Modeling Event Importance for Ranking Daily News Events

Vinay Setty[†], Abhijit Anand[‡], Arunav Mishra[†], Avishek Anand[‡]

[†]Max Planck Institut für Informatik
Campus E1 4,
66123 Saarbrücken, Germany
{vsetty,amishra}@mpi-inf.mpg.de

[‡]L3S Research Center
Leibniz University of Hannover
30167 Hanover, Germany
{aanand,anand}@L3S.de

ABSTRACT

We deal with the problem of ranking news events on a daily basis for large news corpora, an essential building block for news aggregation. News ranking has been addressed in the literature before but with individual news articles as the unit of ranking. However, estimating event importance accurately requires models to quantify current day event importance as well as its significance in the historical context. Consequently, in this paper we show that a cluster of news articles representing an event is a better unit of ranking as it provides an improved estimation of popularity, source diversity and authority cues. In addition, events facilitate quantifying their historical significance by linking them with long-running topics and recent chain of events. Our main contribution in this paper is to provide effective models for improved news event ranking.

To this end, we propose novel event mining and feature generation approaches for improving estimates of event importance. Finally, we conduct extensive evaluation of our approaches on two large real-world news corpora each of which span for more than a year with a large volume of up to tens of thousands of daily news articles. Our evaluations are large-scale and based on a clean human curated ground-truth from Wikipedia Current Events Portal. Experimental comparison with a state-of-the-art news ranking technique based on language models demonstrates the effectiveness of our approach.

1. INTRODUCTION

The primary consumption of news is now increasingly online and consequently, the global news industry has witnessed a drastic shift of its focus from traditional print media to publishing digital content. The vast amount of online information being generated from various news agencies, independent providers, and sometimes the end users themselves has made it difficult to retrospect and develop an holistic understanding of daily news events. Thus, to deal with this *information overload*, there is an increasing need for more meaningful organization of online news which are typically in the form of online news articles reporting *stories* describing an event.

One increasingly popular solution to tackle information overload is to build *news aggregators* which leverage automated methods to organize and rank news events. For example, commercial news aggregation systems like *Google News* and *Bing News* make use of various ranking, clustering, and personalization methods to improve the presentation of news stories to the users. On the other hand, there are also manually curated online portals like *Wikipedia Current Events Portal* (WCEP)[1] as a community effort to organize information by first listing seminal real-world events, and then linking them with online news articles that centrally describe the events. For example, in Fig. 1a an excerpt of daily news from WCEP shows a topic "*2014 pro-Russian unrest in Ukraine*" with corresponding news stories from *Kyiv Post* and *CNN* that mention events related to the topic.

Our work in this paper is motivated by the observation that both automated aggregation and manual curation of news events need to solve two fundamental tasks: mining news events and modeling their importance. Due to the vast amounts of online news, these tasks inevitably need to process millions of news stories to provide comprehensive coverage of all daily news events. Motivated by this, we tackle the problem of mining news events given large-scale news corpora and then model their importance, which can be used to build news aggregation systems.

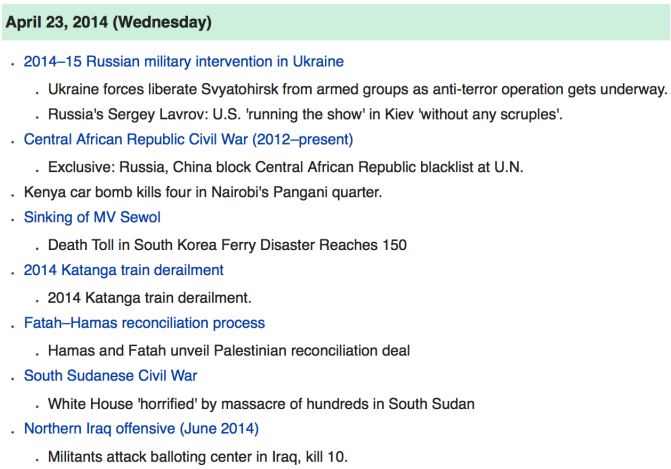
In the realm of information retrieval, a lot of attention [30, 6, 13] has been given to generate a ranking for news articles. However, it is not straight forward to extend these methods to a setting where the goal is to model the importance of wide variety of news events reported by large number of news articles. One prominent issue is that large number of online news articles obtained from diverse sources on any given day can be highly redundant. For example, *Gdelt*[2] collects news from 6K sources from over 150 countries. Therefore, using news stories as the unit of ranking in such diverse corpora leads to ranking redundant events. At the same time adopting stringent redundancy control techniques may lead to the loss of crucial information such as popularity of news events, diversity and authority of event reporting sources. Thus, it is crucial to design strategies with events as the unit of ranking that are represented as a cluster of news articles. Further, in order to appropriately rank mixture of events that may have taken place in past or is relatively recent, we

[1]http://wikipedia.org/wiki/Portal:Current_events
[2]http://www.gdeltproject.org

WSDM 2017, February 06-10, 2017, Cambridge, United Kingdom

DOI: http://dx.doi.org/10.1145/3018661.3018728

(a) Wikipedia Current Events Portal daily summary



(b) Top news events and topics from our approach

Figure 1: Daily Event Summaries for 23^{rd} April 2014

have to consider their popularity and historical importance. For simplicity we refer to a cluster of news story articles as an *event* in the rest of the paper.

Popularity of a news event has been quantified as frequency of its reporting by the news media before. This has been exploited by many news ranking and personalization approaches that cluster news stories using their textual content and then use sizes of these clusters as a proxy for the *popularity* of news events [22, 24]. However, clustering using purely textual features introduces a large number of false positives, i.e. stories which are partially related but do not actually report the event. As an example, a cluster of stories reporting the "*Syrian Civil War*" may still contain stories which report the "*US Presidential Elections*" where one of the presidential candidates mentions the same civil war. Since most of the subsequent ranking computation is based on the cluster size, these false positives affect the downstream ranking.

While these popularity cues reliably reflect the current day importance of an event, they are agnostic to the historical importance. Events often do not exist in isolation, instead it is well known that events can relate to long-running seminal events. For example for the topic *Egyptian crisis*, early protests, and Mubarak's resignation are related events. Therefore, in addition to popularity cues, it is crucial to take historical evidence of event importance into account. However, to the best of our knowledge none of the existing news ranking solutions systematically explore the models to consider the historical importance of news events.

