RssE-Miner: A New Approach for Efficient Events Mining from Social Media RSS Feeds

Nabila Dhahri[1], Chiraz Trabelsi[1], and Sadok Ben Yahia[1,2]

[1] Faculty of Sciences of Tunis, University Tunis El-Manar, 2092 Tunis, Tunisia
[2] Department of Computer Science, TELECOM SudParis, UMR 5157 CNRS Samovar, 91011 Evry Cedex, France
dhahri.nabila@gmail.com,
{chiraz.trabelsi,sadok.benyahia}@fst.rnu.tn

Abstract. Most of the new social media sites such as Twitter and Flickr are using RSS Feeds for sharing a wide variety of current and future real-world events. Indeed, RSS Feeds is considered as a powerful real-time means for real-world events sharing within the social Web. Thus, by identifying these events and their associated user-contributed social media resources, we can greatly improve event browsing and searching. However, a thriving challenge of events mining processes is owed to an efficient as well as a timely identification of events. In this paper, we are mainly dealing with event mining from heterogenous social media RSS Feeds. Therefore, we introduce a new approach, called RssE-Miner, in order to get out these events. The main thrust of the introduced approach stands in presenting a better trade-off between event mining accuracy and swiftness. Specifically, we adopted the probabilistic Naive Bayesian model within the exploitation of the rich context associated with social media Rss Feeds contents, including user-provided annotations (e.g., title, tags) and the automatically generated information (e.g., time) for efficiently mining future events. Carried out experiments over two real-world datasets emphasize the relevance of our proposal.

Keywords: Event Identification, Social Media, Real Time, Really Simple Syndication, News Mining, Unstructured Data.

1 Context and Motivations

The task of event identification embraces two well-known approaches: linguistic approach and statistical one. The former constitute a hindrance since linguistic expert uses rules for manually defining event patterns [11]. In this paper, we do some explorations on the latter direction.

In this context, the topic detection and tracking (TDT) event detection task [16,14], which is similar to our event identification task, was studied on a continuous stream of news documents with the aim to identify news events and organize them. Most techniques mainly focused on formal text stream data. However, social media RSS Feeds, which are the focus of this paper, are substantially different from formal text stream data as the text is so noisy that

A. Cuzzocrea and U. Dayal (Eds.): DaWaK 2012, LNCS 7448, pp. 253–264, 2012.

its handling becomes more difficult [12]. In this respect, searching or browsing through the web becomes difficult and inaccurate.

Recently, RSS Feeds have become ubiquitous with the evolution of the web. Especially, social RSS Feeds are defined as a collection of informal data that arrives over time and each RSS Feed item is associated with some social attributes such as title, description, tags. These features contain information about real world events that are manually added by users. The key challenges that we address are: the identification of events and their associated news items over social media sites(e.g.,Flickr, Youtube, and Facebook).

It can happen that you think to attend the carnival in Rio de Janeiro. Hence, you make your way to the computer to seek information about Rio Carnival with the goal to purchase a ticket as soon as possible. Unfortunately, there are too many photographs dealing with this event. You sift through many different photographs interleaved with photographs of the same event happened at the last year. At the end, you are lost among these results and there is a possibility that you miss the deadline for buying tickets. The above scenario is a likely and perhaps frequent occurrence on the Web today.

Many research studies have then attempted to identify events from social media sites [4,8]. Most of them opt for the popular Support Vector Machine as a machine learning technique. This classifier is memory-intensive and time consuming. That's why, we think about Naive Bayes as an accurate and fast classifier that makes real-time use possible and does well if you have fairly little data [9] as the case in social media. Thus, social media documents contain little textual narrative, usually in the form of short description, title, or tags.

We apply an appropriate algorithm, namely Naive Bayes, for the event identification task in social media RSS Feeds trading off runtime performance and classification accuracy. We apply our approach to two real-world datasets derived from Flickr. We refer to these datasets as Upcoming and Last.fm, as the labels have been extracted from these sites.

This paper is structured as follows. First, Section 2 discusses the related work. Section 3 elaborates in the proposed approach. Then, in Section 4 the approach is evaluated. Finally, Section 5 concludes the paper and provides future research directions.

