TAER: Time-Aware Entity Retrieval

Exploiting the Past to find Relevant Entities in News Articles

Gianluca Demartini*†[1], Malik Muhammad Saad Missen*[2], Roi Blanco[3], Hugo Zaragoza[3]
[1]L3S Research Center, Appelstrasse 9a 30167 Hannover, Germany
[2]IRIT Toulouse, France
[3]Yahoo! Research, Diagonal 177 08018 Barcelona, Spain
demartini@L3S.de, missen@irit.fr, roi@yahoo-inc.com, hugoz@yahoo-inc.com

ABSTRACT

Retrieving entities instead of just documents has become an important task for search engines. In this paper we study entity retrieval for news applications, and in particular the importance of the news trail history (i.e., past related articles) in determining the relevant entities in current articles. This is an important problem in applications that display retrieved entities to the user, together with the news article.

We analyze and discuss some statistics about entities in news trails, unveiling some unknown findings such as the persistence of relevance over time. We focus on the task of query dependent entity retrieval over time. For this task we evaluate several features, and show that their combinations significantly improves performance.

Categories and Subject Descriptors:
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms:
Algorithms, Experimentation, Measurement

Keywords: Entity Retrieval, Time-aware Search

1. INTRODUCTION

Entity search has become a new important feature of current Web search engines. It helps people find directly the information they are after and to reformulate the query precisely in a natural way. For such reasons, Entity Retrieval (ER) is becoming a major area of interest in Information Retrieval (IR) research and is quickly being adopted in commercial applications. Published research on ER has concentrated on the tasks of people search and finding related entities [1], where evaluation corpora have been developed (at TREC [3] and INEX [4] respectively). One of the promising areas of application of ER models in the commercial world is in *news search*[1]. A possible application consists in enriching the user interface by placing retrieved entities next to the news article the user is currently looking at.

In this paper we address the problem of ranking entities in news applications. We define the task of Time-Aware Entity Retrieval (TAER) which takes into account the evolution of entity relevance over time in a news topic thread. To evaluate the effectiveness of systems performing such task we analyze an extension of the TREC 2004 Novelty corpus [8], annotating relevance at the level of entities [5]. We then evaluate features and ranking models for the TAER task. Dealing with ER in news is particularly interesting as news articles are often focused on entities such as people, companies, countries, etc. It is also a challenging task as, differently from standard ER tasks, there is the time dimension involved. Given a news topic, the decision about which entities should be retrieved or not changes with time. Not all frequently appearing entities should be considered relevant to the topic (e.g., news agencies) and new important entities may appear later in the story (e.g., witness of a murder).

We propose a system which takes into account both information from the current news article as well as from the past relevant articles in order to detect the most important entities in the current news. Our main findings, obtained by analyzing dataset and features, are that sentence novelty is worse than pure sentence relevance as an indicator of entity relevance; entities that become relevant have a high probability of remaining relevant the next article and the entire news thread; the relevant history of an article (e.g. the previous relevant articles) can be exploited as a source of information for TAER.

The paper is organized as follows. Section 2 presents previous related work on entity search. Section 3 defines the task we address comparing it to standard ER and introduces the dataset we created for evaluating time-aware entity search. Section 4 presents an experimental evaluation of extracted features for time-aware entity search. The paper ends with a conclusion section.

2. RELATED WORK

Searching for entities is a common user activity on the Web. There is an increasing effort in the research community in developing entity search techniques and in building evaluation benchmarks. One example is the expert search task evaluated in the context of the TREC Enterprise Track [3], where the goal is to find entities (people) that have relevant expertise about a topic of interest. Language models-based approaches [1] are among the most promising techniques for ranking experts. The INEX Entity Ranking Track is another evaluation initiative where the task is to return

*Work performed while intern at Yahoo! Research.

†This work is partially supported by the EU Large Scale Integrated Project LivingKnowledge (contract no. 231126).

[1]http://news.bbc.co.uk, http://news.google.com, http://news.yahoo.com

CIKM'10, October 26–30, 2010, Toronto, Ontario, Canada.

a list of relevant Wikipedia entities for a given topic using an XML collection [4]. In this context, Vercoustre et al. [9] use Wikipedia categories and link structure together with entity examples to improve ER effectiveness. In the TREC 2009 Entity Track [2] the task of finding related entities given one entity as query (e.g., "Airlines that currently use Boeing 747 planes") was investigated. Compared to previous work on ER we analyze the usefulness of the time dimension for this task. In [5] we introduced the task of time-aware entity retrieval and some features for it. In this paper we describe additionally properties of the dataset, exploit sentence relevance, and study machine learning combinations of features.

