Event Identification in Social Media

Hila Becker
Columbia University
hila@cs.columbia.edu

Mor Naaman
Rutgers University
mor@scils.rutgers.edu

Luis Gravano
Columbia University
gravano@cs.columbia.edu

ABSTRACT

Social media sites such as Flickr, YouTube, and Facebook host substantial amounts of user-contributed materials (e.g., photographs, videos, and textual content) for a wide variety of real-world events. These range from widely known events, such as the presidential inauguration, to smaller, community-specific events, such as annual conventions and local gatherings. By identifying these events and their associated user-contributed social media documents, which is the focus of this paper, we can greatly improve local event browsing and search in state-of-the-art search engines. To address our problem of focus, we exploit the rich "context" associated with social media content, including user-provided annotations (e.g., title, tags) and automatically generated information (e.g., content creation time). We form a variety of representations of social media documents using different context dimensions, and combine these dimensions in a principled way into a single clustering solution—where each document cluster ideally corresponds to one event—using a weighted ensemble approach. We evaluate our approach on a large-scale, real-world dataset of event images, and report promising performance with respect to several baseline approaches. Our preliminary experiments suggest that our ensemble approach identifies events, and their associated images, more effectively than the state-of-the-art strategies on which we build.

1. INTRODUCTION

Social media sites (e.g., Flickr, YouTube, and Facebook) are a popular distribution outlet for users looking to share their personal news and interests. As a result, these sites host substantial amounts of user-contributed materials (e.g., photographs, videos, and textual content) for a wide variety of real-world events, ranging from popular, widely known events (e.g., a concert by a popular music band) to smaller events that might receive no coverage in traditional news outlets (e.g., a local social gathering, an annual convention, or a community-specific function). By identifying all these events—and their associated social media documents—we can enable powerful local event browsing and search, to complement and improve the local search tools that Web search engines provide. In this paper, we address the problem of how to identify events and their associated user-contributed documents over social media sites.

Consider a person who is thinking of attending "All Points West," an annual music festival that takes place in early August in Liberty State Park, New Jersey, featuring live performances by popular rock bands. Prior to purchasing a ticket, this person could search the Web for relevant information that would aid in making an informed decision. Unfortunately, Web search results are far from revealing for this relatively minor event: the event's website contains marketing materials, strategically selected by the event's producers, and traditional news coverage is low, with some articles providing the list of performers, and others discussing related topics such as "Festival producers optimistic despite recession." Overall, these Web search results do not convey what this person should expect to experience at this event. In contrast, user-contributed content may reflect prior instances of the event from an attendee's perspective. Such user-centric perspective, as well as coverage of not-so-prominent events such as the "All Points West" music festival, make social media sites a valuable source of event information.

Identifying events and their associated documents over social media sites is a challenging problem, as social media data is inherently noisy and heterogeneous. In our "All Points West" example, some photographs might contain the event's name in the title, description, or tag fields, while many others might not be as clearly linked, with titles such as "Radiohead" or "Metric" and descriptions such as "my favorite band." Photographs geo-tagged with the coordinates of Liberty State Park, New Jersey, and taken on August 8, 2008, are likely to be related to this event, regardless of their textual description, but not every photograph taken on August 8, 2008, or titled "Radiohead," necessarily corresponds to this event. Overall, social media documents generally exhibit information that is useful for identifying the associated events, if any, but this information is far from uniform in quality and might often be misleading or ambiguous.

Our problem is most similar to the event detection task [3, 25, 15], whose objective is to identify news events in a continuous stream of news documents (e.g., newswire, radio broadcast). However, our problem exhibits some fundamental differences from traditional event detection that originate in the social media sources on which we focus. Specifically, event detection traditionally aims to discover and cluster events found in textual news articles. These news articles adhere to certain grammatical, syntactical, and stylistic standards that are appropriate for their venue of publication. Therefore, most state-of-the-art event detection approaches leverage natural language processing tools such as named-

Twelfth International Workshop on the Web and Databases (WebDB 2009), June 28, 2009, Providence, Rhode Island, USA.

entity extraction and part-of-speech tagging to enhance the document representation [17, 26, 9]. In contrast, social media documents contain little textual narrative, usually in the form of a short description, title, or keyword tags. Importantly, as discussed above, this text is often noisy, which renders traditional event detection techniques undesirable over social media documents, as we will see.