Finally, one of the bottlenecks in the area of news ranking as acknowledged in [38] is the lack of benchmarks to evaluate the proposed solutions. State-of-the-art news ranking techniques are mostly evaluated by humans. While this is a reliable method, it limits the scale of evaluation. To address these issues, our main contributions in this paper are:

1. We propose an effective method for *mining news events* from large news corpora employing a clustering technique which exploits a wide-variety of textual, semantic and temporal features. Further we propose post-processing techniques on the event clusters to *avoid false positives*, consequently improving the event importance estimates (cf. Sections 4 and 5).
2. We propose two techniques to consider *historical significance* of events: Firstly, we use *event chaining* across sequences of days to measure historical importance of events in the recent past. Secondly, we make use of already existing human definitions of topics in the form of Wikipedia Event pages to *assign topics* to events and *derive their historical properties* (cf. Section 6).
3. We tackle the challenge of large-scale evaluation for news ranking by proposing a *benchmark adapted from daily top news curated in WCEP*. We then validate the effectiveness of our approach using two large news corpora – STICS [15] (2 mi. stories) and GDELT (8 mi. stories). Using this benchmark we show that our methods outperform a state-of-the-art news ranking technique based on Language Models and Temporal Profiles [22] (cf. Section 7).

As an anecdotal evidence, we can see in Fig. 1b, given news articles published on 23rd April 2014 from the Gdelt dataset (about 26K), our method mines events, labels them with appropriate topics, and also assigns importance values so as to select top news events which are comparable to WCEP summary shown in Fig. 1a.

2. RELATED WORK

We first discuss the news ranking techniques in the literature and explain how our approach is different from all of them. Next, as our historical feature derivation of news events overlap with the tasks of news linking and topic detection and tracking (TDT), we briefly review the state-of-the-art in these areas. Finally, mining news events from news corpora overlaps with news clustering and summarization works as well. Consequently, we place our contributions in the perspective of these areas.

2.1 News Ranking

Ranking stream of news: Many ranking problems on news corpora mostly deal with ranking a stream of news stories incrementally [6, 14]. Del Corso et al. [6] were the first to propose the problem of ranking news on incremental news corpora. They proposed desirable properties which should be exhibited by a good news ranking technique and also proposed a model which exploits a virtual linking relationship between news stories and sources based on their

news posting process. Finally they proposed time-aware approaches which build on the story-source reinforcement model. Gwadera and Crestani in [14] on the other hand mine sequential patterns in a time window of streaming news stories to rank them based on story timeliness and content authority. Both these methods are based on a simplistic clustering substrate, using only textual features, and are difficult to generalize due to a large parameter space. Moreover, since these approaches do not consider historical features the importance of long-running topics may be undermined.

Ranking using external sources: There are works which consider external sources to derive popularity of news. For example, the Top Story Identification Task (TSIT) of the TREC blog track [30] was dedicated to rank news based on their perceived popularity on an external collection namely the Blogosphere. In the state-of-the-art approach for TSIT [22], the authors first cluster the entire collection into a fixed number of topics. Next, they proposed a language modeling approach to combine both the cluster-based importance of an article and its temporal profile contribution to obtain the final score of the article on which it is ranked. The temporal profile derived from Blogosphere to some extent offers the historical evidence of the news stories, but still fails to consider the rich historical data that can be obtained by linking the news stories to long running topics from Wikipedia. Nevertheless, since this method is closest to our approach we adapt this method into our event ranking problem as a baseline.

There are also works which use external sources to rank *news topics* based on media focus and user attention to news articles using aging theory [38, 19]. Wang et al. [38] maintain topics by incrementally clustering news articles when they are discovered. These topics are then ranked by freshness of media focus and user attention omitting history of news topics. Moreover, as acknowledged by the authors themselves in these papers, the results they obtain are based on controlled user studies which is not a good way to evaluate a ranking method. In this paper we propose a method to evaluate news ranking techniques using a high-quality gold standard from Wikipedia Current Events Portal.

Query-based news ranking: Query-based works on personalization [24, 23], vertical search [26] and news result clustering [37] are also partially related. Query-based news search diversification is yet another related area of research [36]. These papers address the problem of ranking news articles given a user query, which is different from the problem of ranking daily news events independent of any queries. In [37], LSH is used for clustering news search results, we use LSH to mine news events. In [33], query-based exploration of interesting time points in news archives is proposed.

2.2 News linking and organization

Other potentially related works are in the domain of news linking and organization [29, 34, 27, 35, 28]. These works are mostly dedicated to identify news topics and organize them for exploration without ranking them.

Topic detection and tracking (TDT): The broad area of TDT [1, 29] was intended to investigate methods for automatically detecting breaking news stories and organizing news stories by events. However, all these approaches work on exploiting relatedness between news articles towards a particular mining task.

News Event Canonicalization: Kuzey et al. [21] use a graph coarsening over event graphs to organize news articles into a causal network of events along with canonicalization of events. Although our event canonicalization task varies slightly, we use a baseline inspired from [21] for comparison. Fetahu et al. [8] also try to analyze news articles for recommending facts to Wikipedia entity pages. Unlike our canonicalization task, they are interested in identifying articles which are novel fact containers for improving coverage in incomplete *entity pages*. Mishra et al. [28] propose a technique to link Wikipedia event excerpts to online news articles, while in this paper we are interested in the opposite. Similarly Fetahu et al.[9] provide a method for finding and updating the citations for news statements in Wikipedia.

2.3 News Event Mining and Summarization

News event mining: There are works that cluster news stories [37, 10] but their goal is not mining news events. There are also standard clustering techniques such as K-means and DB-Scan which are widely used for clustering text and news stories. While any of these state-of-the-art text or news clustering techniques can be used to group news articles, as mentioned earlier these technique can introduce false positives and mislead event mining. Recent work on so called "temponym" mining [20] identifies and resolves textual phrases with temporal scope. However, it is not specific to events.

News Summarization: News topic summarization has been widely explored in the literature before [3, 5, 32, 11]. These works tackle a specific task of summarizing news topics, rather than ranking them. Another related area is Temporal Summarization Task, one of the TREC tracks. The goal of this task as defined in [2] was to develop systems for efficiently monitoring the information associated with an event over time. Specifically, they were interested in developing systems which can broadcast short, relevant, and reliable sentence-length updates about a developing event. The goal of this task is orthogonal to the problem of news ranking. Similarly, in [18], Kabadjov et.al. propose a multilingual, multi-document news summarization system with a goal to capture the popularity of a news event in a given small time window.

In summary, none of the earlier approaches consider ranking news with events as the unit of ranking. Moreover, popularity estimation of events based on purely textual features makes them susceptible to false positives which in turn affects ranking. In addition, historical information is rarely used and in a limited manner which does not truly reflect the importance of top news events. Finally, there is a lack of large-scale news ranking evaluation in the literature due to the lack of groundtruth data. Our contributions in this paper focus mainly on addressing these limitations.

3. PROBLEM DEFINITION

In this section we introduce the terminology used in this paper and formalize the problem of ranking daily news events. Then, we briefly describe the high level ranking approach.

