corpus of articles grouped according to time intervals has been prepared in the previous step, each of the buckets is processed by the keyword extraction module. The module uses the Natural Language Toolkit's shallow parsing capabilities in order to extract noun phrases (both single noun terms and compound phrases with head nouns augmented by other nouns or modifiers). The process defines regular expression patterns on part of speech tags that are then applied to the tagged text in order to delimit the noun phrases as candidate keywords.

Example 3 *In one of the NYT articles in our sample data set (see* `http://www.nytimes.com/2012/12/27/opinion/egypts-flawed-constitution.html`*), the first sentence is:* Ideally, a new constitution in Egypt would unite citizens around a consensus vision for their country and set a firm foundation for a democratic transition. *From this sentence, two candidate keywords can be extracted (among others not relevant to the examples mentioned here):* new constitution, Egypt.

3.4. Co-occurrence Calculation

From the potential keywords identified in the selected articles, we build a co-occurrence network for each of the time buckets using the normalised point-wise mutual information scores (Manning et al., 2008). For the candidate keywords x, y, the score is defined as

$$npmi(x,y) = \frac{pmi(x,y)}{-log_2(p(x,y))},$$

where $pmi(x,y) = log_2 \frac{p(x,y)}{p(x)p(y)}$ is the point-wise mutual information. The $p(x,y), p(x), p(y)$ expressions cor-

