Scholarly Engagement of Web Archives and Historical Search

Masterarbeit
zur Erlangung des akademischen Grades
M.Sc. Internet Technologies and Information Sytems

Jaspreet Singh

Leibniz Universität Hannover
Fakultät für Elektrotechnik und Informatik
Institut für verteilte Systeme
Fachgebiet Wissensbasierte Systeme
Forschungszentrum L3S

Hannover
13.11.2014

Examiner I	Prof. Dr. Wolfgang Nejdl
Examiner II	Dr. Avishek Anand

Abstract

In this thesis, we propose a approach of identifying rumors in Twitter.

Titter is a mircoblogiging service that which are used by millions users. Users can publish and exchange information with short tweets whatever when and where. This makes it a ideal media for spreading breaking news and false rumors.

So automatic detecting rumors on social media has become a trending topic. But early researches mostly focused on rumors during one or several (??) events like earthquake or terrorist attack. But in our work, we more focus on general rumors.

And most of previous work for rumor detection focused on static features like the content of tweets or propagation features, and they ignored that those features change during the information's propagation over time.

we use Dynamic Series-Time Structure (DSTS)(wenxian) to capture the temporal features. And we add Spike Model, SIS Model and SEIZ Model as time series features. To improve the time series model's performance at early stage of the event we develop a single tweet's credibility scoring model which only using features which can be extracted from single tweet on the Twitter interface.

Our experiments using the events from Twitter and our model demonstrates better performance on detecting rumors.

Contents

1 Introduction

Twitter is a mircoblogging service which are used by millions users. Users can publish and exchange information with short tweets within 140 characters. It is cheap and can be accessed though several like website, email or mobile phone. That makes it a ideal media for spreading breaking news and false rumors. A study by the Pew Research Center showed that the people in USA under age of 30 consider Internet as the major resource of news and Internet became the second important media overall(wenxian).

But these advantages make Twitter which became one of the most importance resources of breaking news, at the same time becomes into a ideal media for spreading unverified information. On Twitter everyone can be a journalist and publish news or rumors without any substantiation which must be done by traditional journalists before news' publishing.

Rumor could be defined as a statement whose truth value is unverified or deliberately false(wenxian).And they could be harmful to the government, market and society. One case is some hacked accounts spread a rumor about Obama had been injured in white house. The S&P crashed and wiped off 130 Billion dollars of stock value (wenxian).

So a method of detecting rumors on Twitter that could detect rumors in the early stage of propagation can be very usefull.

1.0.1 Contributions

This thesis develops a model of classification of single tweet with high credibility or low credibility and a time series model for detection of rumor event.

The classifier of single tweet is random decision forest with accuracy 81%.(xiugai) And we use the result of this classifier as a feature of s In this thesis, we make the following contributions:

- We develop a model of classification of single tweet with high credibility or low credibility. We call it credit scoring model. Considering it could be set up online in the further and it should response as quick as possible. So we use only the features which can be extracted from one tweet in the Twitter's interface. The

classifier uses random decision forest. Although the features are limited, it still gets 81% accuracy(xiugai). The result of this model we called it credit score.

- We develop a time series model for detecting rumor events. We used Dynamic Series-Time Structure (DSTS)(wenxian) to capture the changes of features over time. And we tested 3 time series model as feature: modified Spike Model(wenxian), SIS model(wenxian) and SEIZ model(wenxian). We fit the results of credit scoring model as a feature into time series model to improve the performance in the early stage. And we approved that credit score is one of the best feature.
- We study how top features changes over time during the spreading of rumors.

1.0.2 Thesis Outline

(xiugai) The rest of this these is organized as follows: In Chapter **??** we try to understand the research practices of humanities scholars when working with web archives. We also seek to find problems faced by the scholars when accessing web archives using keyword search. Based on our findings we introduce Historical Query Intents(HQIs) in Chapter 3. In the same chapter we also motivate and develop a novel retrieval model suited for HQIs. The experiments to test the performance of this model against its competitors is covered in the same chapter. The penultimate Chapter 4 details the architecture of the system that was built to demonstrate the efficacy of the retrieval models to users. Finally, in Chapter 5 we add some concluding remarks and describe future work.

2 Background and Related Work

2.1 Twitter

Twitter is a mircoblogging service. Now there are more than 140 million active users[1]. User can publish short messages within 140 characters aka tweets.

2.1.1 Retweet

A retweet is re-posting of a tweet by other users. One way of retweet is using RT at the beginning of the tweet which is retweeted. The other way is using "retweet button" which is officially launched by Twitter after 2015. The difference between these two retweet is tweets are retweeted by "retweet button" can't be searched by Twitter's searching interface, but manually retweets with RT keyword can. So in our work retweet means the tweets are retweeted manually. The number of how many times of this tweet has been retweeted is showed behind it.

2.1.2 Mentions

Mentions are in form like "@username" which are added in the text of tweet. The users are mentioned will receive the notification of this tweet on their homepage.

2.1.3 Hashtags

Mentions are in form like "#topic". It means this tweet belongs to some topics.

2.1.4 Favorite

Favorites means how many users like this tweet. It is showed on the interface of Twitter

[1] https://blog.Twitter.com/2012/Twitter-turns-six

2.1.5 Verified User

Verified User means this account of public interest is authentic by Twitter. It is showed by a blue icon behind the name of the poster.

2.1.6 Followers

The followers of a user are accounts who receive this user's posting. The total number of followers can been seen in the profile of poster.

2.1.7 Following

The following are other accounts who follow this user. The total number of following can been seen in the profile of poster.

2.1.8 Twitter API

Twitter API is provided by Twitter[2] for developer. But the search API only return a sampling of recent Tweets published in the past 7 days[3]. We need the full stories of the events, so we crawled the data directly from the searching interface[4].

2.2 Definition of Rumor

The definition of rumor in our work is unverified information spreading on Twitter over time. It is a set of tweets including the the sources, retweets and debunking tweets.

2.2.1 Rumor Event

And if a rumor didn't widely spread and it could be harmless. So we more focused on the rumors which are widely spread and contain one or more bursty pikes during propagation. We call it "rumor event".

2.2.2 Time Period of an Event

The beginning of a rumor is hard to definition. And every formal work didn't mention how to define the beginning of one rumor event. One rumor may be created serval years ago, but it can be triggered by other events and quickly spread. For example, a rumor[5] claimed that Robert Byrd was member of KKK. This rumor has been circulating in Internet for a while, as shown in figure(wenxian) that every day there are several tweets. But this rumor was triggered by a picture between him and Hillary Clinton.We defined the hour which has the most tweet's volume as t_{max} and the first

[2] https://dev.twitter.com/overview/api
[3] https://dev.twitter.com/rest/public/search
[4] https://twitter.com/search-home
[5] http://www.snopes.com/robert-byrd-kkk-photo/

tweet before t_max within 48 hours we defined it as the beginning of this rumor event t_b. And the end time of events we defined as $t_{end} = t_b + 48$.

2.3 Machine learning

2.3.1 Machine learning Overview

Machine learning covers vast numbers of algorithms and has been successful applied in different field. The challenges of ML are finding the best model which is suitable for this task, fitting the parameters and selecting the features.

Normally we split the ML methods into Supervised learning, Unsupervised learning and Reinforcement learning[24].

The supervised learning is the most popular method of ML. It needs a set of inputs and a set of desired outputs. And the algorithm will learn to produce the correct output based on the new input.

The unsupervised learning needs a set of inputs but no outputs. The algorithm will generate the outputs like clusters or patterns. The unsupervised learning task can be used for example when people can't label the outputs.

2.3.2 random forest (RF)

Classification is a supervised data mining technique. Our work can be considered as a classification task. And the classification model is random forest.

RF is an algorithm of supervised learning which developed by Leo Breiman[5]. It's built by a set of classification trees[6]. Each tree is trained by a small bootstrap sample of training set and while prediction each tree votes one single candidate. By taking the majority vote RF can produce the result of prediction.

Because RF uses a random subset of features instead of the best features in every node, so it can avoid the overfitting[5].

Another benefit of RF is that it can return the features' importance. RF is built up by a subset training data and the data we didn't selected we call it out-of-bag (OOB) data and we can validate the model by using OOB data we got OBB error $E_{oob}(G) = \frac{1}{N}\sum_{n=1}^{N} err(y_n, G_n^-(X_n))$ with X_n are features only in OOB. At last we get the importance of feature by permuting one feature to random numbers. $importance(i) = E_{oob}(G) - E_{oob}^p(G)$ where $E_{oob}^p(G)$ is the OBB error after permuting a feature. We use this method to rank the fearers in section(wenxian).

2.4 Machine learning

3 Historical search

3.1 Introduction

From our study of scholarly access of web archives, we found that keyword search is the primary access method. Our findings also show that ranking search results is a problem faced by scholars at the British Library. Users are interested in a getting an overview of search results not only across aspects of the query but also across time. Keyword queries with an inherent historical intent over longitudinal text corpora like web archives are interesting for the scholars like historians, social scientists and journalists. While searching documents published over time, a key preference as we have seen is to retrieve documents which are from the important aspects from important points in time. To this extent, we introduce the notion of a *Historical Query Intent* and define an aspect-time diversification problem over archives.

In particular, we chose news archive since we have reliable publication dates and a wider time span when compared to some of the standard web crawls available for experimentation. Also since we do not address the problem of multiple versions of the same page, news archives are the ideal sample to test our work.