We first introduce the terminology used in this paper, borrowed from the TDT community [1]. We operate on a document collection $\mathcal{D}$ containing news articles published by several news agencies.

News story: Each *news story* $d \in \mathcal{D}$ is a news arti-

cle document. The set of stories published on a day t is represented as $\mathcal{D}^t \subseteq \mathcal{D}$.

News event: A *news event* c, on a day t is reported through a cluster of news stories $d \in c \subseteq \mathcal{D}^t$. We refer to c as a cluster of stories associated with a news event and is quantified with a date t as c^t when necessary.

News topic: Each daily news event c might be a part of a *long-running news topic* σ belonging to a universal set of topics Ξ. These topics are derived from event pages in Wikipedia for e.g. "Syrian Civil War". Note that two events c^{t_1} and c^{t_2} appearing on two different days t_1 and t_2 can be related to the same topic σ. Therefore, topics are not specific to any particular day. However, each topic $\sigma \in \Xi$ has a temporal scope $t_b(\sigma)$ and $t_e(\sigma)$ representing two dates when the topic σ first appeared and last observed respectively.

The *daily news ranking problem* aims to find a ranking of k most *important* news events on a given date t, given a set of news stories $\mathcal{D}^t$. Our definition of importance is based on both current day importance and historical significance of a news event. Since the input to our problem is a set of news stories, in order to derive necessary features for computing importance, we first need to find a coherent group of news stories that represent concrete news events. Additionally, we need to identify $\{c \in \sigma\}$ which are a part of a long-running topic σ and *assign* events with the topic σ to obtain historical context of the news events.

We approach the news ranking problem as a *Learning-to-Rank* task which has been shown to learn a sophisticated ranking model combining diverse set of features [25]. Specifically, we use *SVMRank* [17] which is a pairwise learning to rank approach. Note that although our ground truth data from WCEP does not induce a ranking yet we require a ranked output. We introduce relevance levels among the relevant clusters by soft-labeling the larger event clusters with higher relevance values than the smaller ones. This is done for two reasons: (1) the pairwise learning to rank approach can now learn preferences even among the relevant events, (2) larger clusters have a lower *error probability* of event representation than smaller clusters thereby enabling us to learn a discriminative model.

4. MINING DAILY NEWS EVENTS

In order to rank at an event granularity, we first need to mine events from the news collection. Since, important news events are often reported by multiple news sources, the most natural way to mine them is to cluster similar news stories. On the textual content level, stories reporting the same event share similar keyphrases. Some of these phrases might refer to entities involved in the event, i.e. people, locations and organizations. Also, entities might be mentioned in different surface forms in the textual content of similar stories. Consider three news story headlines (a) How POTUS, FLOTUS and Zuckerberg Celebrated Star Wars Day? (b) Michelle Obama Dances Real Cool To Celebrate 5 Years Of "Let's Move!" (c) Barack and Michelle Obama Adorably Dance With Stormtroopers, R2-D2 for Star Wars Day.

In such scenarios, vanilla text clustering using a bag of words model might either result in large number of false positives putting stories (a) and (b) in the same cluster, or might not capture similarity due to dissimilarity in the surface forms, i.e. disregarding the similarity between (a) and (c). Additionally they also share phrases salient to the event that are not entities, i.e. "Star Wars Day". Consequently, we model a news article as a bag of entities and phrases and further canonicalize candidate entity mentions using established named-entity disambiguation approaches [16, 7].

Given a news story $d \in \mathcal{D}_t$ in the news collection of a given day t, it is represented by the following features: (1) A bag of entities $\mathcal{E}(d)$ that contains disambiguated names of entities. (2) A bag of shingles $\mathcal{S}(d)$ that contains a unique set of n-gram shingles derived from the process of w-shingling typically used for measuring document similarity.

Two news stories reporting the same news event may overlap in text, entities or both. Therefore, we treat entities and shingles equally and combine them into a single bag $\mathcal{F}(d) = \mathcal{E}(d) \cup \mathcal{S}(d)$ using a multiset union. Then we compute the distance between two documents using the weighted Jaccard distance metric as below:

$$\delta(d, d') = 1 - \frac{\sum_{e \in \mathcal{F}(d) \cap \mathcal{F}(d')} weight(e)}{\sum_{e \in \mathcal{F}(d) \cup \mathcal{F}(d')} weight(e)} \quad (1)$$

Where, $weight(e)$ is the frequency of unique entities and shingles e in the news story.

A key aspect of our problem is the inability to accurately determine the true number of events on a given day. This means that clustering approaches like k-means that are sensitive to the number of clusters are not applicable. Other variants that try to automatically estimate the number of clusters prior to clustering are not scalable to our input sizes. Typically, the true distribution of cluster sizes has a long-tail with the majority of news events being outliers (only reported as a single story). Density based clustering approaches like DB-Scan that try to account for outliers during clustering are desirable in such a scenario. However, such approaches are often sensitive to density-based parameters that are often hard to estimate. Hence, our clustering algorithm needs to: (1) efficiently process high dimensional large-scale data, (2) carefully identify all news events irrespective of their sizes, and (3) takes minimal number of parameters.

Consequently, we utilize a nearest-neighbor-based clustering technique tailored to meet the desiderata while relying only on a single parameter ϵ (also used in DB-Scan but easier to estimate) that defines the *neighborhood*. There are neighborhood distribution analysis techniques using which ϵ can be easily chosen [31]. Computing ϵ neighborhood in a large-scale corpus with high dimensionality can be prohibitively expensive. Therefore, it is essential to consider efficient techniques to compute distance values and nearest neighbors. To address this problem we resort to Locality Sensitive Hashing (LSH) with min-wise independent permutations [12] which is known to efficiently handle high dimensions and provide provably scalable approximation of the Jaccard distance $\delta(d, d')$ and efficiently find similar stories to any given news story which can be considered as nearest neighbors. A similar technique is used in [37] in a different context for clustering news search results, which is different from mining news events from heterogeneous news sources.

The clustering algorithm iterates through each news story and retrieves the corresponding stories within ϵ neighborhood using precomputed LSH buckets. Using these neighbors a new cluster c_i is created. Note that unlike DB-Scan we do not check if there are minimum number of neighbors ($minPts$) to form a cluster. Further, if any of these neighbors are already assigned to a different cluster c_j, then the story is reassigned to the cluster that maximizes the cluster

cohesiveness given in Definition 1. This step is crucial as it considers all potential clusters with a goal to maximize the cluster cohesiveness of the selected cluster.