2 Related Work

We describe related work in four areas namely event identification or detection, RSS Feeds studies, social media analysis, and Naive Bayes applications.

The event detection task [16,14] was studied on a continuous stream of news documents with the aim to identify news events and organize them. This is one of the important tasks considered by the topic detection and tracking (TDT) [2]. However, our work is distinguished by the fact that we are interested in event identification in social media RSS Feeds where the text is so noisy that its handling becomes more difficult [12]. In our current endeavors, we aim to have a fully automated application for processing social media events, fetched from

Really Simple Syndication(RSS) Feeds in such a way that the essence of the social media resources is extracted and captured in events that are represented in a machine-understandable way .

There are several studies that have focused on the RSS Feeds processing: We can talk about SPEED (Semantics-Based Pipeline for Economic Event Detection) [11] which is a framework that aims at extracting financial events from news articles(announced through RSS Feeds) and annotated these with metadata at a speed that makes real-time use possible. It's modeled as a pipeline that reused some of the ANNIE GATE components and develop new ones. We can found also SemNews [13] which is a Semantic Web-based application that aims at accurately extracting information from heterogenous sources. It seeks to discover the meaning of news items. These items are retrieved from RSS Feeds and are processed by the NLP engine OntoSem. Our work also involves the processing of RSS Feeds but using the statistical approach and not the linguistic approach. The latter constitute a hindrance since linguistic expert uses rules for manually defining event patterns, which is prone to errors.

Several efforts have focused on social media analysis [7,3]. There are those which have focused on Flickr tags as significant descriptors [17,1]. Others are interested in the wealth of available context [4,8] including date and location and concluded that despite the noise that clutters the social media, take advantage of all its context can provide relevant information about the event. Most techniques mainly focused on context and ignore time needed for the context analysis. In this respect, we explore the rich context of social media with respect to the time constraint providing top-tier accuracy with a fraction the training time of alternative methods.

There are various applications of the Naive Bayes algorithm. It has been applied for automatic categorization of email into folders [5]. Email arrives in a stream over time. It was mentioned that Naive Bayes is the fastest algorithm compared to, respectively, MaxEnt, SVM and Winnow. Naive Bayes was also used as a pre-trained model for real-time network traffic classification [6]. Naive Bayes represents,yet, one of the most popular machine learning models applied in the spam filtering domain [18]. Importantly, the learning process of Naive Bayes is extremely fast compared with current discriminative learners, which makes it practical for large real-world applications. Since the training time complexity of Naive Bayes is linear to the number of training data, and the space complexity is also linear in the number of features, it makes Naive Bayes both time and storage efficient for practical systems. This led us to opt for the choice of the Naive Bayes algorithm for social media RSS Feeds processing.

3 RssE-Miner: Efficient Events Mining from Social Media RSS Feeds

We elaborate on event identification in social media RSS Feeds. To fulfill this task, we propose our approach for efficient events mining from social media RSS Feeds (c.f., figure 1). Indeed, The Naive Bayes classifier is our choice in the

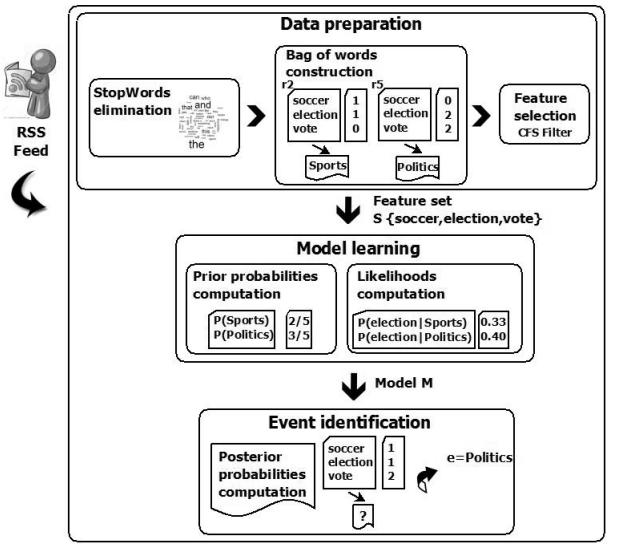

Fig. 1. The RssE-Miner approach

classification-based technique to predict when RSS Feed items correspond to the same event. The classifier presented in this section is based on a probabilistic model simply trying to give each resource r the label of the event e which is most likely. Hence, our proposed approach has three steps: **Data preparation**, **Model learning** and **Event identification**. In the following, we describe these three steps in more details.