Consider **Case 3** from the previous section. Lets say the scholar interested in the history of Rudolph Giuliani, the ex-mayor of New York City in the time interval between 1987 to 2007. She is not only interested in the important facets like `mayoral campaigns, his mayorality, race for senate` and `his efforts during 9/11`, but she is interested in articles which cover these aspects when they were important. News articles of historical interest can be classified into breaking news, opinion pages, or summary and reflective pieces. Although reflective or summary articles might mention important aspects, events and news they might always belong to the time of interest of the aspect in question. Thus, presenting a news article about Giuliani's efforts for 9/11 during 2001-2002 is deemed more interesting than articles from other time periods. Similarly, articles talking about the mayoral campaigns in 1989, 1993 and 1997 (the years when the mayoral elections were held) are more effective than documents covering this aspect in 2007. We introduce the notion of such historical query intents or *HQI* which aims to diversify search results by explicitly taking into account the aspects to which the news articles belong alongwith the

time of importance of the aspect.

In this chapter, we present a novel algorithm, HistDiv, that explicitly models the aspects and important time windows to which the results of the query belong. We also propose a new metric Tia-Sbr to evaluate the effectiveness of methods intending to solve it. We test our methods by constructing a test collection based on *The New York Times Collection* with a workload of 30 queries assessed manually. Our experiments show that we outperform all the competitors in most of the measures, and remain competitive in a select few.

3.2 Historical Query Intent

Historical Query Intent is the moniker we choose to describe a user's intent to cover as many historically relevant subtopics and time windows for a given topic. I operate on a document collection $\mathcal{D}$, where each document d_t is associated with a timestamp t corresponding to its publication date. The entire span of the document collection is sub-divided into a set of non-overlapping time windows $\mathcal{T} = \{t_1, t_2, t_3 ... t_n\}$. I require that each document d_t has exactly one publication date t implying that it belongs to exactly one time window denoted by $w(d_t) \in \mathcal{T}$, i.e., $begin(w(d_t)) \leq t \leq end(w(d_t))$ where $begin(w(d_t))$ and $end(w(d_t))$ denote the begin and end time boundaries of $w(d_t)$. Additionally, each $d_t \in \mathcal{D}$ is labeled with a set of aspects which are used to describe the content of d_t. I let $\mathcal{A}$ be a universal set of all aspects such that $A(d) \subseteq \mathcal{A}$ is the set of aspects for the document d_t.

In order to satisfy the aforementioned historical query intents which requires documents from relevant aspects and relevant points in time, I propose a search result diversification problem called *Historical Search Result Diversification* task. In this task, given a set of retrieved results $R_q \subseteq \mathcal{D}$ for a query q, I intend to diversify aspects from $\mathcal{A}$ and time windows from $\mathcal{T}$ such that I get a set $S \subseteq R_q$ of k documents that cover the most important aspects and time windows for a given topic expressed by q. As with prior work on *explicit search result diversification* [3], we assume each topic q has a set of subtopics $c \in C(q)$, with a probability distribution $P(c|q)$ which S looks to satisfy. Generalizing the traditional search result diversification tasks to include time I seek to maximize an objective function $P(S|q)$ to find the best S over a set of subtopics $C(q)$.

Definition 3.1 *The Historical Search Result Diversification task intends to find a set S which maximizes $P(S|q)$ over a set of subtopics $C(q)$ as well as a set of time windows $T(q) = \bigcup_{d_t \in R_q} w(d_t)$.*

$$P(S|q) = \sum_{t \in T(q)} P(t|q) \sum_{c} P(c|q)(1 - \prod_{d \in S}(1 - V(d|q, c, t)) \tag{3.1}$$

$P(S|q)$ represents the probability that the user finds at least one relevant document from an important time window. $V(d|q, c, t)$ is the utility of a document d given subtopic c, query q and in the time window t. This utility decreases for other documents if we add a document to S which has a high probability of satisfying a user

interested in time t and subtopic c. By doing so we also expose ourselves to quirks of their interpretation of diversification, i.e, if there is a dominant time window or subtopic then more documents are added to S until they are sufficiently satisfied. The problem we have defined is akin to a 2-dimensional diversification problem. Being a generalization to earlier formulations, which were based on the classical Maximum k-Coverage Problem, the historical search diversification task is also $\mathcal{NP}-$hard.

3.3 Time-Aware Effectiveness Measures

Since we follow a diversification based approach to historical search we adapt the existing diversity measures to take time into account. We first extend existing diversification metrics by making them two dimensional, and then we proceed to propose a new metric which takes into account the importance of the aspects along with time.

3.3.1 Time-augmented Diversification Metrics

Similar to aggregating metrics like Precision, NDCG, ERR and Subtopic Retrieval across intents in [26], we aggregate the contribution of each of these metrics to every time window. Let C is the set of subtopics for a topic q in the workload Q and $rel(j|c)$ is the relevance of a document at j to a subtopic c belonging to the time window t. In this subsection we show how th traditional intent aware measures can be augmented to include time.

Time aware alpha-IA-NDCG NDCG (Normalized Discounted Cumulative Gain) is a measure that rewards result lists of length k that rank relevant documents closer to the top of the list. The core of the measure is cumulative gain which is then discounted based on the position of the documents in the ranked result list. Intent awareness is added to the measure by computing a weighted sum of NDCG for each subtopic c such that $\sum_c P(c|q) = 1$. We introduce time awareness in a similar fashion. The importance of a time period or window is represented by conditional probability of a time window over the query. We compute intent aware NDCG over each time window within the time range of the collection and compute the weighted sum over all time windows where $\sum_t P(t|q) = 1$. Time aware Intent aware α-NDCG rewards result lists of length k that rank, for each intent, relevant documents from important time windows closer to the top of the list.

$$P(t|q) = \frac{|d_{t,q}|}{|d_q|} \tag{3.2}$$

where $d_{t,q}$ is a document with time stamp t and relevant to the query q.

$$DCG = \sum_{j=1}^{k} \frac{2^{rel(j)} - 1}{\log(1+j)} \tag{3.3}$$

where j is the rank of the document and $rel(j)$ is the binary relevance judgement of the document at j. For intent awareness, DCG is modified by changing $rel(j)$ to $rel(j|c)$. NDCG is computed as $\frac{DCG_k}{IdealDCG_k}$.

$$\text{ia-Ndcg(Q)}_k = \sum_c P(c|q)NDCG(Q, k|c) \tag{3.4}$$

$$\text{Tia-Ndcg(Q)}_k = \sum_t P(t|q) \sum_c P(c|q)NDCG(Q, k|c) \tag{3.5}$$

Time aware IA-ERR Intent aware estimated reciprocal rank (IA-ERR) has been used by TREC as its primary measure for measuring diversity performance. It is a cascade user model based metric and it is shown to be more accurate than position based metrics like NDCG. Temporal ERR-IA is computed by introducing time as an extra dimension of intent in the original computation of ERR-IA.

$$\text{ia-Err}_k = \sum_r^n \frac{1}{r} \sum_c P(c|q)rel(r|c) \prod_{i=1}^{r-1}(1 - rel(i|c)) \tag{3.6}$$

$$\text{Tia-Err}_k = \sum_r^n \frac{1}{r} \sum_t P(t|q) \sum_c P(c|q,t)rel(r|c) \prod_{i=1}^{r-1}(1 - rel(i|c)) \tag{3.7}$$

Time aware IA precision Time-aware intent-aware precision at k, Tia-Precision@k can be defined as

$$\text{ia-Precision}_k = \frac{1}{|C|} \sum_{c=1}^{|C|} \frac{1}{k} \sum_{j=1}^{k} rel(j|c) \tag{3.8}$$

where C is the topic represented as a set of subtopics c. $rel(j|c)$ is the relevance of a document at j to a subtopic c. This measure determines how precise a ranked list is over a set of intents. For historical search we need to measure precision not only over subtopics but also over time. We want a ranked list to consist of documents from diverse subtopics and diverse time windows. To measure the precision of covering time as well as subtopics we introduce time in IA-Precision in the following way:

$$\text{Tia-Precision}_k = \sum_{t \in T(q)} P(t|q) \underbrace{\frac{1}{|C|} \sum_{c=1}^{|C|} \frac{1}{k} \sum_{j=1}^{k} rel(j|c,t)}_{\text{Precision } \textit{in time wind.}\ t} \tag{3.9}$$

Time aware average IA-precision & MAP-IA Average IA-Precision is used to measure how precise a retrieval model is for $1 \leq k \geq n$. To compute Mean average precision, Average IA-Precision is computed for all queries in Q and then we compute

Measure	Formula
Tia-NDCG$_k$	$\sum_t P(t\|q) \sum_c P(c\|q) NDCG(q, k\|c, t)$
Tia-ERR$_k$	$\sum_r^k \frac{1}{r} \sum_t P(t\|q) \sum_c P(c\|q) rel(r\|c, t) \prod_{i=1}^{r-1} (1 - rel(i\|c, t))$
Tia-AvgPrecision$_k$	$\sum_{k=1}^{n} (\text{Tia-Precision}_k * rel(k))/\|R\|$
Tia-MAP$_k$	$\sum_Q \text{Tia-AvgPrecision}_k/\|Q\|$
T-Sbr$_k$	$\frac{\|subtopics(S)\| + \|timeWindows(S)\|}{\|subtopics(q)\| + \|timeWindows(q)\|}$

Table 3.1: Time-Aware Effectiveness Measures

the mean. We use Time Aware IA-Precision instead of the standard IA-Precision to introduce time awareness.

$$IA - AvgP(q)_k = \frac{\sum_{k=1}^{n} (2d - IA - Precision(k) * rel(k))}{\#relevant\ documents} \tag{3.10}$$

$$\text{Tia-MAP}_k = \frac{\sum_Q IA - AvgP(q)}{|Q|} \tag{3.11}$$

Subtopic recall Subtopic recall is the measure of intent coverage for a given result list at depth k.

$$Sbr_k = \frac{|subtopics(S)|}{|subtopics(q)|} \tag{3.12}$$

$$\text{T-Sbr}_k = \frac{|subtopics(S)| + |timeWindows(S)|}{|subtopics(q)| + |timeWindows(q)|} \tag{3.13}$$

T-Sbr$_k$, our adaptation of subtopic recall, considers both time windows and subtopics to be members of a single set of intents. $subtopics(S)$ and $timeWindows(S)$ denote set of subtopics and time windows covered by the diversified result set S respectively. Similarly, $subtopics(q)$ and $timeWindows(q)$ denote the set of all subtopics and time windows for a given query q.

