Predicting News Story Importance using Language Features

Ralf Krestel
L3S Research Institute
Universität Hannover, Germany
krestel@L3S.de

Bhaskar Mehta
Google Inc.
bmehta@google.com

Abstract

In this age of awareness, people have access to information like never before. Hundreds of newspapers and millions of bloggers present news and their interpretations in an openly accessible manner. With globalization, distant events can have impact on people thousands of miles away. While expert humans can recognize a potentially important piece of news, this is still a difficult problem for an automatic system. Since people are increasingly relying on multiple online sources of information, it is important to support users in filtering news automatically. In this work, we consider the problem of anticipating news story importance, i.e. given a news item, predicting if it will be of interest for a majority of users. Such ranking is currently done manually for newspapers, and we explore automatic approaches and indicative features for the same. Our main conclusion is that importance prediction is a hard problem, and pure textual features are not sufficient for classifiers with 90% accuracy.

1. Introduction

Online news is a fast growing source of information for people around the world. According to surveys by Nielson/NetRatings, online news readership grew at 47% in 2004 and more than 40% in 2005. Importantly, many users have indicated online news as their primary source (47%) compared to conventional news sources like radio (16%), TV (18%) and printed newspapers (12%). *Pew Internet*[1] also reports a robust growth in blog readership, doubling in one year to 32 million at the end of 2005. 5% of responders also reported using RSS feeds, thus combining multiple sources of information; this number is also forecasted at 50% growth for the next few years. All these trends point to impending information overflow, where automatic support for filtering information is important.

Given that users increasingly prefer online sources, mainly due to their 24-hour presence and quick turn around time, people are also keen to learn about important events as soon as they happen. This is especially important for events with economic or political implications. Economics researchers have found a high correlation of news items and changes in stock market prices as a result [4]. Various stock market trading houses have in-house news analysts who model the impact of a news item. For such companies and their employees, filtering important news and predicting if they will become important enough for the market to take notice, is of paramount importance. Electronic news information systems which notice such news immediately are thus highly desirable; input from these systems can further be tied into financial prediction models.

We consider in this work a learning system which identifies crucial factors that make a news item important. Note that such factors may not be directly observed, or be directly mapped to observed signals. In particular, we consider a regression setup which predicts if a news item lies in one of 4 categories of news: *extremely important, highly important, moderately important, and unimportant*. These categories can also be represented on a numerical scale of 1–4. Further our approach is content-based; our goal is to find decisive natural language features which make a newspaper article more important than another. We use several natural language signals extracted from news items, and do not rely on social feedback (e.g. Digg); this is important since user feedback, though very helpful for prediction, is associated with high latency, and sufficient user feedback may not be available when information is still new. In this work, we try to find objective measures for the importance of an article without regarding personal preferences of the user or other biased subjective views.

2. Problem Statement and Description

We consider a news item as a collection of text, with additional attributes for title and source. Consider that news is written in language with vocabulary $\mathcal{V}$ of size

[1] www.PewInternet.com

$M = |\mathcal{V}|$; the i^{th} word in the vocabulary is represented by v_i. We use the standard bag-of-words representation, which compresses a document $\mathcal{D}$ into a vector d, such that

$$\mathsf{d} = \{f_1, f_2, \cdots, f_M\} \ , \tag{1}$$

where f_i is the frequency of the word v_i in document d.

Given a collection $\mathbf{C}$ of N document-vectors (features) $\{\mathsf{d}_1, ..., \mathsf{d}_N\}$, we procure *labels* representing importance from a qualified source. Such labels are quantitative, either continuous (say, between 0 to 1), or discrete (say, 1–4). Thus, we have a supervised regression setup: $\{(\mathsf{d}_1, l_1), ..., (\mathsf{d}_N, l_N)\}$. We are interested in learning an Importance Classifier function $\mathcal{I}$ such that

$$\mathcal{I}(\mathbf{C}) \rightarrow \{l\}^{|\mathbf{C}|} \tag{2}$$

where l is the set of labels. Specifically, we would like to learn a classifier with the lowest classification error over a training set with known labels $\mathbf{L}_{\mathcal{C}}$:

$$\mathcal{I}^* = \underset{\mathcal{I}}{\operatorname{argmin}} \sum_{i \in \mathbf{L}_{\mathcal{C}}} |\mathcal{I}(i) - \mathbf{L}_{\mathcal{C}}(i))|^2 \tag{3}$$

