UMass at TREC 2013 Knowledge Base Acceleration Track: Bi-directional Entity Linking and Time-aware Evaluation

Laura Dietz
University of Massachusetts, Amherst
dietz@cs.umass.edu

Jeffrey Dalton
University of Massachusetts, Amherst
jdalton@cs.umass.edu

1 Introduction

This notebook details the participation of the University of Massachusetts Amherst in the Cumulative Citation Recommendation task (CCR) of the TREC 2013 Knowledge Base Acceleration Track. Our interest in TREC KBA is motived by our research on entity-based query expansion. Query expansion is a information retrieval technique for improving recall by augmenting the original query terms with other terms that are likely to indicate relevant documents. Such expansion terms can be inferred with pseudo-relevance feedback techniques (Lavrenko and Croft, 2001). The resulting retrieval model can be interpreted as a weighted mixture model including the original retrieval model and retrieval models for each expansion term.

Instead of expanding the query with terms, our research is on expanding the query with relevant entities from a knowledge base. Such entities are very rich in structure, including name variants, related entities and associated text. An essential component of our entity-based query expansion is to derive a retrieval model for a given knowledge base entity, which can be incorporated into the weighted mixture model. We study the effectiveness of different entity-based retrieval models within the TREC KBA Cumulative Citation Recommendation task.

However, we do not address the novelty aspects of the task, and therefore do not distinguish between 'vital' and 'useful' documents. This year we only evaluate memoryless methods, i.e., the prediction is not influenced by predictions on previous time intervals. We segment the stream into week-long intervals which are filtered independently.

2 Structured Entity Data

First we study different ways to derive a retrieval model from an entity in a knowledge base such as Wikipedia. Our methods assume access to different kinds of structured information about the entity: 1) a canonical name such as the Wikipedia title; 2) a set of alternative names with associated confidences; 3) links or relations to other entities; 4) optional free text introducing the entity.

We preprocessed a 2012 Wikipedia Wex dump to make all four kinds of data available easily (more information available in (Dalton and Dietz, 2013b)). Although similar information can be gathered for twitter entities as well, we did not have a twitter corpus available. Instead we vary the method only for Wikipedia entities, where all twitter entities are predicted with the "SDM" method. The evaluation in this paper only considers Wikipedia entities.

3 Document Retrieval Methods

We explore readily available retrieval models and study which kinds of structured entity information provide the most value.

3.1 Traditional IR Models: SDM and RM3

The simplest approach is to use the canonical name as an information retrieval query. We use the sequential dependence retrieval model which scores documents by frequency of unigrams, bigrams and windowed skip-bigrams of the query string, taking document length and corpus wide term statistics into account. Given the query string q, the retrieval score, $\log \mathcal{L}$, is computed according to Equation 1, which is also referred to via the query operator #sdm. In contrast, a query consisting of unigram terms only is represented by $\mathcal{L}_{\text{unigram}}$, which is also referred to as query likelihood.

$$\log \mathcal{L}_{\text{SDM}}(d|q) \quad \stackrel{\text{rank}}{=} \sum_i \Big(\begin{array}{l} \lambda_T \log \mathcal{L}_{\text{unigram}}(d|q_i) \\ +\lambda_B \log \mathcal{L}_{\text{bigram}}(d|q_i, q_{i+1}) \\ +\lambda_W \log \mathcal{L}_{\text{window}}(d|q_i, q_{i+1}) \Big) \end{array} \tag{1}$$

The unigram model is given in Equation 2 where tf referring to the term frequency of the query term q_i in the

Method	Retrieval Model
sdm	#sdm(canonical name)
rm	w_q#sdm(canonical name)+$(1-w_q)\sum_{\text{pseudoterm}} w_{\text{pseudoterm}} \cdot \log \mathcal{L}_{\text{unigram}}(d\|\text{pseudo term})$
rn	w_q#sdm(canonical name)+$(1-w_q)\sum_{\text{name}} w_{\text{name}}$·#sdm(name)
rt	w_q#sdm(canonical name)+$(1-w_q)\sum_{\text{articleterm}} w_{\text{articleterm}} \cdot \log \mathcal{L}_{\text{unigram}}(d\|\text{article term})$
rtn	w_q#sdm(canonical name)+$\frac{1-w_q}{2}\sum_{\text{name}} w_{\text{name}}$·#sdm(name) + $\frac{1-w_q}{2}\sum_{\text{articleterm}} w_{\text{articleterm}} \cdot \log \mathcal{L}_{\text{unigram}}(d\|\text{article term})$