While social media documents present challenges for event detection, they also exhibit opportunities not found in traditional news articles. Specifically, social media documents usually have a wealth of associated "context," including user-provided annotations (e.g., title, description, tags), as well as automatically generated information (e.g., upload or content creation time). Individual features might be noisy or unreliable, but collectively they provide revealing information about each social media document, and this information is valuable to address our problem of focus. In this paper, we exploit this rich family of features to identify events and their associated user-contributed social media documents. We explore distinctive representations of social media documents to analyze document similarity and identify which documents correspond to the same events. To compute the similarity between textual features of documents, we use state-of-the-art techniques from text document clustering. For numeric features (e.g., time and location), which are critical characteristics of events and, correspondingly, their associated social media documents, we consider similarity metrics that are tailored to the domain of each feature. We then determine multiple complementary clustering "votes" for the social media documents and combine these votes in a principled manner using a weighted "ensemble" approach for clustering the documents. Each final cluster corresponds to an event and includes the social media documents associated with the event.

The contributions of this paper are as follows:

- Posing the problem of identifying events and their user-contributed social media documents as a clustering task, where documents have multiple features, associated with domain-specific similarity metrics (Section 3).
- Using a "weighted ensemble" approach for clustering the social media documents that collectively considers the rich features of the documents (Section 4).
- Evaluating the ensemble clustering algorithm on a real-world dataset. Specifically, our preliminary experiments are on a large set of over $270,000$ photographs from Flickr for which reliable "ground truth" event annotations are available, via user tags linked to Yahoo!'s Upcoming event database[1] (Section 5).

We conclude with a discussion of the implications of our findings and directions for future work in Section 6.

2. RELATED WORK

We describe relevant related work in three areas: clustering large-scale high-dimensional data, event detection and tracking in news streams, and social media analysis. There are many existing approaches for clustering large-scale high-dimensional data [6], trading off runtime performance and clustering accuracy. Such scalable clustering approaches include Dignet [23], an incremental, single pass clustering algorithm, and, notably, BIRCH [27], an efficient clustering technique that uses a height-balanced tree of nodes storing sufficient statistics of the data they represent. The problem of clustering in high-dimensional spaces has given rise to many subspace clustering techniques [19]. Most relevant to our task, cluster ensemble approaches provide a robust, scalable, and efficient solution for clustering high-dimensional data, where subsets of features can be distributed across multiple clustering techniques [7, 22] (see Section 4 for further discussion).

The topic detection and tracking (TDT) event detection task [2] was studied in a notable collective effort to discover and organize news events in a continuous stream (e.g., newswire, radio broadcast) [3, 25, 15]. With an abundance of well-formed text, many of the proposed approaches (e.g., [26, 9]) rely on natural language processing techniques to extract linguistically motivated features. Makkonen et al. [17] extracted meaningful semantic features such as names, time references, and locations, and learned a similarity metric that combines these metrics into a single clustering partition. They concluded that augmenting documents with semantic terms did not improve performance, and reasoned that inadequate similarity functions were partially to blame. We show that, in our setting, clustering performance improves when we combine the variety of social media features judiciously.

Several efforts have studied how to extract high-quality information from social media sources [1, 4, 20, 14, 13]. Recent studies [10, 11] showed that social media document tags are accurate descriptors of content, and could be used to train a social tagging prediction system. Tags have also been used in conjunction with other context [14] to retrieve images of geography-related landmarks from Flickr. More directly related to our problem is the work of Rattenbury et al. [20], who analyzed the temporal usage distribution of tags to extract event semantics. However, the authors did not attempt to aggregate social media documents but instead they identify tags and their semantics.