DEFINITION 1 (CLUSTER COHESIVENESS). *We define the cohesiveness of a cluster c as the average pair-wise distance between each news story in a cluster*

$$\phi(c) = \frac{|c|}{\sum_{d \in c} \sum_{d' \in c, d \neq d'} \delta(d, d')} \quad (2)$$

5. IMPROVED POPULARITY ESTIMATION OF NEWS EVENTS

In this section we consider the features which determine the popularity of an event. Intuitively, the size of the cluster is a good indicator of popularity of an event. However, in practice, cluster size alone is not sufficient and we need to consider sub-clusters of stories which are coherent. We further enrich the popularity using the source-based features consisting of two parts: Source Diversity and Source Authority. In the rest of this section we define these features more formally and provide techniques to compute them.

5.1 Improving Cluster Size Estimate

In order to compute coherent sub-clusters we first consider the cluster radius and centroid defined in Definition 2 which encodes the spatial span of the cluster. A cluster with a small radius and large size intuitively represents a coherent event and is potentially important. We further consider the cluster cohesiveness feature defined in Definition 1, as it quantifies the mutual similarity of the stories within a cluster. However, it is not hard to see that both cluster radius and cluster coherence features can be significantly affected by the presence of outliers. These outliers could be entirely unrelated to the event or a result of the concept drift and hence undesirable. For example, a news event about a *blast in Afghan capital Kabul* could share many entities and keyphrases with a news event about *election in Afghanistan.* Even though our clustering technique described in Section 4 is designed to maximize cluster coherence and eliminate outliers, some outliers may still remain in the clusters. This is because of the inherent geometry of the cluster, inflicted partially by the nearest-neighbor computation and cluster merging.

DEFINITION 2 (CLUSTER CENTROID AND RADIUS). *The centroid Γ_c of a cluster c is the story which has the minimum maximum distance to all the existing stories in the cluster. The overall radius ρ_i of c with a centroid Γ_c is the distance between the centroid and the farthest point from it in the cluster.*

$$\Gamma_c = \arg\min_{d \in c} \{\forall d' \in \{c \setminus d\} : \max \delta(d, d')\} \ , \ \rho_i = \max_{d \in c} \delta(d, \Gamma_c)$$

To address these issues and to determine a topically coherent and cohesive sub-cluster we introduce the concept of MAXIMUM SUB-CLUSTER DENSITY. The goal is to choose a sub-cluster with the highest density by omitting the outliers. To do this, we consider sub-clusters of increasing radius sizes and choose a sub-cluster which maximizes the following sub-cluster density. We first introduce radius at size k, with ρ_k as the radius of the sub-cluster containing k nearest neighbors of the centroid. Then the goal is to find a sub-cluster which maximizes the ratio k/ρ_k. The maximum sub-cluster density is given by the sub-cluster with the highest sub-cluster density denoted as ψ_{max}.

The sub-cluster with ψ_{max} and radius ρ_{max} is considered for computing the *effective size*, $S_{eff} = \text{argmax}\, \psi_{max}$. We consequently use the sub-cluster density ψ_{max} and radius ρ_{max} of the selected sub-cluster as structural features. Note that the computation of ρ_{max} and ψ_{max} is fairly efficient given that the cluster items are ordered according to their distance from the centroid.

5.2 Source Diversity

We assume that relying only on structural features may be misleading, as we deal with news corpora collected from a wide variety of sources and geographic locations. We assume that the input news corpora do not cover all geographic locations, topics and languages uniformly. As a result we can expect bias in the input corpus to news from certain geographic locations like North America and Britain. Moreover, some topics might be highly localized to certain geographic locations such as politics, sports and economy.

We tackle the problem of such a *collection bias* by computing a diversity score for each cluster. We observe that such collection bias is typically pronounced in the news source (website or reporting organization). For example, a news story about a Baseball game is typically local to U.S.A. In this regard, we operate on a manually constructed mapping of news sources to their respective countries, and compute the coverage of an event in the country space. The country coverage *cc* of each event is used to compute a NORMALIZED DIVERSITY score for each cluster as: $\frac{cc}{log(1+S_{eff})}$. The logarithmic discounting for size normalization is employed to avoid perfect diversity for small clusters.

5.3 Source Authority

It is known that news from authoritative sources are preferred and deemed to be more important than other sources. Typically, authority based features are derived using link analysis techniques. However, news stories have barely any links pointing to them especially when we deal with pure news collections rendering link analysis techniques [19] less effective. We in turn estimate the authority of news sources based on the importance of these news sources in Wikipedia. We extract all possible news citations from Wikipedia and construct a probability distribution based on their frequencies. There are other methods such as Alexa rankings of top news domains to compute features similar to authority. However, Alexa classifies domains such as reddit.com and weather.com as news domains as well. We address this issue by building features based on a distribution derived from reliable and more relevant news sources cited in Wikipedia by human editors.

6. HISTORICAL IMPORTANCE OF NEWS EVENTS

The importance of a news event or news topic taking only information about the current day is agnostic to the history of the event. Thus, ranking events disregarding history might lead to lack of continuity of important events which have been in the top news of the recent past. Important news events are characterized by outbreak, sustained interest for a while and finally decline [1]. Some news topics are

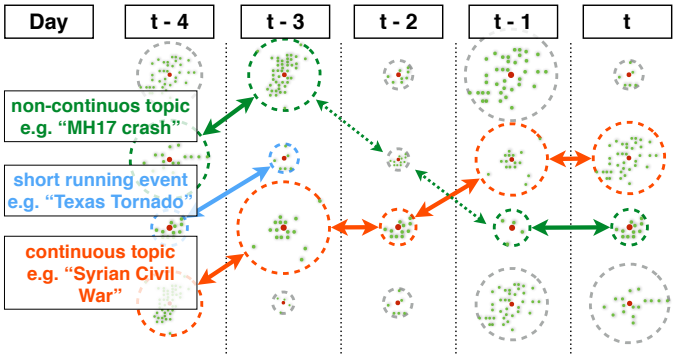


Figure 2: Example for cluster chaining

long-running like wars and conflicts, and stay relevant in the news for a longer period of time while some events like accidents, hijackings and natural calamities are typically short lived. Regardless of the nature of the event, the challenge of attributing for history in ranking events is two fold – (1) to detect and consider the historical significance of an event and (2) to adjust the historical contribution of that event depending on its age using a decay function.

We consider three types of historical features to account for the role of history and to address the above challenges – *previous day similarity*, *cluster chaining* and *event canonicalization*. In the rest of this section we introduce the notion of a cluster from a particular day t with the superscript c^t. Since a cluster may represent a long-running topic or a short event, we use event and topic interchangeably in this section.

6.1 Cluster Chaining

News events often persist across days. Short lived events such as moderately severe typhoons, minor disasters etc. persist for short periods in the top news. Such events may in future develop to become long-running topics. To discover short-running events we connect clusters on consecutive days belonging to the same event. We denote this as the previous day similarity as $psim(c) = \text{argmax}_{c^{t-1}}\{\delta(\Gamma_c, \Gamma_{c^{t-1}})\}$. For example, in Fig. 2, the news event about Texas tornado is quantified with $psim(c)$ as a sporadic event.