We also explore other classifiers which use features sets other than $\mathcal{C}$; examples include only nouns, verbs, adjectives etc.

Note further that using the kernel trick, we can project the news items into feature space. We use several sets of extracted features and explore the predictive performance of these features. We explain the chosen features in Section 4.

3. Related Work

Ranking of news is a rather recent discipline. Most commercial news sites have some mechanism to rank different news articles. While some providers rely on expert editors, other sites like Digg and Slashdot rely on social human filtering. Google News provides an aggregate news service which is automated, and close to our objective. Although they are not publicly available, some of the features they take into account can be inferred from analyzing the pages. [3] discloses the use of large scale collaborative filtering and text clustering as important constituents of the Google news algorithm. A major part of the Google news approach is the identification of a topic to group articles dealing with the same event together. Classification of the news articles into predefined categories precedes this clustering. Obvious features to rank these different topics within one category are the size of the cluster, the time the articles were published, and the sources who published the articles. These features are also relevant for ranking different news articles within one topic.

[1] describes a ranking algorithm for news sources and articles. They take into account that an important news topic generates many articles from different sources. They also included a mechanism to mutually reinforce scores of articles and their sources. Because they are looking at a stream of news, time awareness of the algorithm is a crucial point, since old news are considered less valuable than new news. A last feature of their ranking algorithm is the possibility of online processing of the data and of ranking the different articles on the fly.

A slightly different approach is presented in [11]. In their paper, the authors make two assumptions on what is an important news article: First, important news have a prominent spot in news homepages; second, important news are covered by various news sites. This allows an internal ranking for each article on news sites based on visual layout and mirrors the relation between an event and articles about this event. These relationships between homepages and news, as well as between events and news are used to model a news-importance relationship by a tripartite graph where homepages, news, and events are nodes. Each homepage gets a weight according to its credibility; each news article and each event get weights according to their importance. The weights are computed using an iterative algorithm exploiting the graph information. The evaluation shows that results improve significantly by taking into account both measures: visual layout and event clustering information. Notice that this approach is not suitable for news feeds (e.g.) based on RSS.

In [6] this approach is modified by changing the graph structure and only considering news and sources and the corresponding relations between them. The use of a semi-supervised learning algorithm is proposed to predict the recommendation strength of a news site for articles on other news sites, which leads to more edges in the graph and yields a better performance for the algorithm. Similarity between articles is measured using a vector space model and the relation between sources and articles are weighted using visual layout information.

All these systems have one common feature; they use information from news pages on the Internet, either taking the number of similar news articles into account or the internal ranking of articles within news pages. The drawback of these approaches is that they give an overview of what news are there and they rank these news items without regarding their intrinsic important-importance. However, newspapers and news sites have to publish articles even if nothing really important happened; thus all news which is on the front page of a news site, is not equally important. The result of the described systems is always relative for a given time period. Further, there is an implied dependence on social feedback, or duplication; however, this information is not necessary available when a news item is reported.

4. News Prediction using Text Analysis

Our proposed model for news ranking is exploratory in nature: i.e. we are interested in finding which features are most indicative of importance. Traditionally, tree classifiers have been used for such a supervised classification setup. However, Support Vector Machines (SVM) [9] are the de facto method used for classification nowadays. This is because SVMs can be much more accurate, and use non-linear regression as well by using *kernels*. A drawback of SVMs is the lack of a directly interpretable model; one can however interpret the importance of a feature, by looking at its weight in the support vector learnt. Thus, good feature selection is the key to our solution. As established before, we would like to use features of news which are textual in nature.