3. PROBLEM DEFINITION

Given a set of social media documents, the problem that we address in this paper is how to identify events (e.g., President Obama's inauguration, or Madonna's October 6, 2008 concert in Madison Square Garden) that are reflected in the documents, as well as the documents that correspond to each event. We cast our problem as a clustering problem over social media documents (e.g., photographs, videos, social network group pages), where each document includes a variety of "context features" with information about the document. Some of these features (e.g., title, description, tags) are manually provided by users, while other features (e.g., upload or content creation time) are automatically generated.

Problem Definition. *Consider a set of social media documents where each document is associated with an (unknown) event. Our goal is to partition this set of documents into clusters such that each cluster corresponds to all documents that are associated with one event.*

As the definition of "event," we adopt the version used for the Topic Detection and Tracking (TDT) event detection task over broadcast news [24].

Definition. *An event is something that occurs in a certain place at a certain time.*

[1] http://upcoming.yahoo.com

As a distinctive characteristic, social media documents include a variety of *context features*, as mentioned above. The context features of a document are dependent on the type of document (e.g., a "duration" feature is meaningful for videos but not photographs). However, many social media sites share a core set of features. These features include: *author*, with an identifier of the user who created the document in question (e.g.,"kerrhyphen" is the author of a photograph that corresponds to our example "All Points West" event); *title*, with the "name" of the document (e.g., "DSC01325" for the same photograph); *description*, with a short paragraph summarizing the document contents (e.g., "radiohead performing"); *tags*, with a small set of keywords describing the document contents (e.g., "apw, All, Points, West"); *time/date*, with the time and date when the document was published (e.g., August 9, 2008);[2] *location*, with the location associated with the document (e.g., Jersey City, New Jersey). These complementary context features, collectively, will prove helpful to characterize social media document similarity and, in turn, to identify events and their associated documents, as we discuss next.

To exploit the various context features for our clustering task, we define a similarity metric for each feature, which should of course match the domain of the feature. Specifically, we represent each textual feature (e.g., title, description, tags) as a *tf.idf* weight vector and use the cosine similarity metric, as defined in [15], as the feature similarity metric. For time/date, an important feature in social media documents, we represent values as the number of minutes elapsed since the Unix epoch (i.e., since January 1st, 1970) and compute the similarity of two time/date values t_1 and t_2 as follows: if t_1 and t_2 are more than one year apart, we define their similarity as 0 (it is unlikely that the corresponding documents are associated with the same event in this case); otherwise, we define their similarity as $1 - \frac{|t_1 - t_2|}{y}$, where y is the number of minutes in a year. For location, another important feature in social media documents, we represent values as geographical coordinates (i.e., latitude-longitude pairs) and compute the similarity of two locations $\mathcal{L}_1 = (lat_1, long_1)$ and $\mathcal{L}_2 = (lat_2, long_2)$ as $1 - H(\mathcal{L}_1, \mathcal{L}_2)$, where $H(.)$ is the Haversine distance [21], a widely accepted metric for geographical distance.

After defining similarity metrics for the context features, we could cluster the social media documents using any of these features individually. As we will see in Section 5.2, such a clustering approach is not ideal, since it does not exploit the wealth of context features collectively. In the next section, we describe an approach to leverage these features in concert to produce high-quality clustering results.

4. CLUSTER ENSEMBLES

Ensemble clustering is a clustering approach that combines multiple partitions of a document set [22]. The advantage of using an ensemble approach is in the ability to combine different similarity metrics into the clustering process by learning a weighted similarity normalization technique. In this section, we explain the idea of ensemble clustering and show how we adapt it for the social media document representations described in Section 3.

[2]Often documents also include their capture or creation time/date (e.g., the time and date when a photograph was taken).

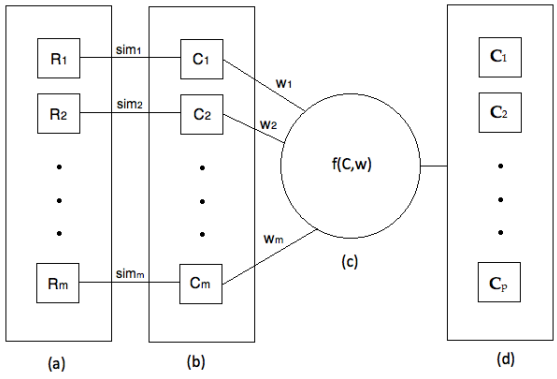

Figure 1: A conceptual diagram of an ensemble clustering process.