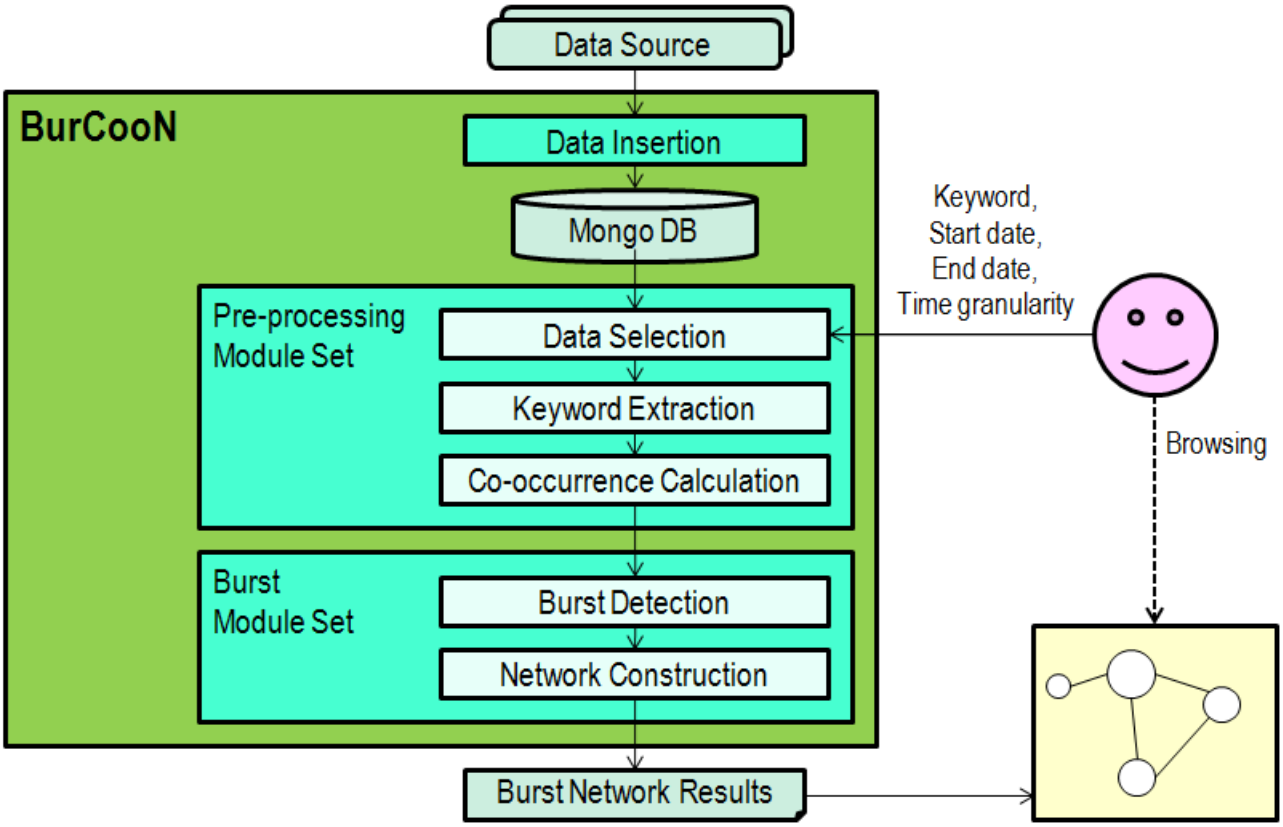

Figure 1: The outline of our approach.

respond to the joint and isolated distributions of x, y, respectively. In our case, the isolated distributions are computed as relative frequencies of the candidate keywords in the bucket texts. The joint distribution of $p(x, y)$ is computed as $\frac{F(x,y)}{N}$, where $F(x, y)$ is an absolute frequency of the cases when x, y occur in the same paragraph in the data, weighted by the distance between the sentences x, y occur in, and N is the total number of all possible combinations of keywords occurring together in the same paragraphs.
Only the keywords with the $npmi$ score higher than a predefined threshold (at least 0 for a non-negative degree of association) are included for further processing, which filters out keyword candidates that do not contribute to the overall co-occurrence networks characteristic to the data.

Example 4 *Based on the distributions of other candidate keywords in the data, the co-occurrence ($npmi$) score of the* (Egypt, new constitution) *tuple is approximately* 0.379935*, which is higher than zero, meaning a positive degree of association between the two terms. Therefore the corresponding keywords and their co-occurrence relationship are to be processed in the consequent steps.*

3.5. Burst Detection

For computing the burst scores, we use the 2-state "batched arrival" automaton defined in (Kleinberg, 2002) to determine if a candidate keyword, in the previously computed co-occurrence network, is bursting[4]. In order to determine if a co-occurrence of keywords is bursting, we use Kleinberg's algorithm while replacing the original σ function by a probability density function of the corresponding normalised point-wise mutual information value multiplied by the frequency. Each keyword and co-occurrence is then associated with one or more tuples in the form: `[burst interval, burst score]`.

Example 5 *The burst detection applied to the two candidates keywords* Egypt *and* new constitution *produces scores* 0 *and approximately* 4.4086*, respectively, for the November 2012 time bucket. The burst score of the* (Egypt, new constitution) *tuple is approximately* 4.6096*. Even though the* Egypt *keyword is not bursting, the co-occurrence of* (Egypt, new constitution) *is itself bursting for the November 2012 time bucket.*

3.6. Network Construction

From the keywords and co-occurrence bursts, we finally compute a burst-network, using the same method as described in (Yao et al., 2012). Vertexes in the network represent bursty keywords and non-bursty keywords involved in a bursty co-occurrence relationship. Edges in the network represent bursty co-occurrence relationships and non-bursty co-occurrence tuples with both of the keywords involved bursting. This means that co-occurrences of Type 1, 2, 3 or 4 in (Yao et al., 2012) will be part of the output network. We exclude Type 5 co-occurrences as they considerably increase the size of the network and the overload of the users.

[4] A keyword is identified as bursting when it is encountered at an unusually high rate.