Part-of-Speech Tagging. Since the news space consists of free-form text, we can have scalability problems in dealing with large vocabularies. We therefore introduce part-of-speech (POS) tagging: a POS tagger labels words in a sentence as a noun, verb, pronoun, adverb, adjective etc.

A POS tagger divides the feature space into smaller subsets: we can then use a bag-of-words model to represent each word category separately. This classification of text helps us to understand which part-of-speech carries the maximum information w.r.t. importance and allows for reducing the vectorspace by discarding certain word categories.

We modified the bag of words algorithm to only include words tagged with certain part-of-speech information (nouns, verbs, adjectives or combinations of them). The necessary information is gained by using the part-of-speech tags generated by the Hepple tagger [5]. Terms tagged as noun or proper noun are included in the bag of words approach for nouns, terms tagged as any kind of verb are used for building the feature set for the verb approach, and for adjectives the same.

Named Entities. We use Named Entity Recognition to identify named entities in the news articles. The different named entity categories are then used as feature sets for the SVM. The first feature is the *Location* of the event which the news describes. A gazetteer list containing entries about locations like countries, cities, etc. is used to extract location information from the articles. The same approach is used to identify *Organizations* like “United Nations”, and *Job Titles* like “President”. Another feature set used consists of the *Persons* as found by the Named Entity Transducer. These individual features are normalized by dividing the number of occurrences by the number of total occurrences of a category in each article.

To evaluate the effectiveness of different feature sets used to build predictive models we designed a component-based system which allows us to select the desired features for each run.

Bigrams. To find out whether the coocurence of two consecutive terms in a text can help to assess the importance we investigated the usefulness of bigrams. Therefore we removed stopwords from the text and built bigrams out of the lemmas of the two terms.

5. Experimental Setup

For evaluation we collected data from Google News [2]. We downloaded stories displayed in the “World”-category between Nov 15th, 2007 and July 3rd, 2008. This resulted in a total of 1295 topics, each containing between 3 and 5 articles from the time when the topic first appeared. We performed some cleaning of the HTML pages to the navigation and advertisement information on the pages as well as HTML tags. Stopwords were removed and the text was lemmatized. Table 1 gives an overview of the properties of our corpus

Feature	No. of Occurrences
Nouns	39488
Verbs	3504
Adjectives	9794
Organizations	6560
Job Titles	589
Locations	3502
Persons	15543
Bigrams	726917

Table 1. Number of occurrences for individual features in our corpus

Since importance is a subjective concept, we require robust importance feedback for training our classifier. We first considered manual annotations by human classifying articles as important or unimportant. Since this data is bound to have individual biases, we chose to use a statistic provided by Google News for each news item. This statistic is the *cluster size* reported for each article group (*topic*). Google News uses text clustering to group similar news; updates and growth in the news story is usually clustered in the same group. We argue that the relative importance

[2] http://news.google.com

of a news article or topic is dependent on cluster size; this concept is related to both popularity used by Social news sites, as well as citation analysis used for web ranking (e.g. pagerank [7]). We also argue that cluster size is a robust indication of importance. Our task is thus to train classifiers and find which features are indicative of importance. Note that other sources can also be substituted for cluster size without any difference in the methodology.

We represent each topic using a space vector model. If a topic consists of more than one article we use all terms and normalize the weights for each term by dividing it by the number of articles. The weights are standard tf*idf scores:$w = \text{tf} * \log(\frac{N}{\text{df}} + 1)$, with N =number of topics. Experiments with giving higher scores to terms if they appear early in the text, e.g. in the headline of an article or the first 3 sentences, gave worse results. Using a cut off value to remove low ranked terms from the topic vector yielded worse results as well.