3.1 Data Preparation

Common Refinements such as Removing Stop Words and Symbols are applied to the data. Each RSS Feed item, which is our resource, is represented as a bag of words *w1*,*w2*,*w3* etc. There are many design choices in the feature construction. In this paper, we use the traditional bag-of-words document representation. We do not apply stemming. Finally, we apply feature selection by identifying the most salient features for learning. In fact, CFS(Correlation based Feature Selection) filter is used[10]. It is a simple filter algorithm that ranks feature subsets according to a correlation based heuristic evaluation function:

$$M_S = \frac{k\overline{h_{cf}}}{\sqrt{k + k(k-1)\overline{h_{ff}}}}. \tag{1}$$

where M_S is the heuristic "merit" of a feature subset S containing k features, $\overline{h_{cf}}$ is the mean feature-class correlation ($f \in S$), and $\overline{h_{ff}}$ is the average feature-feature inter-correlation. In this paper, Best first heuristic search is used.

3.2 Model Learning

The Naive Bayes classifier is provided by a simple theorem of probability known as Bayes' rule [15]:

$$P(e|r) = P(e).\frac{P(r|e)}{P(r)}. \tag{2}$$

Hence, the Naive Bayes classifier need to compute P(e) and P($r|e$). Since our resource is represented as a bag of words, required P(e_j) and P($w_k|e_j$) terms are calculated for each e_j.

The former is given by:

$$P(e_j) = \frac{|r_j|}{|r|}. \tag{3}$$

where $|r_j|$ is a subset of resources for which the target event is e_j.

The latter,for each word w_k in the vocabulary, is given by:

$$P(w_k|e_j) = \frac{n_k + l}{n + l|Vocabulary|}. \tag{4}$$

where n_k is the total number of occurrences of w_k where target event is e_j, n is the total number of words in all training examples whose target value is e_j and l is the Laplacian smoothing.

The likelihoods P($w_1, w_2, ..., w_n|e_j$) are computed using the (naive) independence assumption. A common strategy is to assume that the distribution of $w_1, w_2, ..., w_n$ conditional on e_j can be decomposed in this fashion for all e_j:

$$P(w_1, w_2, ..., w_n|e_j) = \prod_i P(w_i|e_j). \tag{5}$$

The following section will explain the event identification stuff in detail.

3.3 Event Identification

Once the model learning step is performed, we proceed with testing the model for identifying the appropriate event. Classification will occurs when event probability is calculated. To classify, we must find the class label e which is most likely to generate r. Then, we choose e which gives r the best score according to P($e|r$):

$$g(r) = \arg\max_e P(e)P(r|e). \tag{6}$$

And according to the equation 5 and the equation 6, we have:

$$e = \arg\max_{e_j \in E} P(e_j) \prod_i P(w_i|e_j). \tag{7}$$

Typically, the denominator in equation 2 is not explicitly computed since it is the same for all e_j.

3.4 RssE-Miner: Application

Assuming that our data is passed through the Data preparation step, Table 1 represents the output of this step. In this example, we set the Laplacian smoothing l equal to 1.

Table 1. Model learning example

w	soccer	election	vote	label
r1	1	0	0	Sports
r2	1	1	0	Sports
r3	0	0	1	Politics
r4	0	1	1	Politics
r5	0	2	2	Politics

Table 2. P($word|label$) calculation

word	label	P($word|label$)
election	Sports	0.33
election	Politics	0.40
soccer	Sports	0.50
soccer	Politics	0.10
vote	Sports	0.17
vote	Politics	0.50

The model computes the prior probabilities for each class label. Then, Probabilities for every word are calculated.

$$P(Sports) = 2/5. \tag{8}$$

$$P(Politics) = 3/5. \tag{9}$$

The word "election" occurs 1 times in "Sports" resources.

The total number of words in "Sports" resources = 1+1+1= 3. Then

$$P(\text{"}election\text{"}|Sports) = (1+1)/(3+3) = 1/3. \tag{10}$$

The word "election" occurs 3 times in "Politics" resources.