4.1 Ensemble Selection

The first step in any cluster ensemble algorithm is to select techniques for partitioning the data. These techniques, also referred to as *clusterers* ($C_1, \ldots, C_m$ in Figure 1(b)), produce mappings from documents to clusters. Each of these techniques should have a unique view of the data, or use a different underlying model to generate the data partition ($R_1, \ldots, R_m$ in Figure 1(a)). For our ensemble, we select clusterers that partition the data using different social media features and different similarity metrics. In particular, we have separate clusterers for features such as title keywords, description keywords, tag keywords, location coordinates, and time. Our clusterers also vary in the similarity metrics that they use. For each clusterer, we select an appropriate similarity metric based on the feature set (see Section 3).

Our ideal clustering solution should be scalable and not require *a priori* knowledge of the number of clusters, since social media sites are constantly evolving and growing in size. Therefore, traditional clustering approaches that require knowledge of the number of clusters, such as K-means and EM [6], are not suitable for this problem. Other alternatives such as scalable graph partitioning algorithms [12] do not capture the highly skewed event distribution of social media event data due to their bias towards balanced partitioning. In fact, we experimented with graph partitioning algorithms, but do not discuss their results further because of their poor performance for our task. Threshold-based techniques are preferable for our clustering task since they can be tuned using a training set and subsequently generalized to unseen data points. Hierarchical clustering algorithms [6], while relying on threshold tuning, are also not appropriate since they require a fully specified distance matrix, which does not scale to the large size of our data. Furthermore, online or incremental clustering algorithms, which are able to handle a constant stream of new documents, are also desirable in our social media setting, where new documents are continuously being produced.

Based on these observations, we propose using incremental clustering algorithms with threshold parameters that can be tuned in a principled manner during a training phase. Incremental clustering has been shown to be an effective technique for event detection in textual news documents (e.g., [25, 3]). Specifically, we use a single-pass algorithm with

centroid similarity, as follows: Given a threshold μ, a similarity function σ, and documents to cluster $d_1, \ldots, d_n$, the algorithm considers each document d_i in turn, and computes its similarity $\sigma(d_i, o_j)$ against each existing cluster centroid o_j, for $j = 1, \ldots, k$. (Initially, $k = 0$.) If there is no centroid whose similarity to d_i is greater than μ, we create a new cluster $k+1$ for d_i, with $o_{k+1} = d_i$ as its centroid. Otherwise, d_i is assigned to a cluster j with maximum $\sigma(d_i, o_j)$, and o_j is recomputed to reflect the new cluster contents. The centroid for a cluster of documents $\mathcal{S}$ is defined as $\frac{1}{|\mathcal{S}|}\sum_{d\in\mathcal{S}} d$. Depending on the document representation, the centroid is either the average *tf.idf* score per term (for textual features such as title, description, tags), the average time in minutes (for time/date), or the mid-point (for location) of all documents in $\mathcal{S}$.

To tune the clustering threshold for a specific dataset, we run each clusterer on a subset of labeled training data (see Section 5.1). We evaluate each clusterer's performance using a range of thresholds, and identify the threshold setting that yields the highest-quality partition according to a given clustering evaluation metric. Although several clustering evaluation metrics are available (see [5]), for brevity we focus on Normalized Mutual Information (NMI) [22, 18], an information-theoretic metric that was originally proposed as the objective function for cluster ensembles (see Section 6 for further discussion). NMI measures how much information is shared between actual "ground truth" events, each with an associated document set, and the clustering assignment. Specifically, for a set of clusters $C = \{c_1, \ldots, c_J\}$ and events $E = \{e_1, \ldots e_K\}$, where each c_j and e_k is a set of documents, and n is the total number of documents, $NMI(C, E) = \frac{I(C,E)}{(H(C)+H(E))/2}$, where $I(C, E) = \sum_k \sum_j \frac{|e_k \cap c_j|}{n} \log \frac{n \cdot |e_k \cap c_j|}{|e_k| \cdot |c_j|}$, $H(C) = -\sum_j \frac{|c_j|}{n} \log \frac{|c_j|}{n}$, and $H(E) = -\sum_k \frac{|e_k|}{n} \log \frac{|e_k|}{n}$.