For our experiments we have two different setups: The first one is a binary classification problem finding out which topics are important and which are not. Therefore we had to define the threshold for the cluster size to classify the topics. Table 2 shows the accuracy for different thresholds t using only nouns as features. Although our performace is only slightly better than that of the baseline[3] for higher thresholds, the recall for finding important topics is always higher (e.g. with a threshold of 500, recall for identifying the important topics is 35% with SVM (vs. 0% for the baseline) and precision is 58%). A manual evaluation showed that 500 is actually a good threshold for the binary classification in terms of what a human would consider important, thus 21% of the articles in the collection can be considered important. The following results are therefore based on a threshold of 500.

Secondly, we classified the data into different bins based on cluster size: we used one bin for the topics with cluster size between 0 and 500 ; one for 500 to 1000; 1000 to 2000; and one bin for the news topics with more than 2000 articles in cluster. The bins were assigned the values '1', '2', '3', and '4' respectively. Figure 1 shows the classification of the topics into the different bins.

5.1. Metrics

We measure the accuracy of classification by computing the 0–1 loss function for a test set. This function reports an error 0 if the correct label has been assigned, and 1 otherwise.

For two class problems, 0–1 errors are very indicative of accuracy: for multi-class problems however, this function would treat the misclassification of *highly important* as *unimportant* the same as *very important*. Clearly, the degree of misclassification should be considered as well. Therefore, we use *Mean Average Error* (MAE) and *Root Mean Square Error* (RMSE) using the label set $\{1, 2, 3, 4\}$.

$$MAE = \frac{1}{n}\sum(Label_{Predicted} - Label_{Actual}) \qquad (4)$$

$$RMSE = \sqrt{\frac{1}{n}\sum(Label_{Predicted} - Label_{Actual})^2} \qquad (5)$$

All reported errors in this paper are measured using cross validation.

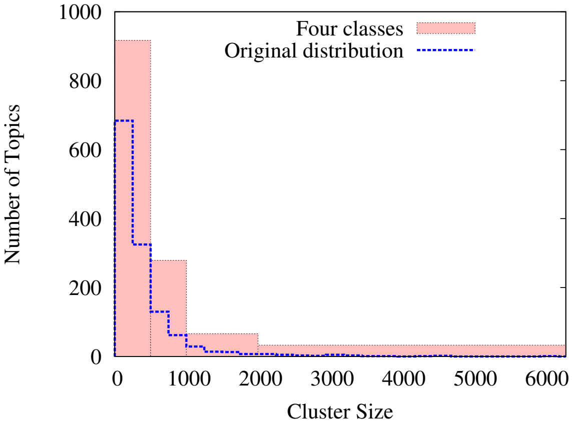

Figure 1. Distribution of the news article corpus into 4 different bins

5.2. Implementation

To preprocess the data and extract the different features we used GATE [2], a NLP-framework. For data mining and classification, we use the WEKA [10] toolkit. WEKA supports several standard data mining tasks, as well as data preparation steps. More specifically, algorithms for clustering, classification, regression, visualization, and feature selection are implemented. In addition, error evaluation protocols like cross validation are supported.

6. Discussion

Prediction Accuracy: Table 3 summarizes the main results of our experiments; we notice that textual features are indicative of importance, but prediction accuracy is not very high. This trend does not change significantly as the ratio of training and test data is varied. We notice that binary prediction (2 classes: important and not important) is an easier problem, with up to nearly 80% accuracy achieved with linear SVMs. An important

[3] Baseline: Classify all topics as unimportant for $t > 200$ and as important for $t = 200$

Threshold t of Cluster size	No. of Unimportant News Topics (Cluster Size $< t$)	No. of important News Topics (Cluster Size $>= t$	Cor. Classified Baseline	Cor. Classified SVM only Nouns
600	1062	233	82.01%	83.32%
500	1002	293	77.37%	79.46%
400	906	389	69.96%	71.97%
300	767	528	59.23%	65.71%
250	658	637	50.81%	66.26%
200	538	747	57.68%	65.41%

Table 2. Results for different thresholds (news cluster size) to seperate unimportant and important topics

observation we make is that our trained classifiers are highly accurate in predicting truly important news correctly. For detecting unimportant news, using only nouns yields the highest precision, whereas for predicting important topics, using all features results in more than 10% higher precision. However, several false positives are generated; this indicates that news reporting might be making some articles appear more important than they actual are.