3. TIME-AWARE ENTITY RETRIEVAL

Standard Entity Retrieval is defined as follows:

- Entity Retrieval (ER): Given a query and a document collection, retrieve a set of entities appearing in the collection which are relevant to the query.

For example, the ER task was performed in [10] using Wikipedia as a document collection. Consider the following user scenario: a user types a query (or topic) into a news search engine and obtains a list of relevant results, ordered by time. Furthermore, the user subscribes to this query so in the future she will continue to receive the latest news on this query (or topic). We are interested in ER tasks related to this user scenario. Standard ER could be used to show to the user the most interesting entities *for the query*. The temporal dimension is not needed here. However, if the user is observing a current document, we may want to show the most relevant entities of the document for her query (or topic). This prompts the following definition:

- Time-Aware Entity Retrieval (TAER): Given a query and a document relevant to it, and possibly a set of previous related documents (the *history* of the document), retrieve a set of entities that best describe the document.

This is a newly defined task that can be useful, for example, in news verticals for presenting the user more than just a ranked list of documents. In the news context we define the task for most considered entity types: persons, locations, organizations, and products. More formally, we define a "news thread" relevant to a query as the list of relevant documents $D = [d_1 \dots d_n]$. Then, given a document d_i we define its history as the list of relevant documents $H = [d_1 \dots d_{i-1}]$ chronologically ordered pre-dating the document d_i. Given an entity e, we note as $d_{e,1}$ the first document in which the entity occurred in the news thread. Note that such a document is not necessarily the first document in D as entities may appear only in subsequent documents. Additionally, we will note as $d_{e,-1}$ as the last document in H containing e.

3.1 A Dataset for Evaluating ER Over Time

The TREC Novelty Track in 2004 was based on a collection of news articles and a set of topics for evaluating retrieval of novel information over ranked lists of documents for each topic. The systems had to retrieve information (i.e., sentences in this case) relevant to the topic and not yet present in the retrieved results [8].

We selected the 25 'event' topics from the latest TREC Novelty collection (2004). We annotated the documents associated with those topics using state of the art NLP tools [10] in order to extract entities of type person, location, organization, and product. Then, six human judges assessed the relevance of the entities in each document with respect to the topic grading each entity on the 3-points scale: Relevant, Related, Not Relevant[2]. Double assessments on six topics shown an assessors' agreement of 0.5232 (Cohen's Kappa). More information about the data is available in [5].

Table 1: Probabilities of relevance for different entity types with 95% confidence intervals.

$P(r_e \mid t_e = person)$	0.406 [0.391-0.421]
$P(r_e \mid t_e = person, r_s)$	**0.560** [0.533-0.588]
$P(r_e \mid t_e = person, n_s)$	0.496 [0.451-0.541]
$P(r_e \mid t_e = organization)$	0.479 [0.471-0.487]
$P(r_e \mid t_e = organization, r_s)$	**0.631** [0.616-0.646]
$P(r_e \mid t_e = organization, n_s)$	0.587 [0.564-0.612]
$P(r_e \mid t_e = product)$	0.179 [0.164-0.194]
$P(r_e \mid t_e = product, r_s)$	**0.237** [0.210-0.265]
$P(r_e \mid t_e = product, n_s)$	0.189 [0.151-0.228]
$P(r_e \mid t_e = location)$	0.284 [0.271-0.297]
$P(r_e \mid t_e = location, r_s)$	**0.403** [0.379-0.427]
$P(r_e \mid t_e = location, n_s)$	0.397 [0.363-0.432]

3.2 Analysis of the Dataset

The TREC 2004 Novelty collection consists of an average of 31.2 articles per topic distributed over time. After the annotation, each document contains on average 26.5 annotated entities among which 7.6 were judged relevant. On average each topic contains 63.4 entities which have been marked relevant at least once over the topic timeline.