Then

$$P(\text{"}election\text{"}|Politics) = (3+1)/(7+3) = 2/5. \tag{11}$$

Table 2 resumes this stuff.

We try to treat the same example mentioned above to test our model. Hence, Table 3 provides a resource that we seek the corresponding event.

Table 3. Event identification

w	soccer	election	vote	label
r6	1	1	2	?

– e_j=Sports :

$$\begin{aligned} P(r6|Sports) &= P(\text{"}soccer\text{"}|Sports)P(\text{"}vote\text{"}|Sports)^2P(\text{"}election\text{"}|Sports) \\ &= 0.0048 \end{aligned} \tag{12}$$

– e_j=Politics :

$$\begin{aligned} P(r6|Politics) &= P(\text{"}soccer\text{"}|Politics)P(\text{"}vote\text{"}|Politics)^2P(\text{"}election\text{"}|Politics) \\ &= 0.010 \end{aligned} \tag{13}$$

Then the event with the highest posterior probability, is selected.

$$e_j = Politics. \tag{14}$$

For this example, the event is correctly identified.

4 Experimental Evaluation

In this section we describe how our approach is evaluated on two real world datasets derived from Flickr. We use traditional classification accuracy[1] as well as precision[2] and recall[3] as our evaluation measures. We elaborate on our choice of Naive Bayes classifier.

4.1 Experimental Settings

Dataset Collect. We collected our dataset from the online photo management and sharing application Flickr using the Flickr API[4]. It consists of RSS Feeds of two real world datasets Upcoming and Last.fm. In this section we describe these two datasets and provide some basic statistics on their content. The dataset used is a set of news items fetched from Flickr RSS Feeds from January to June 2006. The Upcoming dataset contains 5778 images spread over 362 unique events. The Last.fm dataset contains 3356 images spread over 316 unique events. In order to emulate real-world scenario, we order the items in the dataset by their time of upload.

Baseline Models. To the best of our knowledge, event identification (using Naive Bayes) in such social media sites have never been modeled before. Thus, for enhancing the efficiency as well as the effectiveness of our approach, we compare the results of our approach to three baselines:

- **Most popular events identified:** For each event, we counted in how many resources it occurs and used the resources ranked by event occurrence count. For each event, the resources are randomly selected.
- **Most popular event aware extracted:** Events are weighted by their co-occurrence with a given event. Then, resources are ordered without validation.
- **SMO:** we compare our approach vs the Becker et al.(2010) [4] approach (we only make use of its classification-based technique part). In fact, SVM was used to learn document similarity functions for social media. In other

[1] Ratio of correctly classified instances to the total number of instances in the test set.
[2] percentage of resources correctly predicted in the class C from those predicted in this class.
[3] Percentage of resources correctly predicted in the class C from those actually in this class.
[4] `http://www.flickr.com/services/api/`

words, Becker et al.(2010) used SVM as a classifier with similarity scores as features to predict whether a pair of documents belongs to the same event. They selected Weka's sequential minimal optimization implementation. In this respect, We implement Naive Bayes as a part of the Weka[5] software system.

4.2 Efficiency of Our Approach

We report in the following results averaged over 10 test runs. We empirically decide to use only description, title and tags features. Indeed, the presence of other features such as location is an indication of document dissimilarity.

Dataset Phenomena. We evaluate the performance of the proposed approach on a sparse and a dense datasets. The Upcoming dataset is a sparse data since it contains different kind of events which are of public interest. That's why, it is less likely to find two or more items that belong to the same event. thus, it includes fewer items per event. However, the Last.fm dataset is a dense data since it includes only events in the area of music. That's why, it is more likely to find many items per event. As mentioned in Figure 1, the behavior of the two classifiers is almost the same in the two datasets.

Figure2 (Left), depicts averages of accuracy on the Upcoming sparse dataset. Figure 2 (Right), depicts averages of accuracy on the Last.fm dense dataset. On both datasets, SVM demonstrates the highest accuracies. In fact, SVM outperforms Naive Bayes - by notable 1% in the case of Upcoming dataset and by notable 1% in case of Last.fm dataset, but the difference is not statistically significant. However, the performance of Naive Bayes could likely be improved by applying a more sophisticated smoothing method than Laplace. Naive Bayes accuracy, for now, is acceptable since we seek a compromise between runtime performance and classification accuracy. Next, we elaborate in the runtime performance.