3.3.2 Time-Aware Subtopic Recall

To accurately measure the historical value of a result set we need a metric that models the coverage of important time windows and subtopics. It is fair to consider the equal importance of each subtopic although doing the same for time would not provide an accurate way to discriminate. For example, it is safe to assume that a user searching for the history of the 9/11 attacks would prefer to get relevant documents from the year 2001 rather than 2007. Hence it desirable to reward result lists that rank relevant documents from important time periods. We introduce a variant of subtopic recall called time aware subtopic recall. This measure's discriminatory power comes from direct modeling of time windows as bursts rather than

Algorithm 1: Temporal IA select

Input: $k, q, A(q) \in l_R, R(q), A(d) \in l_R, P(a|q), V(d|q,c), S =$
Output: S
$\forall a, U(a|q,S) = P(c|q)$
while $|S| \leq k$ **do**
  for $d \in R$ **do**
    $g(d|q,a,S) \leftarrow \left(\sum_{c \text{ in } A(d)} U(a|q,S)V(d|q,a)\right)$
  $d^* \leftarrow \text{argmax}\, g(d|q,a,S)$
  $S \leftarrow S \cup \{d^*\}$
  $\forall a \in A(d^*)\,,\ U(a|q,S) = (1 - V(d^*|q,a))U(a|q,S/d^*)))$
return S

new subtopics like in our adaptation of subtopic recall. Each time window t is given a burst weight $P(t|q)$ just like each subtopic.

Definition 3.2 *The time-aware subtopic recall at k,* Tia-SBR$_k$*, is defined as*

$$\text{Tia-SBR}_k = \alpha \underbrace{\sum_{c}^{C \in S} P(c|q)}_{\text{Sbr}_k} + (1-\alpha) \underbrace{\sum_{t}^{t \in S} P(t/q)}_{\text{burst recall at k}} \tag{3.14}$$

where Sbr_k is a weighted interpretation of subtopic recall. Considering equal importance for all subtopics leads to $P(c|q) = \frac{1}{|C|}$ which is the standard subtopic recall used for diversity evaluation. Intent aware measures will just favor algorithms that can cover more subtopics. Adding time awareness to these measures will have favorable results for algorithms that select important time periods and subtopics. Hence by combining time awareness and intent awareness in the standard measures we are able to determine just how well an algorithm performs for a historical query intent. We also choose Tia-Sbr as the measure of choice for the historical diversification task.

3.4 The HistDiv Approach

In devising a retrieval model for our historical search let us look at how we can add time to the existing diversification models which explicitly model aspects. A naïve approach is to enrich the aspect space by adding time as aspects. Since we deal with two dimensions we can project or *linearize* the temporal dimension onto the aspect dimension. Say the result set R has a document d with m aspects $\{a_1, \ldots, a_m\}$ and belongs to the time window t_i, it contributes m linearized aspects denoted as $l_d = \{a_{1,i}, \ldots, a_{m,i}\}$. Thus the overall linearized aspect-time space l_R or a result set R can be represented as $\cup_{d \in R} l_d$. Methods like Ia-Select [3] and Pm2 [14] can then operate on this enriched aspect-time space.

We consider these time-enriched diversification models as baselines in our experiments denoted as Tia-Select and Tpm2. The algorithms are detailed in Algorithm 1

Algorithm 2: Temporal PM2

Input: $k, q, A(q) \in l_R, R(q), A(d) \in l_R, P(a|q), P(d|q,a), S = null$
Output: S
for $a_i \in A(S)$ **do**
  $quotient[i] = \frac{v_i}{2a_i+1}$
$i^* \leftarrow argmax\ quotient[i]$
$d^* \leftarrow argmax_{d_j \in R}\ \lambda * quotient[i] * P(d_j|q,a_{i^*}) + (1-\lambda) \sum_{i \neq i^*} quotient[i] * P(d_j|q,a_{i^*})$
$S \leftarrow S \cup \{d^*\}$
$R \leftarrow R/\{d^*\}$
for $a_i \in A(d^*)$ **do**
  $a_i = a_i + \frac{P(d^*|q,a_i)}{\sum_{a_j} P(d^*|q,a_j)}$
return S

and Algorithm 2 respectively. Both retrieval models offer different takes on diversity. Ia-Select is based on the assumption that if a user has seen a high quality document d with aspect a_i then there is no need to immediately serve the user with another document from a_i. Ia-Select uses a greedy approach to finding the set S which offers an approximation guarantee. Documents are sequentially added to the result set S based on the utility score g of the documents at that step. The utility of a document depends on the document utility V (given by the relevance score) and the aspect utility U (given by the probability of an aspect in the set R R). The aspect utility score depends on the aspects covered already by the set S at that point. If a document d with a high relevance score covers an aspect a which is already in S then the aspect utility of d is very low and causes the overall utility of the document g to be low.

Pm2 approaches the problem of diversity by proportionality. It assumes that the user's intent is to get a result set S that has the same proportion of aspects as R. It models diversity as a parliament seat allocation problem where aspects are parties which need to be assigned to a limited number of seats. Here each aspect is given votes based on the number of documents it covers in R. Based on these votes, aspects are assigned a portion of the seats in parliament which is the set S and the seats are placeholders for documents. They use the St. Lauge algorithm to solve the problem of proportionally assigning seats to aspects. Instead of computing utility scores for documents like Ia-Select, Pm2 computes the quotient of an aspect at every step and then selects a document from the aspect with the highest quotient score. This quotient depends on the proportion of aspects already covered by S at the given step as compared to the number of votes it has garnered. When a document d is added to S then the proportion of aspects covered in S is updated instead of discounting utility like Ia-Select.

A potential drawback with the linearized models is that newly formed aspects do not always ensure maximum coverage of the temporal and aspect space. An alternative would be to keep the dimensions separate akin to the multi-dimensional approach proposed in [17]. In this general framework for the diversification of n arbitrary dimensions, the utility score $g(d|q,c,S)$ computation reflects how the dimensions are combined. The marginal utility of aspects given a document d is computed

Algorithm 3: The HistDiv Algorithm

Input: $k, q, A(q), R(q), T(q), V(d|q,c), S = \emptyset$
Output: Set S of diversified documents

$\forall c \in A(q)$, $\forall t \in T(q)$, $U_aspect(c|q,S,t) = P(c|q,t)$
$\forall t \in T(q)$, $U_time(t|q,S) = P(t|q)$
while $|S| \leq k$ **do**
  while $d \in R$ **do**
    $g(d|q,S) \leftarrow \alpha.V(d|q) + (1-\alpha).(\beta.\sum_c^{A(d)} U_{aspect} + (1-\beta).U_{time})$
  $d^* \leftarrow argmax_d\ g(d|q,c,S)$
  $S \leftarrow S \cup \{d^*\}$
return S

based on rank of d for the given aspect. This approach is a general framework for the diversification of n arbitrary dimensions. The way the dimensions are combined is reflected in the computation of the utility score g(d|q, c, S).

$$g(d|q,c,S) \leftarrow \alpha P(d|q) + (1-\alpha)\sum_C^{\mathbb{C}} \mu(C)V(d|S,C) \tag{3.15}$$

where $\mathbb{C}$ is the set of dimensions and $\mu(C)$ is the weight of a dimension C such that $\sum \mu(C) = 1$. $V(d|S,C)$ is the aspect utility of the given document for the current state of S and the dimension C. The interesting feature of this algorithm though is the computation of the utility of aspects which depends on the rank of the document given an aspect.

$$V(d|S,m) = \sum_c^C w_c \phi(c,S) r(d,c) \tag{3.16}$$

where w_c is the weight of an aspect, $\phi(c,S)$ is utility of the aspect which is calculated as follows:

$$w_c\phi(c,S) = \begin{cases} 1 \& \text{if } S = \varnothing \\ \prod_{d_i}^{S}(1 - r(c,d_i)) \& S \neq \varnothing \end{cases} \tag{3.17}$$

$$r(d,c) = \frac{1}{\sqrt{rank(d,c)}} \tag{3.18}$$

and $rank(d,c)$ is the rank of document d in R|c. This method works well for the TREC diversification task but it is shackled by its generality when it comes to a specific task like historical search. We can naturally add time as a second dimension and use it for diversification. However, they discount each aspect in the same way which might not work well with all aspects. For example, an exponential discounting function is typically associated with time much different from 0/1 or document relevance based discounting for other *non-topical* aspects. We also use this approach as a competitor referred to as MDIV.

Our approach HistDiv(c.f. Algorithm 3) builds on Mdiv [17]; however, we differ significantly in the way we model each dimension's utility.

HistDiv is a greedy algorithm, similar to Ia-Select and Mdiv, and retains the $(1 - 1/e)$ approximation guarantee due to the fact that we treat time windows also as sets. In HistDiv aspects are topical labels assigned to a document with equal probability. Time, on the other hand is modeled as time windows and each time window has its own probability $P(t|q) = \frac{|d_t|}{|d|}$, $d \in R$. Each document, as described earlier, can belong to a single window designated by its publication timestamp whereas it can have one or more aspects. Similar to [28] we consider that aspects are more relevant at certain periods when compared to others (especially for historical intents). Each aspect also has a span ranging from its first occurrence to its last in R.