We now investigate the relation between entities, sentence and relevance. Let n_s, r_s indicate that a sentence s is novel or relevant respectively. Let t_e indicate the type of entity e, and let us denote by r_e the fact that e is relevant, and $\overline{r}_e$ otherwise. On average, a sentence contains 1.46 entities, a relevant sentence contains 1.88 entities, and a novel sentence contains 1.92 entities which indicates the presence of more information. The unconditional probability of a relevant entity in a sentence $P(r_e)$ is 0.411 (we first sample a sentence and then an entity in that sentence). The probability of finding a relevant entity in a relevant sentence $P(r_e|r_s)$ is 0.547 with a 95% bootstrap confidence interval of $[0.534 - 0.559]$, well above $P(r_e)$. The probability of a relevant entity in a novel sentence $P(r_e|n_s)$ is 0.510 $[0.491 - 0.531]$ which is below the probability in a relevant sentence.

This gives the following high level picture. Relevant sentences contain slightly more entities than non-relevant ones. Novel sentences contain slightly more entities than relevant (but not-novel) ones; however, entities in novel sentences are more likely to be irrelevant than in not-novel sentences.

In Table 1 we look at relevance probabilities per entity type (e.g., the probability of person entity being relevant would be noted $P(r_e|t_e = person)$). We see that sentence novelty is less important than sentence relevance *regardless of the entity type*. Organization entities are more likely in a relevant sentences than the rest.

As compared to a classic document collection, in a news corpus the time dimension is an additional available feature. How useful is the information from past news articles? The probability of an entity being relevant in a document given

[2]The evaluation collection we have created is available for download at: `http://www.l3s.de/~demartini/deert/`

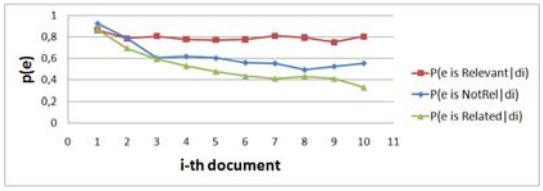

Figure 1: Probabilities of entity relevance given its relevance in the i-th document of its history (i.e., past related articles).

that it was relevant the first time it appeared ($d_{e,1}$) is 0.893 [0.881 − 0.905] which shows how in most cases an entity which is relevant at the beginning of its appearance stays relevant for the rest of the news thread. It is also important to observe just the previous document where the entity appeared. The probability of an entity being relevant in a document given that it was relevant the previous time it appeared is 0.701 [0.677 − 0.726]. Conversely, the probability of a relevant entity changing relevance status form one story to the next is 0.3. Another characterization of this is the probability of an entity being relevant in a document given that it was relevant in the i-th document of its history. This is shown in Figure 1 for relevant, related and not-relevant entities. We can see that relevant entities are the most stable over time while related entities tend to change relevance status over time (either to relevant or to not-relevant).

4. MODELS FOR TAER

4.1 Features

For performing the TAER task we exploit features defined in [5]. In detail, we consider the following features: the frequency of an entity e in a document d, noted $F(e,d)$. We will use this feature as our baseline. Then, we consider the average or the sum of BM25 score of the sentences where e appears in d (noted $AvgBM25s(e,d)$ and $SumBM25s(e,d)$ respectively)[3]. We also consider the number of times an entity e appears as subject of a sentence in the document d, noted $F_{subj}(e,d)$; the length of the first sentence where e appears in document d, noted $FirstSenLen(e,d)$; and the position of the first sentence where e appears in d (e.g, the 4th sentence in the document), noted $FirstSenPos(e,d)$.

As the dataset analysis shown that past related articles may contain important information about entity relevance, we also consider a number of features that take into consideration the document history H: the frequency (i.e., the number of times it appears) of the entity e in the history H, noted $F(e,H)$; the document frequency of e in H, noted $DF(e,H)$; the frequency of entity e in the first document where the entity appeared, noted $F(e,d_{e,1})$; the frequency of entity e in the previous document where the entity appeared, noted $F(e,d_{e,-1})$; and the number of other entities with which the entity co-occurred in a sentence in the set of past documents H, noted $CoOcc(e,H)$.