Regression Accuracy: As explained earlier, a regression task is more sensitive to larger misclassification errors. Table 3 shows the experimental results for various strategies. Best regression results for the 4 class problem were achieved when using all features (highest correlation and lowest RMSE); however, the lowest MAE was observed when using only job titles. To verify this, we trained SVM models using a RBF kernel, as well as a C4.5 decision tree [8] classification tree[4]. Linear SVMs were observed to provide the highest 4-class classification accuracy.

Most Discriminative Features: We mined the SVM model to find the most discriminative features; the features with the highest weights are indicated in Table 4. We note that world leaders are identified as influential features (Musharraf, Bush), and disaster related events (e.g. wreckage, explode) also figure highly. News containing terms like ”grow”, “probe” (related to Economic news) is also considered more important that others. Adjectives usually carry smaller weight-age than nouns and verbs; this trend is noticeable in both 2-class and 4-class classification.

Category	Weight	DF	Labels
person	0.018	58	Mohammed
adjective	0.018	86	state-run
verb	0.016	425	head
adjective	0.016	535	dead
bigram	0.016	17	dead police
verb	0.015	295	gather
adjective	0.014	57	authorized
bigram	0.014	29	close ally
bigram	0.013	151	identity trap
noun	0.013	890	violence
noun	0.013	741	car
job title	0.013	1199	spokesman
person	0.013	53	President Vladimir Putin
location	0.012	18	eastern Baghdad
noun	0.012	188	reaction
person	0.011	115	Prime Minister Brown
job title	0.011	1069	official
job title	0.011	43	CEO
location	0.010	105	Bali
location	0.010	119	Scotland
verb	0.009	1008	remain
orgs	0.008	22	Labor Party
orgs	0.008	106	Senate
orgs	0.007	117	Pentagon

Table 4. Three most indicative terms for each feature for binary classification. DF is the document frequency with a total of 5182 articles (1295 topics) in the corpus

Changing importance of terms: We also investigate if the weights of features change with time; this seems intuitive as really important world events can make a location or a person suddenly famous. In Figures 2, 3, 4 and 5, we see the importance the classifier assigned to selected terms over a period of six months. We notice exploding importance of certain locations for different months (e.g. “Northern Ireland” in March, “Palestine” in June), v/s rather steady weights for general terms like “warfare” or “bomb”. The changing political environment also influences the importance of politicians’ names,“Musharraf” is steadily loosing importance, whereas “Abbas” experiences an importance peak in the month of May. Terms like “violence” and “arrest” are also likely to suddenly become more important. These interesting trends are prominent in our data collection; we expect that analysis over longer term data will lead to a robust set of features and higher accuracy.

[4] WEKA refers to its C4.5 implementation as J48

Used Features	Binary Classification			4 Class Regression Task		
	Correctly Classified	Precision for unimp. Topics	Precision for imp. Topics	Correlation Coefficient	Mean Abs. Error	Root Mean Squared Error
only nouns	**79.46%**	**82.9%**	57.6%	0.3535	0.5104	0.6771
all features	79.31%	79.8%	**69.8%**	**0.4110**	0.5111	**0.6455**
bi-grams	78.92%	79.5%	67.2%	0.3985	0.5153	0.6507
all named entities	78.53%	81.6%	55.0%	0.3453	0.5085	0.6751
verbs,nouns,adj	77.99%	81.7%	52.4%	0.3744	0.5034	0.6626
only persons	77.53%	81.2%	50.6%	0.3441	0.5032	0.6793
only job titles	76.37%	78.4%	40.8%	0.1421	**0.5020**	0.7863
only organizations	74.52%	79.9%	39.2%	0.1609	0.5707	0.7977
only locations	74.44%	81.2%	41.3%	0.1145	0.6312	0.9651
only verbs	71.89%	81.7%	37.7%	0.2087	0.6562	0.8782
only adjectives	71.00%	79.8%	34.2%	0.2434	0.5618	0.7452