The importance of an aspect is measured by a utility function $U_{aspect}(c|q, S, t)$, where $c \in \mathcal{A}$, with the exception that we treat the aspect in various time windows differently. The probability of an aspect c in time window t is $P(c|q, t) = \frac{|d_{c,t}|}{|d_t|}$, $d \in R$. This preferential treatment of aspects is due to the usage of the decay function in the time based discounting factor Δ_t while computing utility.

$$U_{aspect}(c|q, S, t) = P(c|q, t) \underbrace{\prod_{d \in S} \left(1 - \frac{1}{1 + e^{-w+|t-t_d|}}\right)}_{\Delta_t} \tag{3.19}$$

We use the decay function suggested in OnlyTime [4] to discount all aspects at any time t, where t^* is the timestamp of the winner document at a given step. In the time dimension, we need to be wary of discrediting a time window too heavily. Consider OnlyTime which produces a result set with high temporal diversity. More formally this is considered as an optimization problem of finding a set Sthat maximizes the following sum:

$$\sum_{t}^{T} \left(P(t|q).(1 - \prod_{d_i}^{R}(1 - P(R|t, t_i).P(R|q, d_i))) \right) \tag{3.20}$$

where $P(t|q)$ is the relative importance of the time point t for the query q. $P(R|t, t_i)$ indicates the probability that a user interested in time point t is satisfied with a document published at time t_i. Similarly, $P(R|q, d_i)$ is the probability that a user issuing the query q is satisfied with the document d_i. OnlyTime is a greedy approach to find S very similar to ia-Select except for the discounting of utility which is based on the decay function Δ_t.

OnlyTime selects relevant documents from important bursts but discounts that burst heavily with the aforementioned decay function so as to get better temporal coverage. This approach to discounting bursts doesn't consider the fact that a single burst could consist of many diverse aspects. For example, in 2000-2001 Giuliani was divorced, diagnosed with cancer and was involved in helping New York recover from 9/11. Hence we discount the burst of the winning document d^* in the time dimension by the weighted proportion of aspects covered by it denoted by Δ_a and where D_t is

the set of all documents in R in time window t.

$$U_{time}(t|q,S) = P(t|q) \underbrace{\prod_{d_t \in S} \left(1 - \frac{|\cup_{c \in A(d_t)} d_{c,t}|}{|D_t|}\right)}_{\Delta_a} \tag{3.21}$$

3.5 Test Collection and Setup

Competitors

Traditional diversity algorithms such as Ia-Select, OnlyTime, Pm2 and MDIV were considered as the main competitors to our method HistDiv. We also strengthen Ia-Select and Pm2 by incorporating time in the algorithm by linearinzing the aspects with the publication year of the document. The improved versions of these algorithms, called TIa-Select and T-Pm2, are also added to the list of competitors. We do not consider xQuad [25] and other diversity techniques that explicitly model subtopics as query expansions due to the absence of a reasonable query log for this time period. MDIV is a general framework for an n dimensional diversity problem. The 2 dimensions we consider for our experiments are document aspects and the publication year. Overall we compare our method HistDiv against the following competitors: TIa-Select, Ia-Select, TPm2, Pm2, MDIV, OnlyTime and a language model LM with Dirichlet smoothing as the baseline. The smoothing parameter is set to 1000.

3.5.1 Test collection

Since there are no established test collections which we can measure the effectiveness of our retrieval model we build our own test collection. As a dataset we use the *Annotated New York Times* collection [2] which qualifies as a news archive since it spans for 20 years, i.e., 1987 - 2007. Also, the timestamps associated with the articles are accurate and do not have to be estimated as in other web collections.

To build the test collection we followed Soboroff's tutorial [27] and suggestions made by Costa in [12]. Soboroff defines 5 basic steps to building a test collection which are as follows:

- Determine the task. - Task abstraction, define relevance, define measures (covered in section 3.2 and 3.3)
- Identify a document collection - *Annotated New York Times* collection.
- Build topics.
- Make relevance judgments.
- Conduct experiments to measure stability of the collection.

A group of experts were tasked with the creation of topics and subtopics for historical query intents given the NYT dataset. They explored the corpus with a simple keyword search interface. To form the topics the experts first described the intents verbosely and then proceeded to identify keywords that represent the user's intent. The user intents chosen are from a set of historically relevant issues related specifically to the USA and some of a more global nature due to characteristics of our news corpus. A key point to note is that topic and subtopic creation was not guided by a query log due to the unavailability of a suitable set of logs spanning this time period. To define the subtopics of each topic the experts used their prior knowledge, documents from the corpus and the history sections from relevant *Wikipedia* articles. Each intent has a description, input keywords (query) and a set of subtopics. We have a total query workload of 30 topics. On average there are 6 subtopics per topic. The topics can be broadly classified into 3 types: *profile queries* for entities like `Rudolph Giuliani` and `the World Trade Organisation`; history of an event like the `reunification of germany` and `team usa soccer world cup`; and controversial subjects like `gay marriage` and `sarin gas`. A key assumption made when creating subtopics is the omission of historical facts that lie outside of the 20 year time period of the NYT corpus.

The evaluations for the test collection are gathered using pooling. In general, competitiors submit runs after which a union is made of these documents to form the pool of documents to be evaluated. The key to pooling is to set a reasonable pool depth. We choose a pool depth of 100 for each topic. Evaluators were instructed to assign binary relevance judgements to topic,subtopic,document triples. For example, an assessor was asked to judge if the document d^* with the headline 'Giuliani Fighting Prostate Cancer; Unsure on Senate' published on 28.4.2000 was relevant to one or more of the manually created subtopics for the topic *Rudolph Giuliani*. When judging the relevance of a document, the assessor is given the headline of the article, the body and the publication date. An article can also be relevant to more than one subtopic.

Once the pools were evaluated, a standard robustness test was carried out with Tia-SBR_k as the primary measure. We selected 25% of the query workload at random and split them into two equal sets. We selected 50% of the runs at random for retrieval depth 10 and calculated ranked the system runs for both sets of queries. We found that the rankings were consistent for $p<=0.05$.

3.5.2 Setup

We chose to mine the aspects of each document using a wikipedia based annotator. By doing so, the aspects we used were possible wikiedia articles that could be linked to from the document. In our implementation aspects are mined using wikiminer[1] on the first 1000 words of each article. We only use mined aspects which have a confidence rating of 0.8 or higher from wikiminer for our computations. For example, some of the aspects mined for document d^* are: New York City, Hillary Rodham Clinton, Prostate cancer and Mayor of New York City. For temporal diversity we use a fixed window size of 1 year akin to OnlyTime, which implies

	T-Sbr	TIA-Ndcg	TIA-Prec.	TIA-Map	TIA-Err	TIA-SBR
LM	0.302	0.209	0.01	0.01	0.023	0.453
TIA-Select	0.325	0.213	0.01	0.012	0.028	0.456
T-PM2	0.182	0.107	**0.011**	0.012	0.022	0.322
IA-Select	0.258	0.161	0.008	0.009	0.02	0.376
PM2	0.295	0.192	0.011	0.011	0.025	0.444
MDIV	0.309	0.209	0.009	0.011	0.025	0.454
OnlyTime	0.344	0.22	0.007	0.01	0.024	0.482
HistDiv	**0.351**	**0.275**	0.01	**0.012**	**0.031**	**0.519**

Table 3.2: Retrieval Effectiveness (k = 10)

	Sbr	TIA-Ndcg	TIA-Prec.	TIA-Map	TIA-Err	TIA-SBR
LM	0.211	0.293	0.01	0.011	0.02	0.321
T-IA-Select	0.203	0.316	0.01	0.012	0.023	0.315
T-PM2	0.131	0.175	0.011	0.012	0.018	0.233
IA-Select	0.164	0.239	0.008	0.009	0.016	0.253
PM2	0.195	0.3	0.01	0.011	0.022	0.299
MDIV	0.228	0.368	0.01	0.011	0.022	0.36
OnlyTime	0.228	0.352	0.008	0.011	0.021	0.358
HistDiv	0.237	0.425	0.011	0.014	0.027	0.377

Table 3.3: Retrieval Effectiveness (k = 5)

	Sbr	TIA-Ndcg	TIA-Prec.	TIA-Map	TIA-Err	TIA-SBR
LM	0.361	0.152	0.01	0.01	0.026	0.513
T-IA-Select	0.387	0.162	0.009	0.011	0.029	0.52
T-PM2	0.222	0.077	0.01	0.012	0.022	0.372
IA-Select	0.329	0.133	0.008	0.009	0.021	0.465
PM2	0.349	0.143	0.01	0.011	0.027	0.497
MDIV	0.376	0.156	0.01	0.01	0.028	0.53
OnlyTime	0.443	0.184	0.007	0.009	0.026	0.573
HistDiv	0.421	0.195	0.01	0.012	0.034	0.579

Table 3.4: Retrieval Effectiveness (k = 15)

$d_t^* \in w(d_t^*)$ where $w(d_t^*) = 2000$. The baseline retrieval model used was a language model with a smoothing factor of 1000. Our competitors are state of the art techniques in search result diversification that we have strengthened for historical query intents. In the first phase of experiments, we tune each competitor on a random set of 10 evaluated topics. These tuned competitors are then evaluated over the entire query workload in the final phase of experiments. We evaluated all competitors for metrics mentioned in 3.3.1 at retrieval depth k=10. For Tia-NDCG and Tia-SBR we set α to 0.5 and β to 0.5. In all metrics we assumed equal distribution of subtopics such that $P(c|q) = \frac{1}{|C|}$. The distribution of time window weights was modeled from the collection as document bursts. For a time window t, $P(t|q) = \frac{|D_{t,q}|}{|D_q|}$ such that $\sum_t P(t|q) = 1$, where $D_{t,q}$ is the set of all documents relevant to q from time window t and D_q is the set of all documents relevant to q.