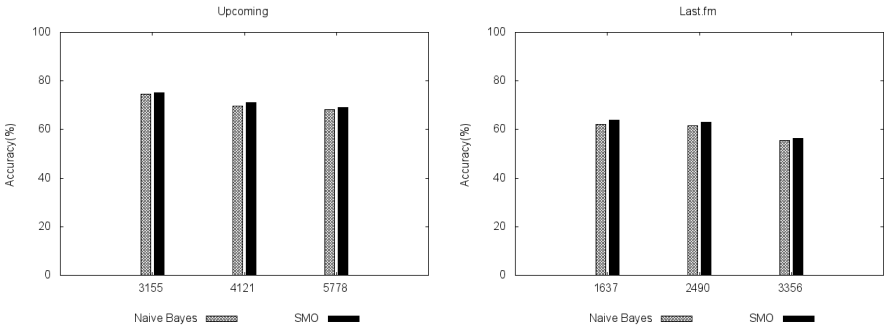

Fig. 2. Left: Average of Accuracy on the Upcoming dataset; **Right**: Average of Accuracy on the Last.fm dataset

[5] `http://www.cs.waikato.ac.nz/ml/weka/`

Running Time. As mentioned above, our main goal is to achieve a meaningful trade-off between runtime performance and classification accuracy. Table 4 shows that our approach has succeeded in fulfilling this task. Thus, according to the results given, we can point out that our approach outperforms baseline one. In fact, as expected, the Runtime of the SVM classifier are much slower than those achieved by our approach for both datasets. We note that Naive Bayes is by far the fastest classifier. It takes no more than 2.278 seconds on Upcoming dataset and no more than 0.739 seconds on Last.fm dataset. In particular, our approach outperforms the baseline one by a large and statistically significant margin. To this end, Naive Bayes makes real-time use possible. This is of paramount importance in the case of event identification in social media RSS Feeds as faster processing of data enables one to make better informed decisions.

In all, evaluation underlines fast and accurate performance by applying our approach. Results show that event identification using Naive Bayes model can work in near real-time without obvious decrease in accuracy.

Table 4. Average of Runtime on the Upcoming and Last.fm datasets

	Instances	Features	Events	Runtime(s)	
				Naive Bayes	SMO
Upcoming	3155	30	203	**0.596**	34.213
	4121	33	280	**1.215**	66.392
	5778	36	362	**2.278**	115.33
Last.fm	1637	21	171	**0.18**	24.571
	2490	27	243	**0.519**	50.143
	3356	23	316	**0.739**	85.829

4.3 Effectiveness of Our Approach

We present in figure 3 precision as well as recall measures in Upcoming and Last.fm datasets. Indeed, according to the sketched histograms, we can point out that our approach outperforms both baselines. On Upcoming dataset, the average recall achieves high percentage for higher value of N. Indeed, for N = 58, the average Recall is equal to 0.742, showing a drop of 98,38 % compared to the average Recall for N = 36. On Last.fm dataset, the average recall achieves high percentage for higher value of N. Indeed, for N = 46, the average Recall is equal to 0.878, showing a drop of 99,4 % compared to the average Recall for N = 41. In this case, for a higher value of N, by matching resources with their corresponding events, the proposed approach can achieve event identification task successfully. In addition, the percentage of precision for the proposed model outperforms the two baselines. On Upcoming dataset, our approach achieves the best results when the value of N is around 58. In fact, for N = 58, it has an average of 68,3% showing an exceeding about 4% against the first baseline and around 59,6% against the second one. On Last.fm dataset, our approach achieves the best results when the value of N is around 41. In fact, for N = 41, it has an average of 87,3% showing an exceeding about 12.9% against the first baseline and around 64,5% against the second one. These results highlight that the proposed approach can better improve event identification task even for a high number of extracted events.