Table 3. Results for binary classification and for regression task using 4 classes: "extremely important", "highly important", "moderately important", and "unimportant"

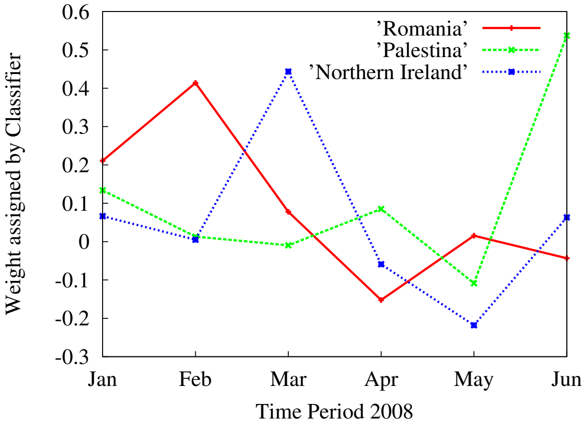

Figure 2. Assigned weights over different periods for selected locations

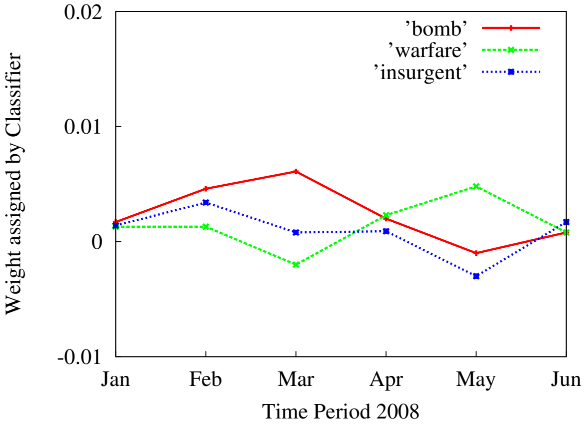

Figure 3. Assigned weights over different periods for selected nouns

7 Conclusions and Future Work

We have explored the use of textual features for creating an understandable prediction model for news importance. Experimental results indicate that this is a hard problem, with pure textual information being insufficient for creating accurate classifiers. We report interesting observations of nouns being more indicative of importance, and the sensitivity of classifiers to world events. Clearly, adding more training data increases prediction accuracy, we thus try to build up a larger corpus. We also try to improve the fulltext extraction of the articles from the websites to get less noisy data. There is significant scope in improving accuracy, and we continue to explore additional features which will make this possible.

8 Acknowledgments

The authors would like to thank Keith Hall for his valuable comments on the paper.

References

[1] G. M. D. Corso, A. Gullí, and F. Romani. Ranking a stream of news. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 97–106, New York, NY, USA, 2005. ACM.

[2] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proc. of the 40th Anniversary Meeting of the ACL*, 2002.

[3] A. Das, M. Datar, A. Garg, and S. Rajaram. Google news personalization: scalable online collaborative filtering.