	Sbr	TIA-Ndcg	TIA-Prec.	TIA-Map	TIA-Err	TIA-SBR
LM	0.391	0.109	0.009	0.01	0.026	0.544
T-IA-Select	0.447	0.131	0.009	0.011	0.031	0.58
T-PM2	0.25	0.061	0.01	0.012	0.023	0.408
IA-Select	0.382	0.11	0.009	0.009	0.023	0.526
PM2	0.381	0.107	0.01	0.011	0.028	0.526
MDIV	0.401	0.115	0.01	0.01	0.027	0.563
OnlyTime	0.481	0.134	0.006	0.009	0.026	0.603
HistDiv	0.48	0.151	0.009	0.011	0.034	0.637

Table 3.5: Retrieval Effectiveness (k = 20)

3.6 Results & Discussion

In this section we analyze and discuss the outcomes of our experiments. In Table 3.2 we present the effectiveness of the considered approaches with respect to the temporal measures introduced in Table 3.1 for k = 10. We see a similar trend for high values of k. Firstly, we note that HistDiv outperforms other competitors in all measures expect Tia-Precision. Also, approaches which take time into account fare better than the non-temporal methods with the exception being Tpm2. This is particularly evident in the *user-centric* measures like Tia-Ndcg and Tia-Err.

The time awareness added to the user centric measures indicates the extent to which a user has to scroll down the list in order to find a relevant result to his intent from the corresponding important time period. Tia-Err helps guage the extent to which the average user has to scroll through this list in order to find a relevant result whereas Tia-Ndcg is used to measure the cummulative information gain of a list. The gain for each document in the list is discounted depending on its rank due to the assumption that the user is less likely to keep scrolling down the list. Thus documents from lower ranks contribute little to the cummulative gain of a ranked list. HistDiv outperforms its competitors in both user centric measures due to its unique calculation of time and aspect utility. The first document in a ranked list generated by HistDiv is from the most important time window with the most important aspects or vice versa depending on the tuning parameters. The aspects of the document are then discounted only within a certain time interval defined by the decay function. This allows HistDiv to reselect the same aspect but from a different time window thereby increasing its temporal awareness. Similarly if the document only covers a small portion of important aspects in a time window, HistDiv can revisit the time window to find a relevant document with diverse aspects which could increase the probability of covering a different subtopic.

Since HistDiv tends to select important aspects from important time points and hence tends to satisify historical intents better than other methods which optimize for (a) one of the dimensions like OnlyTime or Ia-Select (b) rewards aspects (topical or temporal) which have a high mass contribution like Tpm2 or Pm2. Secondly, HistDiv outperforms other approaches when it comes to T-Sbr. To analyze this a bit further, we consider the temporal and aspect-based contributions to subtopic recall. In our experiments we see that OnlyTime performs best when in the temporal dimension, hence the contribution of the temporal dimension towards T-Sbr

1	Giuliani, Shouting for Quiet, Fights to Concede Graciously - 1989
2	Helen Giuliani, 92, Mother Of Former New York Mayor passed away - 1992
3	Giuliani Pulls a Negative TV Ad As Dinkins Broadcasts First One - 1989
4	Why Giuliani Only Came Close - 1989
5	New York Label May Not Fit All In Giuliani Run - 2007

Table 3.6: Results for `Rudolph Giuliani` - *LM*

1	GIULIANI AND LAUDER: 2 PATHS TO SAME GOAL -1989
2	Giuliani Swamps Lauder In G.O.P. Mayoral Primary - 1989
3	In Debate, Dinkins Ties Innis to Giuliani -1993
4	BADILLO DROPS OUT OF RACE FOR MAYOR WITH SPARSE FUNDS - 1993
5	Who Has to Do What As New York Prepares To Pick a New Mayor - 1989

Table 3.7: Results for `Rudolph Giuliani` - *T-PM2*

1	GIULIANI AND LAUDER: 2 PATHS TO SAME GOAL - 1989
2	Pensively, Giuliani Still Debates His Future - 2000
3	Ready for a Race, Mayor Goes to Saratoga - 1999
4	In Debate, Dinkins Ties Innis to Giuliani - 1993
5	Political Memo; Cuomo and Giuliani, Looking For Allies, Find Each Other - 1994

Table 3.8: Results for `Rudolph Giuliani` - *T-IA-Select*

1	Bush Offers Giuliani Help On the Right - 1999
2	Reporter's Notebook; Giuliani Is by the Sea; Could It Be a Vacation? - 1998
3	Giuliani Wisely Bides Time on Endorsement in Governor's Race - 1994
4	Giuliani Fighting Prostate Cancer; Unsure on Senate - 2000
5	In Homestretch of Campaign, Mayor Endorses Bloomberg - 2001

Table 3.9: Results for `Rudolph Giuliani` - *HistDiv*

is high. However, it still does not outperform HistDiv suggesting that although it selects results from different time periods it tends to choose similar topics. The linearized methods understandably optimize the coverage in the projected aspect-time space but that does not lead to maximizing overall coverage. Interestingly, in proportionality-based approaches Tpm2 fares worser than Pm2 which is unlike the intent-aware approaches. A closer examination suggests that this is due the inability of the proportionality-based approaches to penalize the over representation of an aspect. Proportionality-based approaches attach high preferences to results where certain time windows are dominant with a given aspect and hence suffer in the overall coverage.

Although T-Sbr measures the coverage in both the time and aspect dimension, at times it is deterimental to just introduce a spread over time if the time windows covered are not important. This is where Tia-Sbr (introduced in Section 3.3.2) is a more accurate measure which takes into account the relative importance of time windows and aspects while computing coverage. Table 3.10 presents the effectiveness of various measures at different values of k alongwith the win-loss values in braces. First, HistDiv's modeling of aspect importance decaying over time helps it choose the most relevant aspects for a given time period. These aspects can be covered again thanks

to the temporal utility function which discounts aspect utility only within a certain time frame. Second, HistDiv also reconciles multiple important aspects in a single time window, unlike OnlyTime, thus capturing multiple important events from the same time window rather than trading-off with less relevant events from a different interval. Mdiv is the only other algorithm that also accounts for two dimensions like HistDiv explicitly. However, a generic discounting scheme based on ranks suffers in this scenario because (a) it does encode the degree of relevance per aspect because of using ranks and (b) it treats time similar to the other (more topical) aspect whose semantics are clearly different. These reasons contribute consequently to HistDiv outperforming other competitors both in terms of the actual value and in terms of win-loss ratio for Tia-Sbr.

Finally, we note that although HistDiv performs better in most of the scenarios there are cases where it falters specifically in Tia-Precision. This can be atrributed to the nature of some topics which are bound very closely to a small time window with very little aspect diversity, like `reunification of germany`. HistDiv covers enough important aspects in the important time windows and subsequently diversifies to find aspects in other less relevant windows. In such a scenario proportionality-based approaches do well in precision since they reward documents in the important aspect due to the dominant mass of the time window.

Rudolph Giuliani Table 3.11 shows the performance of algorithms specifically for `Rudolph Giuliani`. We find that HistDiv performs better than its competitors in all metrics including precision based metrics at retrieval depth 10. We observe a similar trend at varying depths. Figure **??** shows the temporal distribution of the result set $\mathcal{R}$ as well as the diversified set $\mathcal{S}$ for various competitors. We observed that OnlyTime being a purely temporal diversification algorithm produces a temporal distribution similar to the global trend. T-PM2 T-IASel also try tend to follow the temporal distribution of $\mathcal{R}$ because of the linearization of the aspect-time space. This shows that the mined aspects have an inherent temporal nature, i.e. some aspects only exist at certain points in time. However the selective treatment of time and aspects causes HistDiv to not adhere to the global temporal distribution trend as strictly as its competitors. HistDiv is also aware of the aspect dimension explicitly which helps it produce a document set that is more suitable for a historical intent as shown by the numbers in 3.11. The top 5 results for HistDiv, shown in Table 3.9 shows better diversity in both time and aspects when compared to its competitors which seem to be dominated by his election campaigns.

Consider the topic Summer Olympics Doping Scandals. The user's intent is to know the history of all doping scandals at the summer olympics between 1987 and 2007. The subtopics of this topic as per our test collection are:

1. 1988 Seoul - doping mostly in wrestling and judo

	k =5	k =10	k =15	k =20
LM	0.318	0.453	0.502	0.540
T-IA-Select	0.325(15/15)	0.456(16/14)	0.525(17/13)	0.574(18/12)
T-PM2	0.243(8/22)	0.322(6/24)	0.376(7/23)	0.409(7/23)
MDIV	0.350(13/17)	0.454(12/18)	0.520(12/18)	0.555(10/20)
OnlyTime	0.362(12/18)	0.482(14/16)	0.569(18/12)	0.605(19/11)
HistDiv	**0.376**(17/13)	**0.519**(22/8)	**0.577**(25/5)	**0.641**(24/6)

Table 3.10: Tia-SBR(win/loss) at k=5,10,15 & 20

	Sbr	TIA-Ndcg	TIA-Prec.	TIA-Map	TIA-Err	TIA-SBR
LM	0.185	0.04	0.003	0.004	0.007	0.203
T-IA-Select	0.333	0.223	0.008	0.009	0.023	0.457
T-PM2	0.148	0.052	0.006	0.006	0.011	0.18
IA-Select	0.259	0.21	0.005	0.004	0.006	0.32
PM2	0.074	0.017	0.004	0.004	0.004	0.087
MDIV	0.222	0.16	0.004	0.004	0.007	0.298
OnlyTime	0.333	0.314	0.008	0.011	0.029	0.457
HistDiv	0.407	0.461	0.01	0.013	0.033	0.628

Table 3.11: Retrieval Effectiveness (k = 10) for the topic `Rudolph Giuliani`

	Sbr	TIA-Ndcg	TIA-Prec.	TIA-Map	TIA-Err	TIA-SBR
T-PM2	0.008	0.044	0.005	0.077	0.111	0.007
HistDiv	0.004	0.217	0.013	0.192	0.246	0.005

Table 3.12: Results for *summer olympics doping scandals*

2. 1992 Barcelona - mostly athletics related cases
3. 1996 Atlanta - very few cases. only 2 in athletics.
4. 2000 Sydney - quite a few in athletics and weight lifting. Marion Jones was the most popular one.
5. 2004 Athens- Massive increase in doping compared to previous years
6. Measures taken by the IOC to combat doping

From the subtopics there is no clear important time window for this topic and it seems like a straightforward temporal diversity case. However since the underlying corpus is the New York Times, the majority of doping stories emanate from the Marion Jones case from 2000. Reports in the following years then continue to follow Jones and talk about her until the 2004 olympics. The 2004 olympics saw a sudden spike in doping cases as well leading to a large number of articles publised about doping in 2004. Thus the time awareness of metrics here hurts methods which actually try to introduce wide temporal spread to cover the other subtopics. Temporal PM2 due to its nature to proptionally represent aspects restricts the majority of top documents to the dominant time window. ASPTD on the other hand does not favour

proptionality so it tends to start traveling outside the important time windows to look for diverse events once it has covered the important time periods and the most important aspects. Time awareness in the metrics here penalizes ASPTD unfairly. For topics with disproptionately heavy time windows, temporal PM2 does well.