4.2 Experimental Evaluation

We compare the effectiveness of different features and feature combinations using several performance metrics. In order to evaluate the complete entity ranking produced by the proposed features, we compute Mean Average Precision (MAP). For completeness, as we aim at showing the user few entities, we check for early precision as well. We report values for Precision@3 (P@3), Precision@5 (P@5), and we test for statistical significance using the t-test. To compute the measures we consider related entities as non-relevant. Many of the features we use are based on entity frequency, hence entity scores in the ranking will have many ties. For this reason, the evaluation measures we have computed are aware of ties, that is, they consider the average value of the measure for all possible combinations of tied scores [7].

[3]We computed the BM25 scores of sentences with respect to a disjunctive query consisting of all the terms in the topic title using $b = 0.75$, $k_1 = 1.2$.

Evaluation of Single Features.

We can have an initial analysis of such features by checking how entity relevance probability changes with the features value. Figure 2 shows the probability of an entity being relevant given different values of the features described above. We see that all are correlated with relevance over their entire domain.

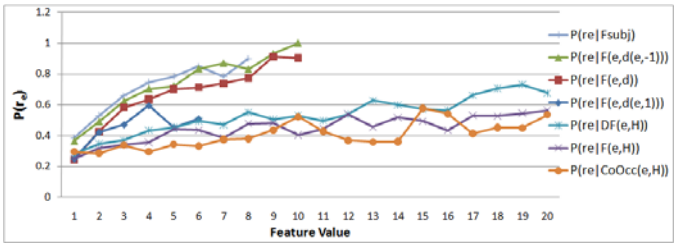

Figure 2: Probability of an entity being relevant given different feature values for several features.

Table 2 shows effectiveness values obtained when ranking entities in a document according to defined features, where no local feature performs better than the simple frequency of entities in the document. For comparison, a feature that assigns the same score to each entity would obtain a MAP value of 0.42 with a ties-aware measure. The second best local features is $SumBM25s$ (0.52 MAP) which takes into consideration relevance of sentences where the entity appears. On the other hand, the features looking at the first sentence where the entity appears in the news article (FirstSenLen, FirstSenPos) do not perform well (0.45 and 0.43 MAP respectively). In order to exploit the position of the first sentence where an entity appears we need to deal with the problem of headers in news articles (e.g., news agency codes): as articles have different header lengths, it is not easy to detect the beginning of the article body.

In general, history features perform better than local features and the highest performance is obtained by ranking entities according to its frequency in the past documents ($F(e,H)$). All history features but $F(e,d_{e,1})$ significantly improved over the baseline in terms of MAP. In terms of early precision (P@5) only $F(e,H)$ and the similar feature $DF(e,H)$ improve over the baseline. Moreover, features using the entire history H are performing better than features looking at single documents in the past.

It is also interesting to note that, when identifying relevant entities for a document, the frequency of the entity in the previous document in the story $F(e,d_{e,-1})$ is a better evidence than the frequency in the current document. This may be an indication of how people read news: some entities

become relevant to readers after repeated occurrences. If an entity appears in the current and previous documents it is more likely to be relevant.

We additionally weighted the scores obtained from different documents in H with both the document length and BM25 score of the document with respect to the query. This approach did not improve the effectiveness of the original features without per-document weighting. Given these results we conclude that the evidence from the past is very important for ranking entities appearing in a document. Thus, we expect effectiveness of methods that exploit the past to improve as the size of H grows. That is, the more history is available the better we can rank entities in the current news.

Table 2: Effectiveness of individual features and of features when combined together using ML. Bold values indicate the best performing runs. * (**) indicates statistical significance w.r.t. F(e,d) and †(††) w.r.t. F(e,H) with paired t-test p<0.05(0.01).

Feature	P@3	P@5	MAP
All Ties	.34	.34	.42
Individual Features (Local and History)			
$F(e,d)$	**.65**	**.56**	**.60**
$FirstSenLen$	.37	.36	.45
$FirstSenPos$	.31	.31	.43
F_{subj}	.49	.44	.50
$AvgBM25s$	.27	.30	.41
$SumBM25s$	.50	.44	.52
$F(e,d_{e,1})$	.58	.53	.56
$F(e,d_{e,-1})$	.64	.56	.62*
$DF(e,H)$	.63	.57*	.65**
$F(e,H)$	**.66**	**.59****	**.66****
$CoOcc(e,H)$	.62	.57	.65**
Features combined with Logistic Regression			
Local	.65	.58*	.63**
History	.65	.60**	.66**
All	**.70**††**	**.63**††**	**.69**††**

The y-axis of Figure 3 plots the average MAP for all the documents with history size $|H|$ using the feature $F(e,H)$. For $|H| < 20$ the effectiveness of $F(e,H)$ increases together with $|H|$ up to values of 0.7. Results for higher values of $|H|$ show no clear trend due to the fact that there are just a few datapoints.