Rank	Top Stories	Flop Stories
1	5.4 Million Dead in DRC Since 1998, Says New Survey (01.22.08)	New faces for Irish politics (04.14.08)
2	Bush and Sarkozy seek united front against Iran (07.14.08)	Australia to remove almost 100 anti-gay laws (04.30.08)
3	Myanmar death toll soars (05.17.08)	Bangladesh government announces national elections (05.12.08)
4	Serb protesters attack U.S. Embassy (02.21.08)	Motive still unknown as serial killer faces rest of life in prison (02.22.08)
5	Bomb Blast in Yemen Kills 18 at Mosque (05.02.08)	BAE chief subpoenaed in U.S. over Saudi arms deal (05.18.08)
6	Democrats take White House campaign to Puerto Rico (05.24.08)	As Bush leaves Mideast, he gives Arab leaders a to-do list for reform (05.19.08)
7	Raj Thackeray arrested in Mumbai, gets bail (02.12.08)	Teachers struggle with immigrant pupil influx (03.21.08)
8	MPs back animal-human embryos for research (05.19.08)	Rudd targets UN council seat (03.30.08)
9	Mugabe Is Sworn In to Sixth Term After Victory in One-Candidate Runoff (06.30.08)	Iraqi militia to hear Saturday whether to resume fighting (02.20.08)
10	Iran Threatens To ”Explode” Ships (01.08.08)	No More Deja Vu: A Tenacious Negotiator Cuts A Deal On Hebron (02.27.08)

Table 5. Top 10 and flop 10 stories as predicted by our system

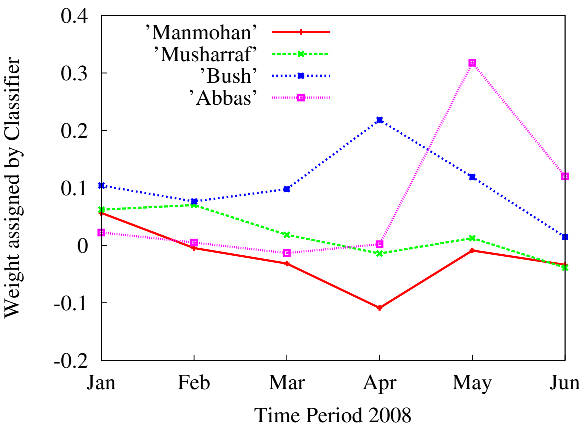

Figure 4. Assigned weights over different periods for selected persons

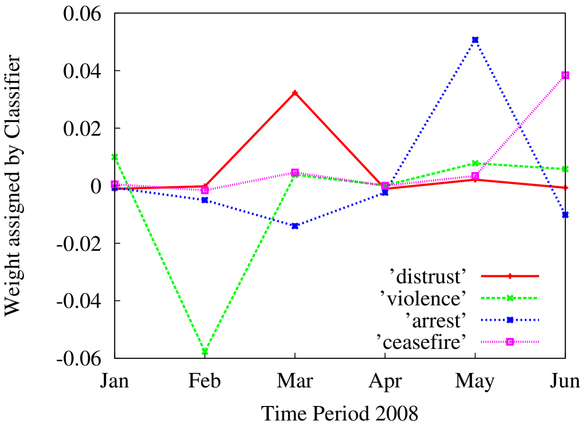

Figure 5. Assigned weights over different periods for selected nouns

Proceedings of the 16th international conference on World Wide Web, pages 271–280, 2007.

[4] R. Goonatilake and S. Herath. The volatility of the stock market and news. *International Research Journal of Finance and Economics*, 3(11):53–65, September 2007.

[5] M. Hepple. Independence and commitment: assumptions for rapid training and execution of rule-based POS taggers. *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 278–277, 2000.

[6] Y. Hu, M. Li, Z. Li, and W.-Y. Ma. Discovering authoritative news sources and top news stories. In H. T. Ng, M.-K. Leong, M.-Y. Kan, and D. Ji, editors, *AIRS*, volume 4182 of *Lecture Notes in Computer Science*, pages 230–243. Springer, 2006.

[7] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[8] J. R. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann, San Francisco, CA, USA, 1993.

[9] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 2000.

[10] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, second edition, June 2005.

[11] J. Yao, J. Wang, Z. Li, M. Li, and W.-Y. Ma. Ranking web news via homepage visual layout and cross-site voting. In M. Lalmas, A. MacFarlane, S. M. Rüger, A. Tombros, T. Tsikrika, and A. Yavlinsky, editors, *ECIR*, volume 3936 of *Lecture Notes in Computer Science*, pages 131–142. Springer, 2006.