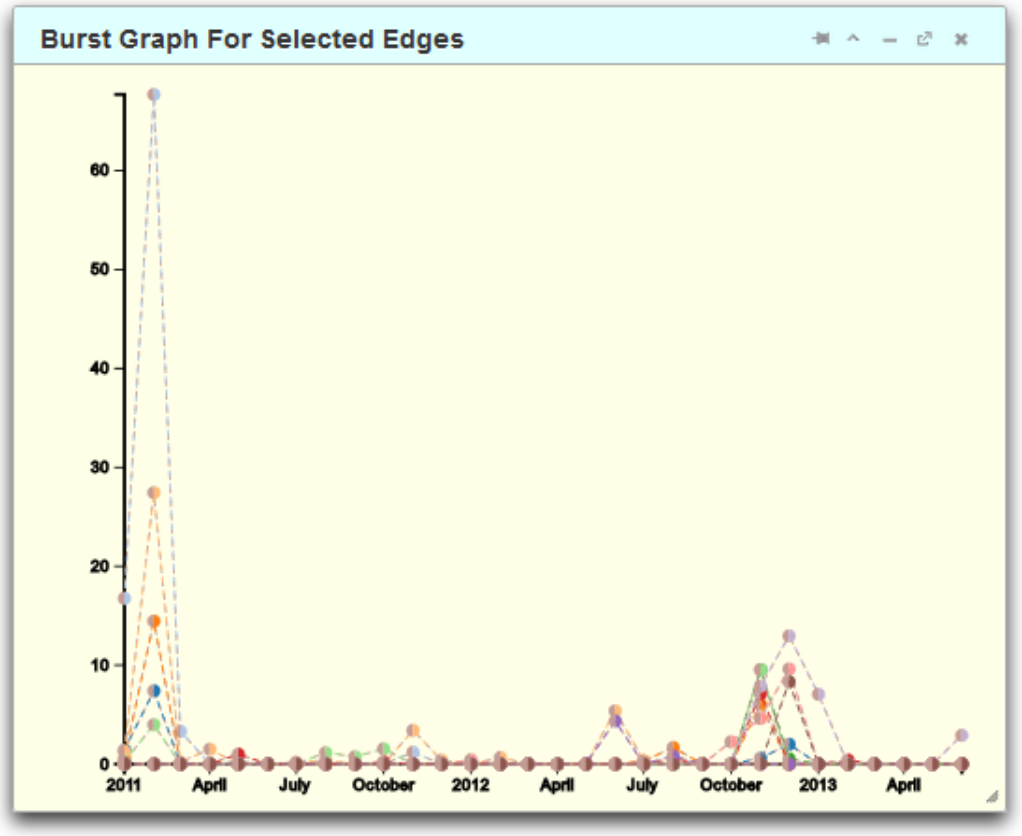

Figure 2: Trending edges for Egypt.

Example 6 *An example of the network generated by the system, which contained the edges composed of the bursty co-occurrence* (Egypt, new constitution)*, is illustrated in figure 3. The flag '1' associated with the* new constitution *vertex means that this keyword is itself bursting whereas the flag '0' associated to the* egypt *vertex means that this keyword is not bursting for the November 2012 time bucket.*

4. Results

In this section we give an overview of the results we have produced so far, namely concerning a service implementing the introduced approach, a specific application of that service and the proposed method for its evaluation.

4.1. The BurCooN Service

The approach presented in this paper has been implemented as a domain independent service that can digest any type of English text and relevant meta-data (title, author, publication date, the actual content) using two JSON files (one with the data and one with configuration telling the system where to look for particular fields). Users can then specify their search parameters as explained in section 3.2.. As a result, BurCooN produces a language resource consisting of a co-occurrence network annotated with burst scores and with provenance information pointing to the original data. By visualisation and exploration of this resource, one can quickly get an overview of the data and important trends within them.

4.2. Arab Spring Scenario

We have applied the BurCooN service to a recent global news topic—Arab Spring—to test our approach on real data and commence its evaluation with sample users. The entry point to the service is a search interface that allows for retrieval of bursting keywords and co-occurrence edges. When entering the keyword *Egypt* to the interface and limiting the search results to co-occurrence relations in year 2012, one can see the overview presented in Figure 2.

The points in the figure are the bursting co-occurrence edges. After hovering over them, one can see details about each point (the keywords associated with the point, the score and the date). The huge spike in February of 2012 in Figure 2 is related to the big news of the former president's Mubarak resignation (the keywords occurring there refer to revolution, protests, resignation, etc.). This is perhaps quite a common knowledge, however, there is also a smaller spike around November 2012, which seems to be related to rather less informative (i.e., more general) keywords like *Egypt, islamist, president mohamed mursi*.

Using our service, one can easily explore a broader context of the November 2012 spike. In Figures 3 and 4, the co-occurrence context related to the points in the spike in November and December, respectively, is displayed.

The full lines in the figures correspond to bursting edges, while the dotted ones connect the merely co-occurring keywords. One can immediately see that the important notions in the context are related to a *new* or *draft constitution, decree, protest*.