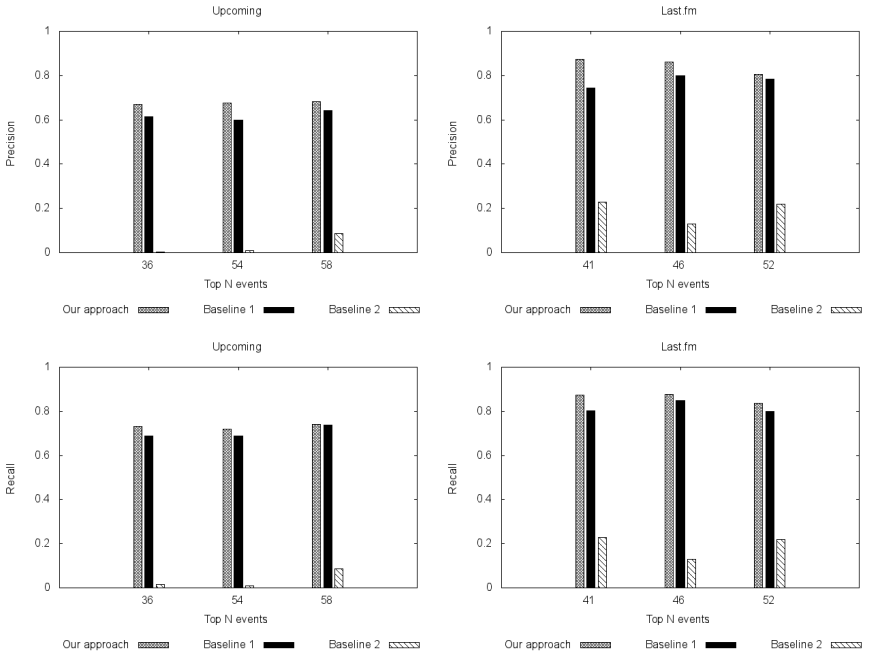

Fig. 3. Left: Precision and Recall on the Upcoming and Last.fm datasets

4.4 Online Evaluation

We present in figure 4 the runtime of RssE-Miner. Since it is hard to measure the exact runtime of the proposed approach, we simulated an online execution of our system among the Upcoming as well as the Last.fm datasets with different

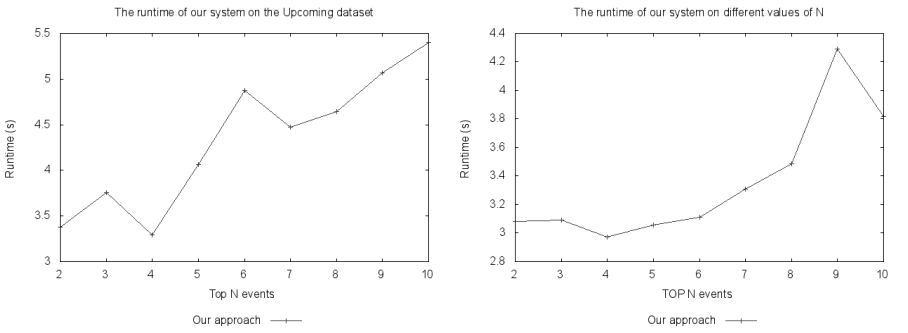

Fig. 4. Left: The runtime of our system online with different values of N on the Upcoming dataset; **Right**: The runtime of our system online with different values of N on the Last.fm dataset

values of N, i.e., the number of extracted events, ranging from 2 to 10. Hence, for each Flickr RSS Feed, we report the average runtime of the related top N events extracted. With respect to Figure 4, the maximum value of runtime is about 5.403(s) in the Upcoming dataset and about 4.292(s) in the Last.fm dataset, whereas the minimum value is around 3.292(s) in the Upcoming dataset and about 2.975(s) in the Last.fm dataset which is efficient and satisfiable.

5 Conclusion and Future Work

In this paper, we have tackled the task of event identification in social media RSS Feeds. We have formulated this task as a real-time problem and introduced a novel probabilistic approach for events mining from heterogenous social media RSS Feeds, called RssE-Miner, in order to get out these events. In particular, our approach relies on a better trade-off between event mining accuracy and swiftness by applying the probabilistic Naive Bayesian model to Flickr data. Our experiments suggest that our approach yields better performance than the baselines on which we build. To the best of our knowledge, event identification (using Naive Bayesian model) in such social media sites have never been modeled before. In future work, we will focus on further study other more sophisticated smoothing method than Laplace to improve Naive Bayes performance. Our future research will focus also on event ontology enrichment. Indeed, from these events, we aim to enrich an event ontology. Such an ontology is useful in providing accurate, up-to-date information in response to user queries.