Another case where HistDiv is not the best is for historical intents whose subtopics are broad disjoint categories like *elections in the middle east*. Here the subtopics are divided based on region and no temporal aspect is present. As expected in such cases, Ia-Select is the best performing algorithm. Since there are no major time windows the lack of a temporal dimension does not hurt the performance of the algorithm. From these observations we can conclude that HistDiv, without specific parameter tuning, is not particularly competitive to diversification problems of a purely temporal or non-temporal nature.

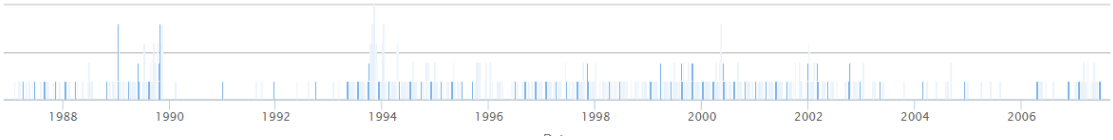

Figure 3.1: Temporal distribution of top 1000 documents for `Rudolph Giuliani`

Figure 3.2: Temporal distribution of top 1000 documents for `Rudolph Giuliani` after diversification using **T-PM2**

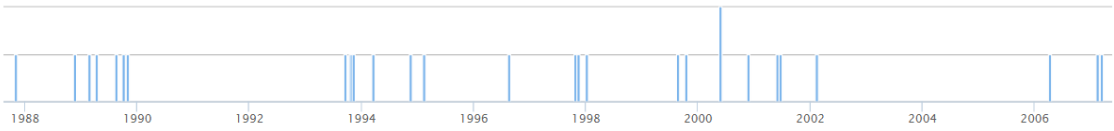

Figure 3.3: Temporal distribution of top 1000 documents for `Rudolph Giuliani` after diversification using **T-IASEL**

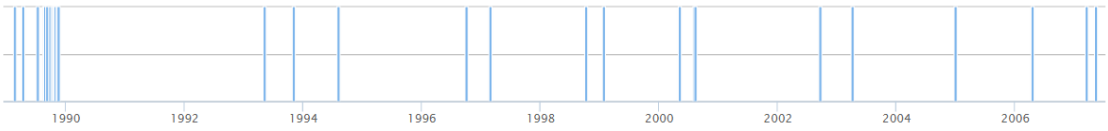

Figure 3.4: Temporal distribution of top 1000 documents for `Rudolph Giuliani` after diversification using **T-MDIV**

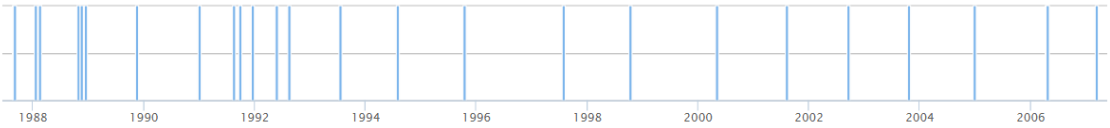

Figure 3.5: Temporal distribution of top 1000 documents for `Rudolph Giuliani`after diversification using **Only Time**

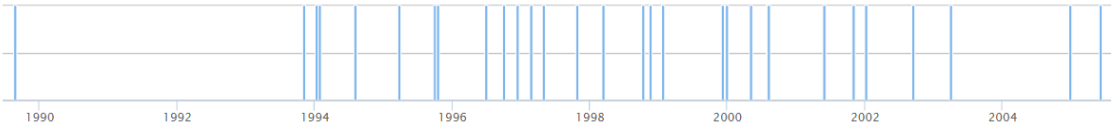

Figure 3.6: Temporal distribution of top 1000 documents for `Rudolph Giuliani`after diversification using **HistDiv**

3.7 Related Work

One of the first attempts to incorporate time in search result ranking was suggested by Li et al. [20] who used temporal language model approach where time and term importance are handled implicitly. Various approaches have been suggested that consider time more explicitly. Identifying the time period most useful for query expansion [19, 9] has been shown to improve result ranking. Ranking for recency on the other hand considers the freshness of a document when ranking [16].

For queries which are temporally ambiguous, temporal diversification is applied. Burst detection is first used to find the possible periods of interest for the query, also known as the query's temporal profile[15, 23], followed by selecting documents from these bursts. Berberich et al. in [4] proposed a diversification model which considers time windows as a set of intents for a query while modeling the importance of each intent as the weight of its burst. In traditional aspect-based diversification tasks like [11, 10], intent importance is considered static over time. However, intent importance was shown to vary across time; thus affecting the diversity evaluation of queries issued at different time points [28]. Keeping this in mind, Kanhabua et al.[22] consider the time at which the query is issued to diversify intents based on their temporal significance at that time. Their approach also explicitly models time and aspects, although latent, but rewards recency. HistDiv, on the other hand, is query-time agnostic, since it is intended for historical search, and seeks to diversify documents based on both time and aspects.

Diversity in search (both explicit and implicit) has seen a rich body of literature lately in [13, 7, 3, 25, 8, 29, 21, 14]. However, none of them take time into account or model the historical information intent.

Work on temporal test collections for information retrieval has seen limited attention. The work which comes closest to our test collection is *Temporalia*[18] however, our query intents are different and moreover their dataset does not have the temporal spread required for our historical queries.

4 The Historical Search System

To show researchers the qualitative results of the HistDiv algorithm, a search system was developed. The system is also a showcase for the baselines so that results from various algorithms can be compared qualitatively. A seperate system was also developed for evaluators to judge documents from the pools. In this chapter, the architecture and user interface of the historical search system are described in detail.

4.1 Architecture

4.1.1 Server side

The backend of the Historical Search system comprises of 2 major components: the **retrieval module** and the **aspect miner**. The development of both components was carried out in Java. The documents from the NYT corpus were first indexed using the standard Lucene 4.0 API. All fields for all documents were indexed using the standard analyzer.

The aspect miner is responsible for mining topics from the documents. Wikiminer is a REST API that is freely available. A java client was developed for the annotation service so that it can be easily used in the aspect miner. The mined topics with their confidence scores are stored in a Redis database and not in the Lucene index. This is because we only need to lookup the topics for each document when computing diversity. Redis is a high performance key value store that maximizes throughput. The data is stored in JSON which Redis supports natively. The aspect miner component is heavily parallelized and can be used during index creation as well as on demand.

The main component of the system though is the ranking component housed inside the retrieval module. A REST API using the JAX-RS specification is the interface between frontend and backend. The API has a set of endpoints that cater to different funtionalities of the system. The data exchange format used by the API is JSON. The major endpoint is the `reRank` endpoint which accepts a HTTP GET request with the following parameters:

- q: the query specified by the user

- dmin: start date filter
- dmax: end date filter
- method: the type of retrieval model to use.
- k: the number of results

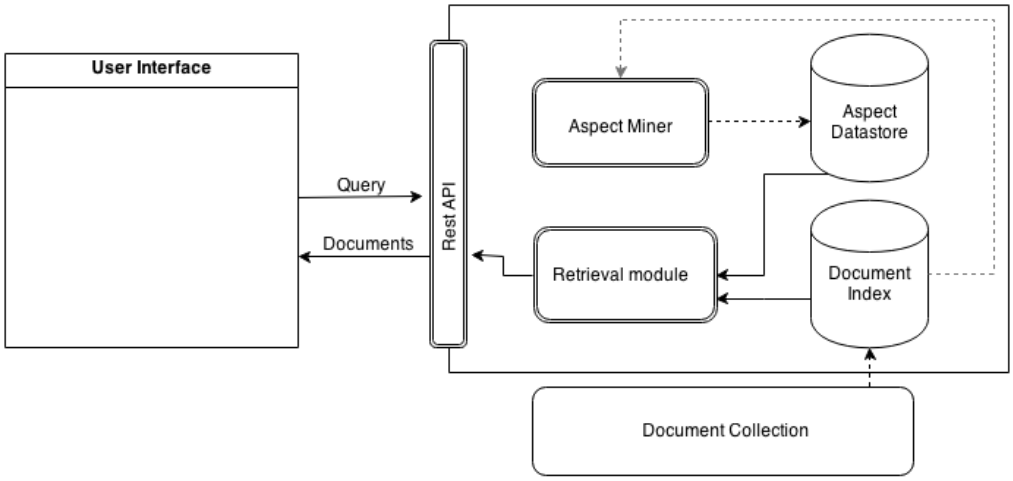

Figure 4.1: System Architecture

The architecture of the system is shown in Figure 4.1 where the dashed lines represent processes that have completed before the system went online. Every competitor mentioned in the experiments section is available to the user. The user can also specify a date range to limit his search results. When the reRank endpoint receives a request, it first retrieves the top 1000 documents using the query q and specified date range. If the range is `null` then the whole 20 year span of the corpus is considered. The results returned from the index are then given as input to the algorithm mention in the method parameter. The algorithm then returns the top k results from the 1000 given as input. The results are returned to the user as a JSON array with all the fields of each document populated. CORS headers are also sent with the result to allow clients running on different servers to use the API as well.