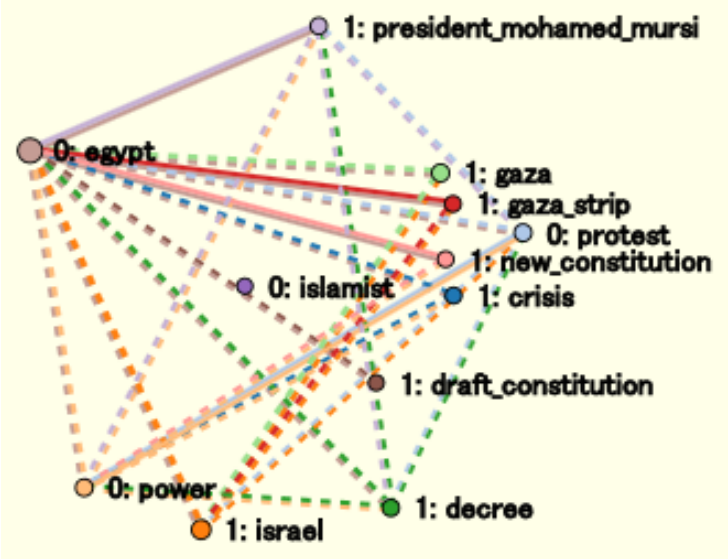

Figure 3: Context in November 2012.

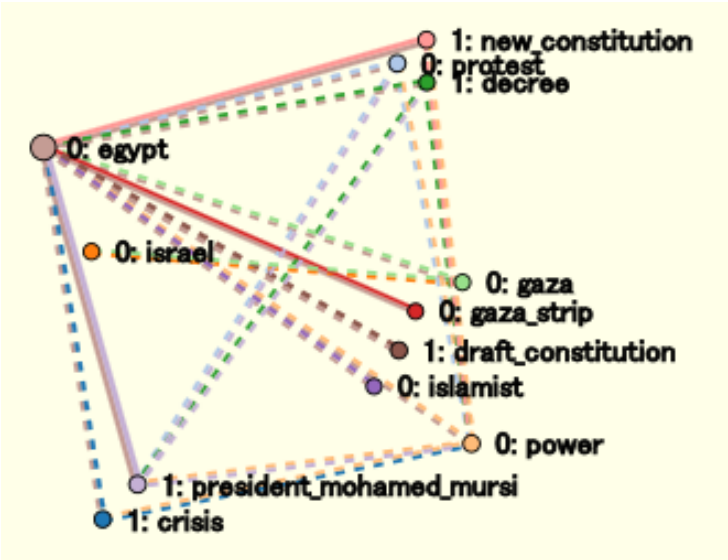

Figure 4: Context in December 2012.

At this stage, it is possible to display additional contextual information – the links to the original articles that are relevant to the currently explored portion of the co-occurrence network. Some of the most relevant NYT articles related to the new constitution, draft constitution, decree keywords are at the following URLs: `http://goo.gl/29aKgJ`, `http://goo.gl/gVZcGB`, `http://goo.gl/qYjtKQ`. By looking at the articles, it is now much more obvious that the spike around November 2012 is related to the president Mursi's decree giving him extended powers within the newly elaborated Egyptian constitution.

Note that the burst-annotated co-occurrence network that supports the application of our method to the Arab Spring domain is available here `http://goo.gl/e1OO8S`. The data are released under the Creative Commons attribution (CC-BY) license and the reader is welcome to experiment with them, modify and share them freely provided an attribution to the authors of this article is made.

4.3. Ongoing Evaluation

We will use the previously introduced service, deployed online, for an evaluation of the approach presented in this paper. The sample users are currently being recruited amongst students and staff of the Insight research centre with the aim of getting at least 20 users for each domain the BurCooN service has been deployed on. The primary method for collecting data from sample users is a web-based questionnaire presenting a random sample of the generated burst data to each user. The samples include both bursting vertices and edges in the co-occurrence networks and are generated so that every result is evaluated by at least three different users to gauge the objectiveness of their judgment. For each result item in the questionnaire, the users are able

to retrieve / study the context (i.e., the neighbourhood in the co-occurrence graph) and the original sources (i.e., NYT articles) so that they can support their judgement using all the available information. Each user is asked to do the following for each result:

- required: mark the result as valid or invalid,
- optional: provide more textual feedback.