Based on the fact that an ongoing event leads to topically similar clusters, we examine clusters on adjacent days for inter-cluster similarity. We create a chain of clusters by connecting clusters most similar on adjacent days initiated from c, and continue until the previous day cluster similarity $psim(c)$ is below a certain acceptable threshold θ. For example, in Fig. 2, while the topic "Syrian Civil War" starting from day t is continuous until day $t-4$, for topics like "MH17 crash" the chain breaks after day $t-1$, since the previous day clusters are not sufficiently similar. We assume the cluster similarity to be zero when the cluster similarity $psim(c) < \theta$ and that marks the end of the chain. Intuitively, the length of the news topic in the recent past is quantified by the length of the chain. The score contribution of a link between a current day cluster c and the previous day cluster with maximum similarity c^{t-1} in the chain is computed using a linear combination of $psim(c)$ and $S_{eff}(c)$. To account for more importance of recent days, we also penalize the score contributions of the days in the past by an exponential decay function based on the number of days between them. The overall historical value $\mathcal{H}_c(c)$ for a chain initiated from c is:

$$\mathcal{H}_c(c) = \sum_{i \leq t} (\beta.S_{eff}(c^i) + (1-\beta).e^{\alpha.(i-t)} psim(c^i)) \quad (3)$$

Where, the parameter α is the weight for the decay function and β is a tuning parameter to control the importance of cluster chain strength to the matching previous day cluster size. We use both the $psim(c)$ and $\mathcal{H}_c(c)$ as features in our experiments.

6.2 Temporal Profile from Named Events

For long-running topics like "Syrian Civil War" on international repute, there are already existing external sources which could be leveraged to account for historical significance. Moreover, the historical importance from cluster chaining ($\mathcal{H}_c$) does not provide the complete historical context for such long-running topics. Therefore, we assign canonical topic names from Wikipedia to the news events mined using the clustering technique described in Section 4. We coin the process of assigning a news event (or a representative story) to long-running Wikipedia topics as *Canonicalization* of news events.

For canonicalization we rely on the rich collection of manually curated and moderated annotations of news events in WCEP. A news story is assigned to a Wikipedia topic page with maximum similarity score based on KL-Divergence of the language models, Jaccard similarity of the entities and temporal profile of the news topic on the day news event was published. We combine all these three scores with feature weights estimated using an radial basis function kernel SVM classifer with degree $d = 3$ and $\epsilon = 0.01$.

One of the challenges of using Wikipedia topics is that long-running topics have several sections and sub sections. Each section or even paragraphs within the sections, often refer to different sub events. For example, within Wikipedia the page for the topic "Syrian Civil War" there are several sub topics such as "Battles of Damascus and Aleppo", "ISIL Offensive and U.S. air-strikes" and "Russian intervention" etc. In such situations deriving features from the entire Wikipedia articles may mislead the classifier. To address this problem, we derive features in a sub-event context s for a topic σ using a moving window denoted as σ_s.

We then define the *Moving Window Language Model* as the negative KL-Divergence score between the statistical unigram language model of the news story and Wikipedia page segment:

$$KL(LM_d || LM_{\sigma_s}) = \sum_w P(w|LM_d) log \frac{P(w|LM_d)}{P(w|LM_{\sigma_s})} \quad (4)$$

Where, $P(w|LM_d)$ and $P(w|LM_{\sigma_s})$ are probabilities of generating a word w from Wikipedia. This is estimated using term frequencies of the words contained in the news story and Wikipedia topic page segments. Further, using the entire Wikipedia corpus, we apply Dirichlet smoothing with smoothing parameter $\mu = 1000$. We also compute the *Moving Window Entity Overlap* using the disambiguated entities from the news topics and Wikipedia page segments as:

$$J(d, \sigma_s) = \frac{|\mathcal{E}(d) \cap \mathcal{E}(\sigma_s)|}{|\mathcal{E}(d) \cup \mathcal{E}(\sigma_s)|} \quad (5)$$

Finally, for long-running topics we obtain the historical importance by computing the *Temporal Prior* of a news topic on a given day. The number of edits received by a topic page

in Wikipedia reflects the evolution of that topic in the real world, we exploit this information as a temporal prior $T(\sigma, t)$ for a topic σ for a given day t and is defined in Equation (6).

In order to derive temporal profiles for named events (topics), we first catalog a comprehensive set of long-running topics and construct a *topic profile* based on their *Wikipedia edit histories*. This is quantified using the temporal prior defined below:

$$T(\sigma, t) = \sum_{t=t_1}^{t_2} \frac{|Edit(\sigma, t)|}{\sum_{t'=t_b(\sigma)}^{t_e(\sigma)} |Edit(\sigma, t')|} \quad (6)$$

Where, $Edit(\sigma, t)$ is the frequency of edits that the Wikipedia page of the topic σ received on a given day t. Since there could be lag between topic updates in real world and Wikipedia, we consider temporal prior for a topic over a range of days t_1, t_2 around the day of interest t. Finally, we compute historical significance $\mathcal{H}_w(c)$ on a day t as below.

$$\mathcal{H}_w(c) = (\beta.S_{eff}(c)) + (1-\beta).\sum_{x=1}^{W} (e^{\alpha.(-x)} T(\sigma, t-x)) \quad (7)$$

Where, W is a window of time period for which temporal profile is considered. Since we use exponential decay, we observed through experiments that in all cases the temporal profile beyond 30 days becomes negligible (psim < 0.0001). Like in cluster chaining we also associate a decay function using α and β to model the decreasing significance of an event. Note that to be able to compute $\mathcal{H}_w$ for any cluster, a canonicalized topic from Wikipedia must be assigned first.

6.2.1 Effectiveness of Event Canonicalization

For measuring the effectiveness of canonicalization we first compiled a list of long running stories from Wikipedia. This was done by first taking a seed set of stories from the WCEP and discarding Wikipedia pages for entities. Additionally, each of these topic pages link to multiple story pages which either represent (1) important sub events of the topic or (2) related topic pages. Next, we crawled all the news stories cited in each of these topic pages based on the assumption that the news stories referenced in the topic pages are most relevant to the topic. Additionally we extracted all news stories referenced along with a topic from WCEP. We only consider news topics with at least 10 news stories to ensure that there is sufficient training data. Finally, we had a set of 1900 long-running stories with 35000 topic-news article pairs. With this we observed that moving-window Language Model to select the best canonicalized topic achieves precision of 0.68, while using more features such as moving-window temporal prior and entity overlap results provides precision of 0.81.