4.1.2 Client side

The frontend of the Historical Search system is a single page client side application developed using Backbone.js. Backbone.js is a javascript framework for building interactive client side applications that work with data through REST APIs. The UI framework used was Bootstrap 3.0, a responsive framework based purely on CSS and Javascript. The UI itself consists of 3 components: the search bar, the timeline and the search result display.

The search bar consists of an input text box for the user to enter his query and a selector for the type of diversification algorithm to use. There is also a search button

for the user to initiate his search. When the user initiates a search, an AJAX request is made to the reRank endpoint of the rank API with the user's query and method. The date filters are initially set to null. Once the results are returned, the timeline component creates the input JSON data it needs from the returned results. Once the timeline component has initialized, the timeline and search results are rendered on the screen. Highcharts API is used to generate the timeline. The timeline is a bar chart of the number of documents at a given date. Highcharts also has a feature for zooming into tinmelines by highlighting the required area on the timeline. This zooming in triggers another search request to be fired albeit with the date range filter now representing the boundaries of the newly selected portion of the timeline. By clicking on the back arrow of the browser the user can zoom out and return to his previous state. Each bar in the timeline when clicked displays a preview of the article from that date.

The search results are displayed in a newspaper style format as shown in Figure 4.2. Each document is represented by its headline, publication date and body. News articles which are more relevant to the query according to the ranking method chosen are given more area to display the contents of the body. To neatly present these results of uneven area freewall.js was used. It uses a packing algorithm to reduces the amount of white space between articles, thus giving it the look of a newspaper.

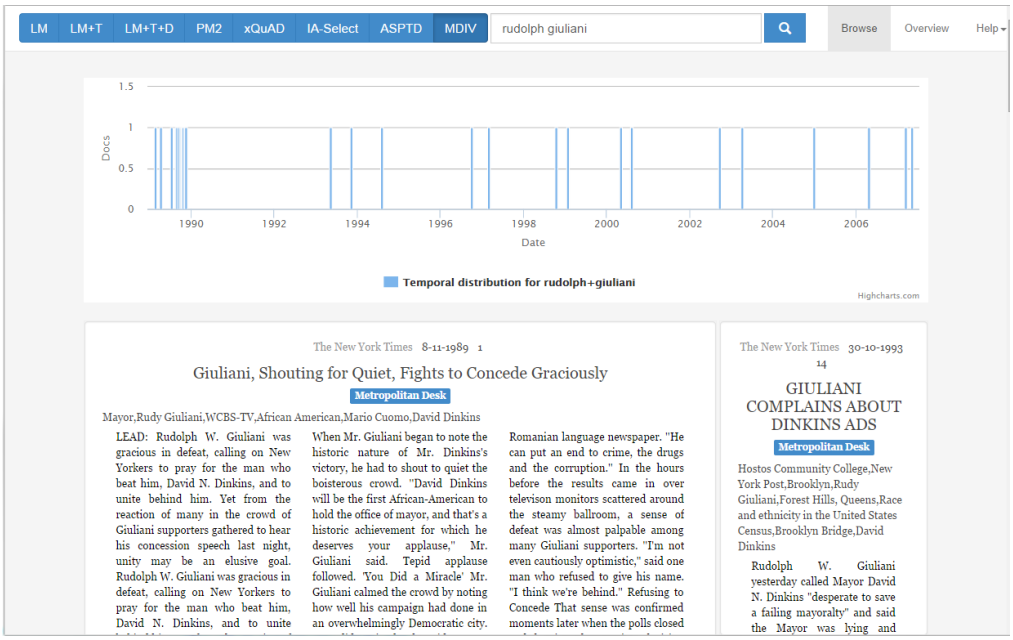

Figure 4.2: Newspaper interface with timeline.

The system was deployed on a live server. The same system also consists of the test collection evaluation setup; both the pooling logic and the interface.

- URL: `http://pharos.l3s.uni-hannover.de:7080/ArchiveSearch/starterkit/`
- Evaluation: `http://pharos.l3s.uni-hannover.de:7080/ArchiveSearch/starterkit/rele`

5 Outlook

5.1 Conclusion

With more nations working actively to preserve the web due to the importance it plays in our lives, scholars from various fields have started looking to web archives as a data source waiting to be analyzed. We find that scholarly research with web archives carried out until today can be broadly classified as link based and content based studies. The link structure of web archives has been studied on a quantitative scale by Internet research institutes and computer scientists. They either study the link structure over time to analyze the evolution of the web or try to correlate the linking structure to real world phenomena. In this thesis we first looked at the usage of web archives by scholars. According to Brügger there are 4 major steps when conducting any type of research on web archives: corpus creation, analysis, dissemination and storage. An ideal system for web archive research should support all 4 steps however we find that adequate steps are only now being made to help researchers with the first step: corpus creation. Humanities scholars are usually interested in doing small scale qualitative and quantitative studies for which they need to explore web archives for relevant material. But the access to web archives with current web archive access systems like the wayback machine and rudimentary keyword search is not satisfactory. There are many other factors as well that have led to the lack of humanities studies on web archives. We attempted to illicit these through literature surveys and group discussions with humanities scholars conducting web archive research. Some of the interesting outcomes of this study were:

- The absence of abilities to explore the web archive discourage scholars from pursuing their ideas. Search engines are the predominant way of exploring the web today whereas archives seem to be left behind.
- The wayback machine is not an effective tool for building a corpus unless you know the exact set of URLs you are interested in. Full text search should be provided for researchers who want to explore the archive.
- More details of the crawl should be exposed so that scholars can motivate their

corpus of study better. The details of the crawl strategy should be added to the meta data of all documents.

- Scholars are used to working with well curated archives. Web archives pose greater burden for scholars in corpus creation since they need to curate the materials themselves.
- Due to the inconsistent nature of web archives, an error margin should be defined and an approximation of confidence as well even for small scale qualitative studies.
- Algorithms and features used to help scholars find new documents should be made transparent to users so that they can effectively document the corpus creation process.
- A web archive search system should allow for the corpus and its history of creation to be exported in standard formats researchers are used to working with.
- Scholars are interested in leveraging digital methods to analyze their data but find it hard to find tools flexible enough for their needs.

Scholars also saw potential in the usage of web archives as a playground for combining qualitative and quantitative studies. Based on our discussions with the scholars who attended the summer school in Aarhus University, we found that many would be interested in the ability to scale results up. They also bemoaned the lack of analysis tools currently available to work with web archives. The lack of technical expertise was clearly affecting a scholar's ability to conduct his research even though they knew what kind of analysis they wanted to do. We proposed that computer scientist should bridge this gap by either providing tools to support scholars or work with them on scaling up their hypothesis using pattern recognition and data mining techniques.

Initiatives are being made kick-start humanities research in web archives but to do the necessary infrastructure to first access web archives has to be in place. These initiatives are currently being carried out by national libraries and institutions like the Internet Archive and the IIPC. The BUDDAH project, a joint effort between the British Library and leading Universities in the UK, is working towards building a web archive search system by working in tandem with humanities scholars working with the UK web archive. We held discussions with members of the BUDDAH project and scholars to identify the pitfalls with the current web archive search system they were working with. The British Library web archive search prototype already provides its users with keyword search and filtering based on domain and crawl date. As noted by Costa et. al. current IR techniques like keyword search do not fare well in web archive search. We observed this first hand with the British Library's choice to rank search results by crawl date. This makes it very difficult for users to easily comprehend a large result set. We decided to better understand a scholar's search intent by analyzing written descriptions of proposed studies on web archives by humanities

scholars involved in the BUDDAH project. We found that scholars are interested in studying results over time and covering a variety of aspect for their topic.

To this end we introduced the notion of a historical query intent over longitudinal collections like web archives. Historical query intent was modeled as a 2 dimension diversification problem where one dimension is time and the other aspects. In this thesis we limit ourselves to a subset of web archives: the news archive. We proposed a new evaluation metric Tia-SBR to evaluate the performance of retrieval models for this task. We also adapted existing diversity metrics to take time into account. To evaluate the retrieval models we built a new temporal test collection based on 20 years of the *New York Times* collection. We strengthened existing 1 dimensional diversity methods by linearizing the aspect-time space into a single set. We also consider temporal diversity retrieval models in our experiment. We introduced HistDiv which shows improvements over temporal and non-temporal methods for most of the time-aware diversification methods. We also outperform all competitors in Tia-SBR showing the suitability of our approach for historical query intents. We observe that HistDiv works well for topics which have aspects that span across multiple time intervals and have fluctuating importance at different times. It trades-off nicely between important aspects and important times which we perceive as important in historical search. HistDiv does not perform quite as well for queries which only one dominant aspect at a certain time window. We also described the architecture of the History Search system which consists of: a REST API to access the retrieval models used in our experiments, a user interface designed specifically to show results as newspaper articles and a time line to provide an overview and ability to filter.

5.2 Future Work

The performance of the HistDiv algorithm encourages us to look beyond the traditional approaches and devise methods more suited to archives. HistDiv currently considers time as a set of disjointed windows. It will be interesting to see if bursts rather than time windows are more effective for historical query intents. In our current implementation, subtopics are mined using wikiminer which is a relatively naive approach. In the future, we want to experiment with different types of aspect mining like LDA for instance. We also want to add more dimensions to the model like geographic locations. Just like aspect utility decays based on time, we can also decay it based on location. The absolute distance between lat long coordinates can be used or a taxonomy based approach.

The results also indicate where HistDiv performs badly and other algorithms perform well. We believe that if the user is presented with a choice of algorithms, she can effectively select the algorithm best suited to her need. A step beyond this would be to train a classifier that selects the appropriate retrieval model based on features from the query's aspect and temporal profile.

The test collection we built is still relatively small when compared to collections by TREC. We must strive to add to this test collection, by increasing the workload, or explore different avenues for evaluating retrieval models meant for archive search.