Acknowledgments. This work was partially supported by the project Utique CMCU 11G1417.

References

1. Ahern, S., Nair, R., Kennedy, L., Naaman, M., Rattenbury, T.: How flickr helps us make sense of the world: context and content in community-contributed media collections. In: Proceedings of the 15th International Conference on Multimedia, MULTIMEDIA 2007, pp. 631–640. ACM (2007)
2. Allan, J.: Introduction to topic detection and tracking. In: Topic Detection and Tracking: Event-Based Information Organization, pp. 1–16. Kluwer Academic Publishers (2002)
3. Amer-Yahia, S., Lakshmanan, L.V.S., Benedikt, M., Stoyanovich, J.: Efficient network aware search in collaborative tagging sites. In: Proc. VLDB Endow., vol. 1(1), pp. 710–721 (2008)
4. Becker, H., Naaman, M., Gravano, L.: Learning similarity metrics for event identification in social media. In: Proceedings of the Third ACM International Conference on Web Search and Data Mining, WSDM 2010, pp. 291–300. ACM (2010)
5. Bekkerman, R., Mccallum, A., Huang, G.: Automatic categorization of email into folders:benchmark experiments on enron and sri corpora. Technical Report, Computer Science department, IR-418. pp. 4–6

6. Dann Wei Li, R., Abdin, K., Moore, A.: Approaching real-time network traffic classification. Technical Report, RR-06-12, Department of Computer Science, Queen Mary, University of London (October 2006)
7. Donato, D., Gionis, A., Agichtein, E., Castillo, C., Mishne, G.: Finding high-quality content in social media. In: Proceedings of the First ACM International Conference on Web Search and Data Mining, WSDM 2008, pp. 183–194. ACM (2008)
8. Drumond, L., Buza, K., Reuter, T., Cimiano, P., Schmidt-Thieme, L.: Scalable event-based clustering of social media via record linkage techniques. In: Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media, ICWSM 2011, pp. 313–320. AAAI Press (2011)
9. Forman, G., Cohen, I.: Learning from Little: Comparison of Classifiers Given Little Training. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.) PKDD 2004. LNCS (LNAI), vol. 3202, pp. 161–172. Springer, Heidelberg (2004)
10. Hall, M.A.: Correlation-based Feature Selection for Machine Learning. Doctoral thesis, The University of Waikato (April 1999)
11. Hogenboom, A., Hogenboom, F., Frasincar, F., Kaymak, U., van der Meer, O., Schouten, K.: Detecting Economic Events Using a Semantics-Based Pipeline. In: Hameurlain, A., Liddle, S.W., Schewe, K.-D., Zhou, X. (eds.) DEXA 2011, Part I. LNCS, vol. 6860, pp. 440–447. Springer, Heidelberg (2011)
12. Hurst, M., Sayyadi, H., Maykov, A.: Event detection and tracking in social streams. In: ICWSM. The AAAI Press (2009)
13. Java, A., Finin, T., Nirenburg, S.: Semnews: A semantic news framework. In: Proceedings of the 21st National Conference on Artificial Intelligence, AAAI 2006, pp. 1939–1940. AAAI Press (2006)
14. Kumaran, G., Allan, J.: Text classification and named entities for new event detection. In: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2004, pp. 25–29 (2004)
15. Lewis, D.D.: Naive (Bayes) at Forty. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 4–15. Springer, Heidelberg (1998)
16. Papka, R., Allan, J., Lavrenko, V.: On-line new event detection and tracking. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 1998, pp. 37–45. ACM (1998)
17. Ramage, D., Heymann, P., Garcia-Molina, H.: Social tag prediction. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, pp. 531–538. ACM (2008)
18. Song, Y., Kolcz, A., Lee Giles, C.: Better naive bayes classification for high-precision spam detection. Softw. Pract. Exper. 39(11), 1003–1024 (2009)