The evaluation metric serves two important purposes in our ensemble approach. The first, as previously mentioned, is to select the most suitable threshold setting for each clusterer. The second is to assign a weight to each clusterer, indicating our confidence in its predictions. The weights are assigned during a training phase, and used to determine each clusterer's influence on the final ensemble prediction. By weighing each approach, we are able to determine how successful each metric is in capturing document similarity.

Once we select the best performing thresholds for all clusterers $C_1, \ldots, C_m$ (see Figure 1(b)), we set their weights $w_1, \ldots, w_m$ to equal their respective NMI scores, and then normalize the ensemble weights such that $\sum_{i=1}^{m} w_i = 1$. In the conclusion of the ensemble training phase, we have learned an optimal threshold for each clusterer, as well as a quality measure that will be used to weigh its decisions. With this information, we can partition a previously unseen set of social media documents and proceed to the ensemble prediction step.

4.2 Ensemble Prediction

When we reach the ensemble prediction step, we have carefully selected the clustering threshold for each technique, as well as a confidence weight associated with each technique. Given a set of documents, we use each technique to generate a clustering partition of this set. In the ensemble prediction phase, we develop a consensus mechanism for combining these individual partitions into a single clustering solution ($\mathcal{C}_1, \ldots, \mathcal{C}_p$ in Figure 1(d)).

Intuitively, each clusterer can be regarded as providing an expert vote on whether two documents belong in the same cluster. The consensus function we use is a weighted binary vote: For a pair of documents (d_i, d_j) and clusterer C, we define a prediction function $P_C(d_i, d_j)$ as equal to 1, if d_i and d_j are in the same cluster, or 0 otherwise. Then, we compute the consensus score for d_i and d_j as $\sum_C P_C(d_i, d_j) \cdot w_C$, where w_C is the weight of clusterer C.

We use the single-pass incremental clustering algorithm for combining the ensemble partitions into one solution, following the rationale of Section 4.1. The similarity function σ that we use in this step is the ensemble consensus function described above. The only required parameter is a similarity threshold μ, which we tune using a labeled subset of the data in the training phase, following how we tuned the clusterer thresholds (Section 4.1). At the conclusion of this step, we determine the document clusters $\mathcal{C}_1, \cdots, \mathcal{C}_p$ that represent the ensemble solution (Figure 1(d)).

5. EXPERIMENTS

We now report our experimental settings (Section 5.1) and results (Section 5.2).

5.1 Experimental Settings

We describe the data and methodology used for our experimental evaluation, along with the various baselines we consider.

Data: Our dataset consists of 270,425 Flickr photographs, taken between January 1, 2006, and December 31, 2008. Using the Flickr API,[3] we collected all photographs that were manually tagged by users with an event id corresponding to an event from the Upcoming event database. The Upcoming tags provide the "ground truth" for our clustering experiments (see Section 4.1). Each photograph corresponds to a single event, and each event is self-contained and independent of other events in the dataset. The dataset includes widely known events such as the "Bay to Breakers" race and "MacWorld Expo," as well as more obscure events such as the "Whitehaven Choir" fundraiser and Erin Antognoli's art show opening. Our dataset contains $9,515$ unique events, with an average of 28.42 photographs per event. The context features associated with each photograph include the title, description, tags, time/date of capture, and geo-coordinates. 38% of our data is tagged with geo-coordinate information. On this subset of the data, we perform reverse geo-coding using the Flickr API, to obtain a textual representation of the location of each photo, which we use as one clusterer in our experiments (see below).