For our test collection it will also be interesting to get actual historians to provide topics. More research is also needed to construct test collections for other query intents for archives.

Having performed admirably in a news archive test collection, it will be interesting to test HistDiv and the other methods on a web archive or at least a subset which is not just news. Existing web crawl datasets only span a small period of time but it might be interesting nonetheless to experiment with smaller time window sizes for these collections. Another important temporal feature that we want to consider is the temporal references within an article. It is reasonable to assume that major bursts are usually covered by summary articles after the end of the burst. These articles are valuable but may get neglected because they do not occur within the burst based on our current algorithm. HistDiv's ability to find documents from important time windows and aspects can also be applied in Temporlia's query classification task to interesting effect.

There is also much to be done to help scholars better engage with web archives. In this thesis we have only scratched the surface of the issues surrounding intelligent access to web archives. We must work together with scholars in order to develop effective algorithms and user friendly systems so that the researchers of tomorrow can conduct their studies with far greater ease than today.

Bibliography

[1] New york times archives. http://wikipedia-miner.cms.waikato.ac.nz.

[2] The New York Times Annotated Corpus. http://corpus.nytimes.com.

[3] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying search results. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pages 5–14, New York, NY, USA, 2009. ACM.

[4] K. Berberich and S. Bedathur. Temporal diversification of search results. In *SIGIR 2013 Workshop on Time-aware Information Access (TAIA 2013)*, 2013.

[5] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[6] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and regression trees*. CRC press, 1984.

[7] J. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336. ACM, 1998.

[8] B. Carterette and P. Chandar. Probabilistic models of ranking novel documents for faceted topic retrieval. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 1287–1296, New York, NY, USA, 2009. ACM.

[9] J. Choi and W. B. Croft. Temporal models for microblogs. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2491–2494. ACM, 2012.

[10] C. Clarke, N. Craswell, I. Soboroff, and E. Voorhees. Nist, overview of the trec2011 web track. In *Proceedings of TREC*, pages 500–295, 2011.

[11] C. L. Clarke, N. Craswell, and I. Soboroff. Overview of the trec 2009 web track. Technical report, DTIC Document, 2009.

[12] M. Costa and M. J. Silva. Evaluating web archive search systems. In *Web Information Systems Engineering-WISE 2012*, pages 440–454. Springer, 2012.

[13] V. Dang and B. W. Croft. Term level search result diversification. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 603–612, New York, NY, USA, 2013. ACM.

[14] V. Dang and W. B. Croft. Diversity by proportionality: An election-based approach to search result diversification. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 65–74, New York, NY, USA, 2012. ACM.

[15] F. Diaz and R. Jones. Using temporal profiles of queries for precision prediction. In *SIGIR*, volume 4, pages 18–24, 2004.

[16] A. Dong, Y. Chang, Z. Zheng, G. Mishne, J. Bai, R. Zhang, K. Buchner, C. Liao, and F. Diaz. Towards recency ranking in web search. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 11–20. ACM, 2010.

[17] Z. Dou, S. Hu, K. Chen, R. Song, and J.-R. Wen. Multi-dimensional search result diversification. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 475–484, New York, NY, USA, 2011. ACM.

[18] H. Joho, A. Jatowt, and R. Blanco. Ntcir temporalia: a test collection for temporal information access research. In *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, pages 845–850. International World Wide Web Conferences Steering Committee, 2014.

[19] N. Kanhabua and K. Nørvåg. Determining time of queries for re-ranking search results. In *Research and Advanced Technology for Digital Libraries*, pages 261–272. Springer, 2010.

[20] X. Li and W. B. Croft. Time-based language models. In *Proceedings of the twelfth international conference on Information and knowledge management*, pages 469–475. ACM, 2003.

[21] S. Liang, Z. Ren, and M. de Rijke. Fusion helps diversification. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, pages 303–312, New York, NY, USA, 2014. ACM.

[22] T. N. Nguyen and N. Kanhabua. Leveraging dynamic query subtopics for time-aware search result diversification. In *ECIR*, pages 222–234, 2014.

[23] M.-H. Peetz, E. Meij, M. de Rijke, and W. Weerkamp. Adaptive temporal query modeling. In *Advances in Information Retrieval*, pages 455–458. Springer, 2012.

[24] S. J. Russell, P. Norvig, J. F. Canny, J. M. Malik, and D. D. Edwards. *Artificial intelligence: a modern approach*, volume 2. Prentice hall Upper Saddle River, 2003.

[25] R. L. Santos, C. Macdonald, and I. Ounis. Exploiting query reformulations for web search result diversification. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 881–890, New York, NY, USA, 2010. ACM.

[26] R. L. Santos, C. Macdonald, and I. Ounis. Intent-aware search result diversification. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pages 595–604, New York, NY, USA, 2011. ACM.

[27] I. M. Soboroff. Building test collections: an interactive tutorial for students and others without their own evaluation conference series. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 1132–1132. ACM, 2013.

[28] K. Zhou, S. Whiting, J. M. Jose, and M. Lalmas. The impact of temporal intent variability on diversity evaluation. In *Advances in Information Retrieval*, pages 820–823. Springer, 2013.

[29] Y. Zhu, Y. Lan, J. Guo, X. Cheng, and S. Niu. Learning for search result diversification. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, pages 293–302, New York, NY, USA, 2014. ACM.

Appendix

['topic': 1, 'type': 'person', 'query': 'rudolph giuliani', 'description': 'I want to know the history of rudolph giuliani the american politician between 1987-2007', 'subtopics': ['subtopic':1, 'type': 'span', 'description': 'Giuliani the litigator. Life as a lawyer in New York.' , 'subtopic':2, 'type': 'span', 'description': 'Mayoral Campaigns - 1989 (losing to Dinkins) 1993 (improving police protection, beating dinkins) 1997(first Republican to win a second term as mayor) 2001 (just before 9/11)' , 'subtopic':3, 'type': 'span', 'description': 'Mayoralty - mayor of New York City from 1994 through 2001. The major obstacles he had to overcome during his time as mayor. (Law enforcement, Appointees as defendants, City services- schooling,)' , 'subtopic':4, 'type': 'span', 'description': '2000 U.S. Senate campaign. His main opponent being Hilary Clinton.' , 'subtopic':5, 'type': 'burst', 'description': 'September 11 terrorist attacks. Giulianis work for helping New York recover' , 'subtopic':6, 'type': 'span', 'description': 'Post-mayoralty- what did Giuliani do in the political scene after leaving his post as mayor (after 2001) (running for president for 2008, endorsing bush for re-election in 2004, Giuliani founded a security consulting business, Giuliani Partners LLC in 2002, In 2005, Giuliani joined the law firm of Bracewell and Patterson LLP)' , 'subtopic':7, 'type': 'span', 'description': 'His personal life - knighthood, time person of the year, cancer, affair, divorce'] , 'topic': 2, 'type': 'location', 'query': 'silicon valley', 'description': 'I want to know the history of the silicon valley in the United States Of America 1987-2007', 'subtopics': ['subtopic':1, 'type': 'span', 'description': 'The rise of the micro computer and microchips industry in the 1980s/70s' , 'subtopic':2, 'type': 'span', 'description': 'Silicon valley's job scenario improves' , 'subtopic':3, 'type': 'burst', 'description': 'The Dot Com Bubble - mid 1990s - Silicon Valley is generally considered to have been the center of the dot-com bubble, which started in the mid-1990s and collapsed after the NASDAQ stock market began to decline dramatically in April 2000' , 'subtopic':4, 'type': 'span', 'description': 'Famous software companies starting in silicon valley'] , 'topic': 3, 'type': 'event', 'query': 'reunification of germany', 'description': 'I want to know the history behind the reunification of germany in 1990', 'subtopics': ['subtopic':1, 'type': 'span', 'description': 'Precursors to reunification - pre 1989' , 'subtopic':2, 'type': 'burst', 'description': 'Process of reunification - Cooperation, Economic merger, German Reunification Treaty, Constitutional merger, International effects, Day of German Unity - 1989 -1990' , 'subtopic':3, 'type': 'span', 'description': ' Foreign support and opposition - Britain and France, The rest of Europe and America' , 'subtopic':4, 'type': 'span', 'description': ' Aftermath - Full German sovereignty, confirmation of borders, withdrawal of the last Allied Forces, Cost of reunification, Inner reunification - post 1990'] , 'topic': 4, 'type': 'product', 'query': 'sony playstation', 'description': 'I want to know the history of the sony playstation between 1987 and 2007', 'subtopics': ['subtopic':1, 'type': 'acyclic bursts', 'description': 'playstation releases' , 'subtopic':2, 'type': 'span', 'description': 'Sony playstation vs the xbox' , 'subtopic':3, 'type': 'span', 'description': 'Sony playstation vs the nintendo and sega' , 'subtopic':4, 'type': 'span', 'description': 'online multiplayer gaming' , 'subtopic':5, 'type': 'span', 'description': 'popular video game titles for the playstation'] , 'topic': 5, 'type': 'sporting event', 'query': 'team usa soccer world cup', 'description': 'I want to know the history behind team USA at the soccer world cups between 1987 and 2007 (mens world cup - from 1990 held every 4 years)', 'subtopics': ['subtopic':1, 'type': 'burst', 'description': 'USA world cup 1994' , 'subtopic':2, 'type': 'span', 'description': 'Tactics, team selection and preparation for the world cup. Famous players and coaches.' , 'subtopic':3, 'type': 'span', 'description': 'growth of soccer in the united states' , 'subtopic':4, 'type': 'cyclic bursts', 'description': 'Team performance in the world cup and qualifiers' , 'subtopic':5, 'type': 'cyclic bursts', 'description': 'Womens world cup'] , 'topic': 6, 'type': 'event', 'query': 'elections in the middle east', 'description': 'I want to know the history behind the elections in

List of Figures

List of Tables

List of Algorithms