7. EXPERIMENTAL EVALUATION

In this section we first present the datasets we consider for our experiments and explain how we create a groundtruth on which we evaluate our approaches. We then describe different evaluation scenarios, employed baselines and evaluation measures. Then in Section 7.5 we evaluate the effectiveness of our event models by comparing different ranking methods. Note that even though our evaluations focus on quality of ranking, we also extensively evaluated the quality of our mined events and their labels using a manual study. Due to the lack of space we omit those results.

7.1 Datasets

We consider two large datasets – GDELT and STICS:

1. The **Gdelt** dataset is derived from the *GDELT Project* which monitors and crawls a large fraction of the world's news media in over 100 languages. For our experiments we consider a dataset of only English stories amounting to around 8 million stories collected from September-2013 to August-2014 (365 days), with a mean daily batch of 22K stories. The number of news sources covered are over 6000 from a total of 167 different countries. GDELT news due to its large scale was annotated using a fast entity disambiguation tool – TAGME [7].

2. The **Stics** dataset is derived from the STICS semantic search system [15] which crawls RSS feeds of news outlets from 300 sources from a total of 10 different countries. From STICS we have a total of 1.69 Million news stories collected from January-2014 to June-2015 (545 days). STICS news corpus was readily annotated using AIDA entity disambiguation tool [16].

7.2 News Ranking Benchmark

The WCEP is a natural choice for evaluating daily summaries as it is manually curated by Wikipedia editors and quality is ensured by the moderators. We believe that WCEP provides a least biased coverage of international news when compared to any single news source which could have regional and topical biases. For instance, between 2010 and 2014 daily news events in WCEP contained 24.5% of the events about Wars and Conflicts, 15.5% about Politics, 13.5% about Law and Crime, 12.9% about Disasters and Accidents, 8.7% about International Relations, 7.1% about Sports , 7% about Business and Finance, 5% about Entertainment, 3% about Technology and 1.9% about Other categories.

7.2.1 Groundtruth for News Ranking

Our ranked list consists of news stories tagged with news topics from Wikipedia whenever possible. For accurate comparison with daily summaries of WCEP, we add the news stories referred in the WCEP summaries into the input collection for the respective days, we call these stories as *Ground Truth Stories* (GTS). Note that GTS are treated like any other news stories in the input collection and they are added before executing any preprocessing or event mining phase. Since we rank news events represented by the corresponding coherent cluster of news stories, we consider a news event to be relevant if it contains one of the GTS in its cluster. If a selected event is not canonicalized to a WCEP topic page, the cluster centroid (as defined in Section 4) is chosen as a representative story. We verified that centroids serve as high quality representative stories by conducting a survey but the results are again omitted due to lack of space. Since from 2013 onwards, each daily summaries, almost always, is qualified with a news reference, we consider the period September 2013-June 2015 for evaluating the daily summaries. After discarding the days for which more than half the WCEP URLs were unreachable we arrived at a groundtruth size of 617 days. It should be noted that the number of GTS per day is much smaller than the daily input batch (on an average 7 GTS per day vs 22k input stories) from the datasets and hence does not influence neither the clustering nor the

event ranking. The groundtruth and the results of our approach is publicly available [3].

7.2.2 Addressing Time Lag in WCEP

We noticed that WCEP was both lagging and leading publication dates of the corresponding cited external news stories links. From our analysis we discovered that around 20% of news events among the 40K events we analyzed in WCEP had their reported date mismatching with their corresponding external links. Interestingly, they were concentrated within the 3 days window of the WCEP dates. We conjecture that this is a consequence of a combination of reasons: incorrect publication times of the news stories, delay in reporting in WCEP and time zone differences. We also noticed in some cases WCEP reported the news events before the publication date of the news stories. This is because in some news websites the publication date is the same as last updated date. To counter the effect of lag we introduce a fixed time window of up to three days for each GTS within which it was actually published. Since, it is possible that due to spreading of GTSs on multiple days, that a GTS might be accounted for more than one day in our evaluation. We avoid such artifacts by strictly considering a GTS occurrence only once for computing the evaluation measures.

7.3 Baselines and Parameter Settings

In our knowledge, this is the first work on ranking news events on a daily basis and we do not know of any competitors which are directly applicable. However, we develop two baselines ranking using cluster size alone and language models proposed in [22]. We compare them with two variants of our solution with and without historical features for ablation.

Cluster-Size: Since the most obvious way to rank news events is based on their popularity, which is estimated from the corresponding cluster sizes, we consider a naive baseline which ranks events purely based on their cluster sizes.

***Story-Rank-LM*:** Next we consider a more sophisticated baseline in which the news stories are ranked on a daily basis based on their perceived popularity in the Blogosphere [22] using a language modeling (LM) approach. They first cluster blogs and compute the importance of a news article based on LM similarity between the news stories and the cluster. Further, they also consider LM-based temporal profiles constructed from temporal vicinity of the article publication date. They finally blend both the cluster-based importance and temporal profile contribution to obtain the final importance of the article on which it is ranked. We adapt their approach, for our task in the following manner. First, since our setup does not consider external sources like blogs, we use our own news collections as a proxy for the Blogosphere. Second, since they rank stories we use our daily clusters or events as a proxy to produce an event ranking. We consider two variants of *Story-Rank-LM* baseline– (1) *Story-Rank-LM* with only Language Model score from the current day and (2) *Story-Rank-LM-Hist* which takes the historical information from the adjacent weeks into account. Note that this approach needs a few parameters which we estimated using our training data. Moreover, we confirmed that these parameters are tuned to maximize their performance while conducting our experiments.

***Event-Rank-Pop*:** Furthermore, to demonstrate the effectiveness of our event-based popularity features, we also introduce a variant of our approach with improved popularity estimation based on event cluster coherence, effective cluster size (S_{eff}), source authority and diversity defined in Section 5 but omitting the historical features.

***Event-Rank-Pop-Hist*:** This is our method with all features including the historical features detailed in Section 6. For historical scores we have three parameters: α for tuning the strength of temporal decay, β for tuning the strength and θ as a threshold for the continuation of the cluster chains (cf. Section 6.1). Using the popular L-BFGS method [4], we estimated these parameters and set their values as: $\alpha = 0.5$, $\beta = 0.8$ and $\theta = 0.5$.

7.4 Evaluation Scenarios and Measures

For evaluating our daily rankings we consider five splits each with the same number of days – split size of 108 consecutive days (Stics) and split size of 73 consecutive days (Gdelt). Using these splits we conduct two types of experiments:

Retrospective: In our retrospective setup we conduct five-fold cross validation, three splits are used for learning, one for validation and one for testing. In sum, we evaluate $108 \times 5 = 540$ daily rankings for Stics and $73 \times 5 = 365$ days for Gdelt.

Online: In our online setting, we prohibit training data from the future. Consequently, we conduct experiments on the most recent split which covers Feb'15 to June'15 (Stics) and Jul'14 to Aug'14 (Gdelt) with all approaches being trained on the previous splits. Another reason for choosing this split, apart from its recency, is due the fact that this is a split with maximum input size and GTS.