To train our algorithm, we first order the photographs according to their upload time, and then divide them into three equal parts. We use the earliest two thirds of the data as training and validation sets, to tune the clusterer thresholds and assign their weights. The last third of the data is used as an independent test set, on which we report our results. We chose a time-based split since it best emulates real-world scenarios, where we only have access to past data with which we can train models to cluster future data.

Ensemble Setup: We use the Lemur Toolkit[4] to compute *tf.idf* vectors and the cosine similarity for the textual features (Section 3). Such features include the title key-

[3]http://www.flickr.com/services/api

[4]http://www.lemurproject.org

words (*Title*), description keywords (*Description*), tag keywords (*Tags*), time/date keywords (*Time/Date-Keywords*, treating the photograph capture time/date as a sequence of keywords), and location keywords (*Location-Keywords*, treating the reverse geo-coding of the location as a sequence of keywords). We also create a separate representation using all the above textual content in the document simultaneously (*All-Text*), as it is often the top performing approach in similar domains [17]. Finally, we create document representations using numeric time/date (*Time/Date-Proximity*) and location coordinates (*Location-Proximity*), with the similarity metrics described in Section 3.

Model Selection and Evaluation: We use Lemur's online clustering implementation to cluster the training data according to each representation. We tune the clustering threshold for each clusterer (Section 4.1) using the first half of the training data and considering thresholds in the range $[0, 1]$, with 0.05 increments. We compute the NMI value for each clusterer and select the threshold of the best performing clusterer for each representation. The second half of the training data serves as a validation set, which we use to tune the final prediction threshold (see Section 4.2). For the ensemble weights, we use the NMI scores of the partitions created by applying the clusterers to the validation set.

Techniques for Comparison: We consider all individual clusterers as baseline approaches, namely, All-Text, Title, Description, Tags, Time/Date-Keywords, Location-Keywords, Time/Date-Proximity, and Location-Proximity. We also experiment with *Multi-Sim*, an ensemble of textual and numeric document representations, including All-Text, Title, Description, Tags, Time/Date-Proximity, and Location-Proximity.

5.2 Experimental Results

We first report the performance of the best *individual* clusterers, according to their NMI scores. Figure 2 shows the NMI of each technique for the best clustering threshold (Section 4.1) over the training set. The Tags clusterer had the highest NMI on the training set, compared with the other individual clusterers, with All-Text being a close second. The Description clusterer, on the other hand, exhibits the worst performance in the ensemble, due to many irrelevant descriptions present in the data, notably "n/a," and URLs to external sites. Titles suffers from a similar drawback, since many users do not change the automatically generated file names for their photos, and the Flickr upload interface uses these file names as the default photograph titles. The performance of Time/Date-Proximity and Location-Proximity is better than that of Time/Date-Keywords and Location-Keywords, supporting our argument for using natural numeric similarity metrics for time and location (Section 3).

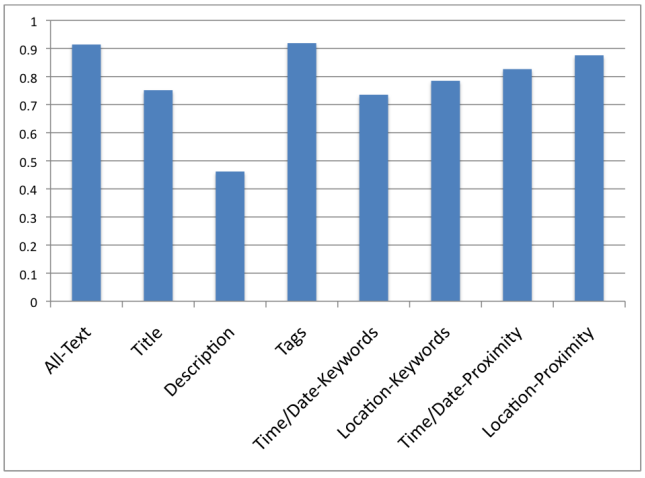

Figure 2: NMI of individual clustering techniques over the training set.

Algorithm	NMI
Multi-Sim Ensemble	0.933
All-Text Clusterer	0.926
Tags Clusterer	0.924

Table 1: NMI of the ensemble technique and the best individual clustering techniques over the test set.