After collecting the feedback from the users for each of the sample results, we will use it to compute the average accuracy of the method as follows:

$$accuracy = \frac{1}{|U|} \sum_{u \in U} \frac{|valid(R,u)|}{|R|},$$

where U is the set of sample users, R is the set of results and $valid : 2^R \times U \rightarrow 2^R$ is a function returning a subset of results that are valid among the input set according to the specified user. To determine the reliability of the user annotations, we will compute their inter-annotator agreement using the Fleiss' κ measure (Fleiss, 1971) for multiple annotators.

5. Conclusion and Future Work

In this paper we presented a novel approach to exploring trending topics in textual data sets by means of burst-annotated co-occurrence networks. By using point-wise mutual information scores instead of mere frequencies in the burst computation, we can improve the accuracy of the method for the co-occurring phenomena. Though the evaluation is not yet finalised we have some encouraging results from our preliminary tests with sample users. The service we are using for the evaluation is going to be demoed in parallel with the paper presentation. As a part of our results, we provide access to the language resource supporting a sample application of the method to the Arab Spring domain, the burst annotated co-occurrence network.

The introduced method has been illustrated on a particular application to trend analysis in New York Times news article, but is applicable to any English textual content. Another application domain we are working on is the trend analysis of financial news in the context of our work documented in (Vandenbussche, 2013).

The next steps include a public release of the BurCooN service we describe in the paper, as well as the completion and publication of the evaluation process. We also plan to incorporate more sophisticated methods for abstract bursting event discovery (focusing not only on keyword and their co-occurrence, but conceptual classes of keywords and more abstract semantic relationships between them).

Acknowledgements This paper is an outcome in a collaboration research project[5] between the Digital Enterprise Research Institute (now transformed into an Insight @ NUI Galway research centre) and the Fujitsu group. We are very grateful to IDA Ireland and Science Foundation Ireland, both of which is supporting the project. This publication has also partly emanated from research supported by a research grant from Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289.

[5]Fujitsu Announces Significant Research Programme with DERI at NUI Galway (Press Release): http://www.fujitsu.com/ie/news/pr/2012/fs_20120718.html

6. References

Diao, Q., Jiang, J., Zhu, F., and Lim, E.-P. (2012). Finding bursty topics from microblogs. In *ACL (1)*, pages 536–544. The Association for Computational Linguistics.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378382.

Fung, G. P. C., Yu, J. X., Yu, P. S., and Lu, H. (2005). Parameter free bursty events detection in text streams. In Böhm, K., Jensen, C. S., Haas, L. M., Kersten, M. L., Larson, P.-Å., and Ooi, B. C., editors, *VLDB*, pages 181–192. ACM.

Kleinberg, J. M. (2002). Bursty and hierarchical structure in streams. In *KDD*, pages 91–101. ACM. http://doi.acm.org/10.1145/775047.775061.

Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

Mathioudakis, M. and Koudas, N. (2010). Twittermonitor: trend detection over the twitter stream. In Elmagarmid, A. K. and Agrawal, D., editors, *SIGMOD Conference*, pages 1155–1158. ACM.

Nováček, V. and Burns, G. (2013). Skimmr: machine-aided skim-reading. In Kim, J., Nichols, J., and Szekely, P. A., editors, *IUI Companion*, pages 59–60. ACM. .

Sayyadi, H., Hurst, M., and Maykov, A. (2009). Event detection and tracking in social streams. In Adar, E., Hurst, M., Finin, T., Glance, N. S., Nicolov, N., and Tseng, B. L., editors, *ICWSM*. The AAAI Press.

Vandenbussche, P.-Y. (2013). Empower flexibility via big data: Unleashing constraints in financial analytics. In *XBRL26*. http://archive.xbrl.org/26th/sites/26thconference.xbrl.org/files/BUSI3Pierre-YvesVandenbussche.pdf.

Yao, J., Cui, B., Huang, Y., and Zhou, Y. (2012). Bursty event detection from collaborative tags. *World Wide Web*, 15(2):171–195. http://dx.doi.org/10.1007/s11280-011-0136-2.