Since the groundtruth we use from WCEP only contains a list of unordered relevant news events for each day, it limits our evaluation to precision-based measures and other ranking quality measures like nDCG are not relevant in this scenario. Hence, we consider the standard IR measures of Precision and MAP at different k values to measure the effectiveness of our daily event rankings.

7.5 News Ranking Results

Now we present the results of our news ranking evaluations by looking at the overall results in each of the settings and contrast them with our baselines.

In Table 1 (Restrospective) we measure the performance of two variants of our approach with the two baselines. For this experiment we consider the *retrospective* setting, hence the results are reported for daily rankings spanning 545 days (Stics) and 365 days (Gdelt). We first observe that our approaches outperform the baselines in most cases ($k > 1$) even with *Event-Rank-Pop* which represents our approach with improved popularity estimation including authority and diversity features. First we observe that the naive baseline using only the cluster size for ranking is inadequate, our features which incorporate sub-cluster cohesion like effective size, density and average pairwise distance improves performance. Due to the improved popularity estimates *Event-Rank-Pop* has a positive effect on the MAP@10 measure

[3] http://vinaysetty.net/data/wsdm2017

Dataset and Approach	Retrospective					Online				
	P@1	**P@5**	**P@10**	**MAP@5**	**MAP@10**	**P@1**	**P@5**	**P@10**	**MAP@5**	**MAP@10**
Stics										
Cluster-Size	0.665	0.491	0.427	0.560	0.507	0.711	0.555	0.508	0.625	0.576
Story-Rank-LM	0.302	0.308	0.294	0.307	0.302	0.376	0.409	0.421	0.396	0.404
Story-Rank-LM-Hist	0.358	0.349	0.350	0.355	0.353	0.462	0.491	0.507	0.492	0.499
Event-Rank-Pop	0.650	0.537	0.453	0.589	0.534	0.650	0.615	0.582	0.627	0.608
Event-Rank-Pop-Hist	**0.816**	**0.765**$^{\blacktriangle}$	**0.712**$^{\blacktriangle}$	**0.787**$^{\blacktriangle}$	**0.760**$^{\blacktriangle}$	**0.929**	**0.882**	**0.882**	**0.889**	**0.885**
Gdelt										
Cluster-Size	0.435	0.366	0.318	0.391	0.363	0.773	0.571	0.487	0.632	0.575
Story-Rank-LM	0.528	0.392	0.344	0.450	0.405	0.570	0.451	0.393	0.517	0.465
Story-Rank-LM-Hist	0.519	0.405	0.367	0.446	0.414	0.612	0.469	0.405	0.530	0.480
Event-Rank-Pop	0.475	0.380	0.339	0.420	0.388	0.797	0.695	0.578	0.750	0.690
Event-Rank-Pop-Hist$^{\bullet}$	**0.680**	**0.589**	**0.540**	**0.631**	**0.594**	**0.923**	**0.740**	**0.675**	**0.812**	**0.750**

Table 1: Comparison of Overall Effectiveness with baselines. *Event-Rank-Pop-Hist* Refers to our approach with all features. The superscript denotes a statistically significant difference (using Student's t-test) when compared to the closest competitor ($p \leq 0.05$). For example, ▲ represents statistically significant to *Event-Rank-Pop* for Stics and • represents statistically significant to the closest competitor *Story-Rank-LM-Hist* for Gdelt.

which increaes by 5% (from 0.507 to 0.534) for Stics and 6% increase (from 0.363 to 0.388) Gdelt data respectively.

The *Story-Rank-LM* models also suffer because the ranking is done using only text-based features and since the popularity is estimated via Language Model from the corpus-wide clusters. This is evident especially with results using Stics data, which shows that using *Event-Rank-Pop* MAP@10 increases by 76% from 0.302 to 0.534. For Gdelt on the other hand *Story-Rank-LM* performs slightly better because in Gdelt sometimes the authority and diversity features could mislead the popularity. After further careful investigation we observed that *Event-Rank-Pop* performs better when the number of GTS per day is lower. Specifically, on days with GTS per day ≤ 8, the average MAP@10 for *Event-Rank-Pop* is 0.37, while it is 0.26 for *Story-Rank-LM*.

Another interesting observation is that in *Story-Rank-LM-Hist*model there seems to be a clear improvement in performance over its counterpart without historical featuresand this is the first indication that historical information is crucial in ranking events. However, *Story-Rank-LM-Hist* method even with historical features still suffers because the time window considered for computing temporal profile is restricted. For example, *Story-Rank-LM-Hist* model does not give high score to an important news event related to Malaysia Airlines Flight 370 on 27th of May, 2014, because the time window considered for computing the temporal profile is restricted and excludes the actual date of the flight crash on 8th of March 2014. In such scenarios mapping the news events to a canonical topic as described in Section 6.2 helps in assessing the true historical significance of the news event. Hence, the superior performance of *Event-Rank-Pop-Hist* over *Story-Rank-LM* models.

It is worth noting that most significant improvement in performance is seen for *Event-Rank-Pop-Hist* which is *Event-Rank-Pop* extended with the historical features. From Table 1 it is evident that the cluster chaining and temporal profile from canonicalized WCEP topic which when added to *Event-Rank-Pop* improves the Prec.@10 by 57% for Stics and 59% for Gdelt. We further investigated the components of the historical features namely canonicalization, chaining and most recent history and found that the chaining provided the maximum improvement as compared to the others. Although the most recent history, i.e. similarity with the best matched cluster from the day before, positively contributes towards the effect of history it is limited to events which pan out for a longer interval. That is, it is a good indicator for an onset of an event. Canonicalization-based similarity is helpful only for a smaller proportion of events which have a Wikipedia events page. Chaining clusters on the other hand is flexible to follow the importance trail of highly related clusters on adjacent days to overcome both these problems.

Next we compare the performance of our approaches in the online setting described in Section 7.4. From Table 1 (online) we see a similar trend as in Table 1 with *Event-Rank-Pop-Hist* consistently outperforming other approaches along with having high MAP and Precision values. Most notable difference is that our approaches perform better for online tasks. This is because training on all splits except the last split significantly enriches the historical features and in retrospective version, training on future data is not leveraged for deriving historical or in this case future significance.

8. CONCLUSION

In this paper we introduced the problem of ranking a daily batch of events for large heterogeneous news corpora. We identified several limitations of the state-of-the-art event ranking techniques. Consequently, we presented our solution which clusters daily batch of news stories to derive events. With the use of improved popularity and historical features for events in a learning to rank framework we came up with an effective daily event ranking. We observed in our results that our rich feature sets are able to capture the typicalities of the two different datasets we considered with the historical features giving us the maximum benefit. Our solution is robust and does equally well in retrospective as well as in the online settings. Additionally, our canonicalization mechanism which takes into account temporal priors performs superior to a language-model based baseline.