Next, we compare the performance of our ensemble technique, Multi-Sim, against the Tags clusterer, which is the clusterer with the best performance over the training set. We also report the performance of All-Text, which was a close second to Tags over the training set. Table 1 reports the NMI scores for the three techniques over the *test* set. As expected, the ensemble method, which considers all document features collectively, outperforms Tags, which only considers one document feature. Interestingly, the Multi-Sim ensemble method, which combines the clustering evidence from the various features in a principled manner using natural similarity metrics, also outperforms All-Text, which also considers all features but without a principled combination strategy. Beyond the NMI results, we observed that the clusters created by the ensemble method are more homogeneous—with more documents corresponding to each event spread over fewer clusters—than the Tags and All-Text clusters. Overall, our preliminary experiments are encouraging, and we will continue to further develop and evaluate our general approach along a number of dimensions, as we discuss next.

6. CONCLUSIONS AND DISCUSSION

In this paper, we presented a novel approach for identifying events and their associated social media documents, by combining multiple context features of the document. We discussed and experimented with a weighted cluster ensemble algorithm that exploits the multiple features (e.g., title, description, tags, location, time) simultaneously. Preliminary experiments showed that this approach is promising, so we plan to explore and refine the key design choices of the initial work that we have reported here.

Our choice of clustering evaluation metric plays an important role throughout the ensemble algorithm (Section 4.1). We use this metric to tune the clusterers for the ensemble, assign weights, and evaluate the final result. Our initial choice of evaluation metric, namely NMI, reflects the clustering properties that we wish to capture. In particular, we want our clusters to be homogeneous, and include all members of each class in a single cluster. NMI balances these two constraints, but it is difficult to interpret in practical terms (e.g., the metric does not directly reflect the fraction of document pairs that were correctly clustered together). There-

fore, we plan to experiment with other evaluation metrics (see [5]) beyond NMI.

In addition to using the clustering evaluation metric as a proxy for the ensemble weights, we will consider weighing schemes that treat the ensemble training step as a classification task. Using the classification framework, we can employ techniques such as fitting logistic regression weights, or incrementally penalizing weights following the weighted majority algorithm [16]. An important property of the ensemble that should be considered during the selection and weight assignment process is the diversity of the individual clusterers. Intuitively, we want each clusterer to provide a different view of the data. We would not, for example, want to include two clusterers that predict according to numeric time difference, since it would give the algorithm a false indication that time is twice as important for the final prediction. We plan to explore ensemble diversification techniques (see [8]), during both the ensemble selection and the weight assignment stages.

Our efforts here are, therefore, a first step in a framework that we hope to develop into a robust set of tools for extracting event information from social media content.

7. ACKNOWLEDGMENTS

This material is based upon work supported by a generous research award from Google as well as by the National Science Foundation under Grant CNS-0717544. We also thank Luis Alonso, Krzysztof Czuba, and Julia Stoyanovich for their feedback on our work.

8. REFERENCES

[1] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne. Finding high-quality content in social media. In *Proceedings of the First ACM International Conference on Web Search and Data Mining (WSDM'08)*, Feb. 2008.

[2] J. Allan. Introduction to topic detection and tracking. In J. Allan, editor, *Topic Detection and Tracking – Event-based Information Organization*, pages 1–16. Kluwer Academic Publisher, 2002.

[3] J. Allan, R. Papka, and V. Lavrenko. On-line new event detection and tracking. In *Proceedings of the 21st ACM International Conference on Research and Development in Information Retrieval (SIGIR'98)*, pages 37–45, 1998.

[4] S. Amer-Yahia, M. Benedikt, L. V. S. Lakshmanan, and J. Stoyanovich. Efficient network aware search in collaborative tagging sites. *PVLDB*, 1(1):710–721, 2008.

[5] E. Amigó, J. Gonzalo, J. Artiles, and F. Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 2008.

[6] P. Berkhin. Survey of clustering data mining techniques. Technical report, Accrue Software, San Jose, CA, 2002.