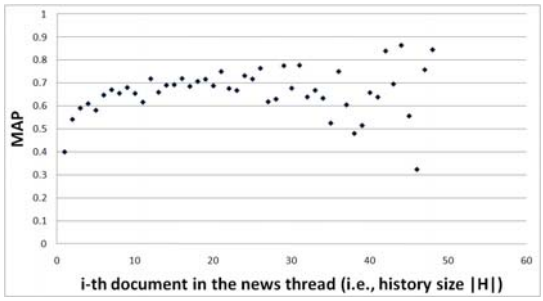

Figure 3: Mean Average Precision values for documents having a certain history size.

Using Machine Learning for combining features.

So far we have presented different features for ranking entities that appear in a document. Combining them in an appropriate manner yields a better ranking of entities; however, because the distribution of relevance probability is different among features, we need a way for combining them.

In order to combine two or more features together we used Machine Learning (ML) techniques. We performed 2-fold cross validation training a multinomial logistic regression model with a ridge estimator [6] with default parameters for ranking entities in each document.

Table 2 presents a combination of every local and history feature. The combination of all local features performs better then the baseline and then the single local features. When all the features are combined (local+history) we obtain the best effectiveness. Such improvements are anyway negligible if compared with the best 2 features combination, that is, $F(e,d)$ and $F(e,H)$ obtaining a MAP of 0.68 [5]. Therefore, we can see how these two simple features perform very well and that it is difficult to improve over such approach.

5. CONCLUSIONS

In this paper we have addressed the problem of entity search and ranking over time. For this purpose, we defined an original entity search task and further analyzed a time-stamped test collection for evaluating it. One of the conclusions is that determining the relevance of a sentence is very important to determine the relevance of an entity; more so than determining sentence novelty. In fact novel sentences introduce more entities than non-novel sentences, but many of these are not relevant.

We have evaluated features both from the current document and from previous ones in the document's history in order to find relevant entities in a given document. We have experimentally shown that past frequency of entities is the most important of the features explored so far, more important than entity frequency in the current document. We have tested several combinations of proposed features obtaining an overall statistically significant improvement of 15% in terms of MAP over the baseline that considers the frequency of entities in the document.

6. REFERENCES

[1] K. Balog, L. Azzopardi, and M. de Rijke. Formal models for expert finding in enterprise corpora. In *SIGIR*, 2006.

[2] K. Balog, A. de Vries, P. Serdyukov, P. Thomas, and T. Westerveld. Overview of the TREC 2009 Entity Track. In *TREC 2009*. NIST. Special Publication.

[3] K. Balog, I. Soboroff, P. Thomas, N. Craswell, A. P. de Vries, and P. Bailey. Overview of the TREC 2008 enterprise track. In *TREC 2008*. NIST, 2009.

[4] G. Demartini, T. Iofciu, and A. P. de Vries. Overview of the INEX 2009 Entity Ranking Track. In *INEX*, pages 254–264, 2009.

[5] G. Demartini, M. M. S. Missen, R. Blanco, and H. Zaragoza. Entity summarization of news articles. In *SIGIR '10*, pages 795–796. ACM, 2010.

[6] S. Le Cessie and J. Van Houwelingen. Ridge estimators in logistic regression. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, (1), 1992.

[7] F. McSherry and M. Najork. Computing information retrieval performance measures efficiently in the presence of tied scores. In *ECIR*, 2008.

[8] I. Soboroff. Overview of the TREC 2004 Novelty Track. In *TREC*. NIST, 2004.

[9] A.-M. Vercoustre, J. A. Thom, and J. Pehcevski. Entity ranking in Wikipedia. In *SAC '08*, USA, 2008. ACM.

[10] H. Zaragoza, H. Rode, P. Mika, J. Atserias, M. Ciaramita, and G. Attardi. Ranking very many typed entities on Wikipedia. In *CIKM '07*, USA, 2007. ACM.