Acknowledgements

This work supported in part by the ERC Advanced Grant ALEXANDRIA (grant no. 339233).

9. REFERENCES

[1] J. Allan. *Topic Detection and Tracking: Event-Based Information Organization*. Springer, Feb. 2002.

[2] J. Aslam, F. Diaz, M. Ekstrand-Abueg, R. McCreadie, V. Pavlu, and T. Sakai. Trec 2014 temporal summarization track overview. Technical report, DTIC Document, 2015.

[3] G. Binh Tran. Structured summarization for news events. In *WWW*, pages 343–348, 2013.

[4] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.

[5] W. Dakka and L. Gravano. Efficient summarization-aware search for online news articles. In *JCDL*, pages 63–72. ACM, 2007.

[6] G. M. Del Corso, A. Gullí, and F. Romani. Ranking a stream of news. In *WWW*, pages 97–106. ACM, 2005.

[7] P. Ferragina and U. Scaiella. TAGME: On-the-fly annotation of short text fragments (by wikipedia entities). In *CIKM*, pages 1625–1628. ACM, 2010.

[8] B. Fetahu, K. Markert, and A. Anand. Automated news suggestions for populating wikipedia entity pages. In *CIKM*, pages 323–332. ACM, 2015.

[9] B. Fetahu, K. Markert, W. Nejdl, and A. Anand. Finding news citations for wikipedia. In *CIKM*, pages 337–346, New York, NY, USA, 2016. ACM.

[10] M. Gallé and J.-M. Renders. Full and mini-batch clustering of news articles with star-em. In *Advances in Information Retrieval*, pages 494–498. Springer, 2012.

[11] W. Gao, P. Li, and K. Darwish. Joint topic modeling for event summarization across news and social media streams. In *CIKM*, pages 1173–1182. ACM, 2012.

[12] A. Gionis, P. Indyk, R. Motwani, et al. Similarity search in high dimensions via hashing. In *VLDB*, volume 99, pages 518–529, 1999.

[13] R. Gwadera and F. Crestani. Mining and ranking streams of news stories using cross-stream sequential patterns. In *CIKM*, pages 1709–1712. ACM, 2009.

[14] R. Gwadera and F. Crestani. Mining news streams using cross-stream sequential patterns. In *RIAO*, pages 106–113, 2010.

[15] J. Hoffart, D. Milchevski, and G. Weikum. Stics: searching with strings, things, and cats. In *SIGIR*, pages 1247–1248. ACM, 2014.

[16] J. Hoffart, M. A. Yosef, I. Bordino, H. Fürstenau, M. Pinkal, M. Spaniol, B. Taneva, S. Thater, and G. Weikum. Robust disambiguation of named entities in text. In *EMNLP*, pages 782–792. ACL, 2011.

[17] T. Joachims. Training linear svms in linear time. In *SIGKDD*, pages 217–226. ACM, 2006.

[18] M. Kabadjov, M. Atkinson, J. Steinberger, R. Steinberger, and E. Van Der Goot. Newsgist: a multilingual statistical news summarizer. In *ECML PKDD*, pages 591–594. Springer, 2010.

[19] L. Kong, S. Jiang, R. Yan, S. Xu, and Y. Zhang. Ranking news events by influence decay and information fusion for media and users. In *CIKM*, pages 1849–1853. ACM, 2012.

[20] E. Kuzey, V. Setty, J. Strötgen, and G. Weikum. As time goes by: Comprehensive tagging of textual phrases with temporal scopes. In *WWW*, pages 915–925, 2016.

[21] E. Kuzey, J. Vreeken, and G. Weikum. A fresh look on knowledge bases: Distilling named events from news. In *CIKM*, pages 1689–1698. ACM, 2014.

[22] Y. Lee and J.-H. Lee. Identifying top news stories based on their popularity in the blogosphere. *Information Retrieval*, 17(4):326–350, May 2014.

[23] L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *WWW*, pages 661–670. ACM, 2010.

[24] L. Li, D. Wang, T. Li, D. Knox, and B. Padmanabhan. SCENE: a scalable two-stage personalized news recommendation system. In *SIGIR*, pages 125–134. ACM, 2011.

[25] T.-Y. Liu. Learning to rank for information retrieval. *Found. Trends Inf. Retr.*, 3(3):225–331, Mar. 2009.

[26] R. McCreadie, C. Macdonald, and I. Ounis. News vertical search: when and what to display to users. In *SIGIR*, pages 253–262. ACM, 2013.

[27] Q. Mei and C. Zhai. Discovering evolutionary theme patterns from text: An exploration of temporal text mining. In *SIGKDD*, pages 198–207. ACM, 2005.

[28] A. Mishra and K. Berberich. Leveraging semantic annotations to link wikipedia and news archives. In *ECIR*, pages 30–42. Springer International Publishing, 2016.

[29] R. Nallapati, A. Feng, F. Peng, and J. Allan. Event threading within news topics. In *CIKM*, pages 446–453. ACM, 2004.

[30] I. Ounis, C. Macdonald, and I. Soboroff. Overview of the TREC-2008 blog track. 2008.

[31] T. Pang-Ning, M. Steinbach, V. Kumar, et al. Introduction to data mining. In *Library of Congress*, page 74, 2006.

[32] G. Raveendran and C. L. Clarke. Lightweight contrastive summarization for news comment mining. In *SIGIR*, pages 1103–1104. ACM, 2012.

[33] V. Setty, S. Bedathur, K. Berberich, and G. Weikum. Inzeit: efficiently identifying insightful time points. *Proceedings of the VLDB Endowment*, 3(1-2):1605–1608, 2010.

[34] D. Shahaf and C. Guestrin. Connecting the dots between news articles. In *SIGKDD*, pages 623–632. ACM, 2010.

[35] D. Shahaf, J. Yang, C. Suen, J. Jacobs, H. Wang, and J. Leskovec. Information cartography: creating zoomable, large-scale maps of information. In *SIGKDD*, pages 1097–1105. ACM, 2013.

[36] J. Singh, W. Nejdl, and A. Anand. History by diversity: Helping historians search news archives. In *CHIIR*, pages 183–192. ACM, 2016.

[37] S. Vadrevu, C. H. Teo, S. Rajan, K. Punera, B. Dom, A. J. Smola, Y. Chang, and Z. Zheng. Scalable clustering of news search results. In *WSDM*, pages 675–684. ACM, 2011.

[38] C. Wang, M. Zhang, L. Ru, and S. Ma. Automatic online news topic ranking using media focus and user attention based on aging theory. In *CIKM*, pages 1033–1042. ACM, 2008.