[7] C. Domeniconi and M. Al-Razgan. Weighted cluster ensembles: Methods and analysis. *ACM Transactions on Knowledge Discovery from Data*, 2(4):1–40, 2009.

[8] X. Z. Fern and C. E. Brodley. Random projection for high dimensional data clustering: A cluster ensemble approach. In *Proceedings of the Twentieth International Conference on Machine Learning (ICML'03)*, pages 186–193, 2003.

[9] V. Hatzivassiloglou, L. Gravano, and A. Maganti. An investigation of linguistic features and clustering algorithms for topical document clustering. In *Proceedings of the 23rd ACM International Conference on Research and Development in Information Retrieval (SIGIR'00)*, pages 224–231, 2000.

[10] P. Heymann, G. Koutrika, and H. Garcia-Molina. Can social bookmarking improve web search? In *Proceedings of the First ACM International Conference on Web Search and Data Mining (WSDM'08)*, Feb. 2008.

[11] P. Heymann, D. Ramage, and H. Garcia-Molina. Social tag prediction. In *Proceedings of the 31st ACM International Conference on Research and Development in Information Retrieval (SIGIR'08)*, July 2008.

[12] G. Karypis, R. Aggarwal, V. Kumar, and S. Shekhar. Multilevel hypergraph partitioning: Application in VLSI domain. In *Proceedings of the 34th ACM Conference on Design Automation (DAC'97)*, pages 526–529, 1997.

[13] L. Kennedy and M. Naaman. Less talk, more rock: automated organization of community-contributed collections of concert videos. In *Proceedings of the World Wide Web Conference (WWW'09)*, pages 311–320, 2009.

[14] L. Kennedy, M. Naaman, S. Ahern, R. Nair, and T. Rattenbury. 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'07)*, pages 631–640, 2007.

[15] G. Kumaran and J. Allan. Text classification and named entities for new event detection. In *Proceedings of the 27th ACM International Conference on Research and Development in Information Retrieval (SIGIR'04)*, pages 297–304, 2004.

[16] N. Littlestone and M. K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.

[17] J. Makkonen, H. Ahonen-Myka, and M. Salmenkivi. Simple semantics in topic detection and tracking. *Information Retrieval*, 7(3–4):347–368, 2004.

[18] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[19] L. Parsons, E. Haque, and H. Liu. Subspace clustering for high dimensional data: a review. *SIGKDD Explorations Newsletter*, 6(1):90–105, 2004.

[20] T. Rattenbury, N. Good, and M. Naaman. Towards automatic extraction of event and place semantics from Flickr tags. In *Proceedings of the 30th ACM International Conference on Research and Development in Information Retrieval (SIGIR'07)*, pages 103–110, 2007.

[21] R. W. Sinnott. Virtues of the Haversine. *Sky and Telescope*, 68:159, 1984.

[22] A. Strehl, J. Ghosh, and C. Cardie. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3:583–617, 2002.

[23] S. C. A. Thomopoulos, D. K. Bougoulias, and C.-D. Wann. Dignet: an unsupervised-learning clustering algorithm for clustering and data fusion. *IEEE Transactions on Aerospace Electronic Systems*, 31:21–38, Jan. 1995.

[24] Y. Yang, J. Carbonell, R. Brown, T. Pierce, B. T. Archibald, and X. Liu. Learning approaches for detecting and tracking news events. *IEEE Intelligent Systems Special Issue on Applications of Intelligent Information Retrieval*, 14(4):32 – 43, 1999.

[25] Y. Yang, T. Pierce, and J. Carbonell. A study on retrospective and on-line event detection. In *Proceedings of the 21st ACM International Conference on Research and Development in Information Retrieval (SIGIR'98)*, pages 28–36, 1998.

[26] K. Zhang, J. Zi, and L. G. Wu. New event detection based on indexing-tree and named entity. In *Proceedings of the 30th ACM International Conference on Research and Development in Information Retrieval (SIGIR'07)*, pages 215–222, 2007.

[27] T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH: An efficient data clustering method for very large databases. In *Proceedings of the 1996 ACM International Conference on Management of Data (SIGMOD'96)*, pages 103–114, 1996.