Table 1: KB-based retrieval models.

document d. We apply Dirichlet smoothing with parameter μ based on collection statistics denoted by C. The bigram and windowed skip-bigram model follow analogously by exchanging the term frequency with bigram and window-bigram frequencies.

$$\log \mathcal{L}_{\text{unigram}}(d|q_i) \stackrel{\text{rank}}{=} \log \frac{tf_{q_i,d} + \mu \frac{tf_{q_i,C}}{|C|}}{|d| + \mu} \tag{2}$$

All our IR methods associate a confidence with each document which is proportional to the retrieval score. The method "sdm" issues a #sdm query with the cannonical entity name (e.g. the Wikipedia title of the query entity).

The method "rm" refers to a sequential dependence model wich is expanded with pseudo-relevance feedback. This refers to a two-pass method, where first a sequential dependence query is issued to retrieve a few top ranked documents from the stream corpus. Assuming that the retrieval score indeed captures the degree with which the document is relevant for the query, a distribution over terms is extracted as follows: A language model is build from each retrieved document to be proportional to the term frequency. Using multinomial mixture weights proportional to the exponentiated retrieval score $\mathcal{L}_{\text{SDM}}(d|q)$, the language models are combined into a mixture model (cf. Equation 3, where normalization constant Z is to ensure the components sum to 1).

$$w_t = \frac{1}{Z} \sum_d \mathcal{L}_{\text{SDM}}(d|q) \frac{tf_{t,d}}{|d|} \tag{3}$$

The k most probable terms under this distribution are used to expand the sequential dependence query using the weight w_t (referred to as $w_{\text{pseudoterm}}$ in Table 1).

3.2 KB-based Retrieval Models: Names and Text

In the following, we extend the "sdm" methods by incorporating further names and text from the knowledge base.

The set of alternative names is exploited in the method "rn". We extract alternative names from structured data available for Wikipedia entities, including name variants from redirect pages, Freebase alternative names, and anchor text of links within Wikipedia. These names are combined into a mixture of sequential dependence retrieval models, weighted by the confidence of respective names.

We assign a disambiguation confidence for each name from anchor text. For each possible name of this entity, we derive the score as the fraction of hyper links with this name as anchor text that refer to the query entity. We also apply this scheme to Wikipedia redirects and Freebase names (which we treat as twice as trustworthy as anchor text) and compute a combined model of disambiguating names for the query entity.

The retrieval model incorporates the names with the highest disambiguation score as a mixture model of sequential dependence models for each name. The disambiguation scores are used as weights w_{name} for each mixture component. The name model is combined with a sequential dependence model on the canonical name as in the "sdm" method. Details are given in Table 1.

We further explore the use of terms extracted from the text that is associated with the entity. For method "rt", we use the text of the Wikipedia article to build a term model, after removing stopwords and normalizing punctuation. The top terms are used in a mixture model of unigram language models with the term probabilities as weights $w_{\text{articleterm}}$. We notice that the text also includes mentions of the query entity under different names as well as mentions of strongly related entities.

However, we additionally explore the use of extending the canonical name with both disambiguating names and frequent terms in the method "rtn".

3.3 Knowledge Sketch Approach

As motivated in the introduction, our research goal is to retrieve relevant entities, documents and relations for a given query. As the approach is currently under submission, we omit details here, but refer the interested reader to a preliminary workshop writeup (Dalton and Dietz, 2013a).

We apply the knowledge sketch approach in method "skq" using the canonical name as a query. The method will retrieve relevant entities, which are used to expand

the original query with named of relevant (neighbor) entities to retrieve documents.

3.4 Converting Retrieval Scores to Confidences

We view stream filtering as a continuous task, where a user checks the pool of predicted documents in regular time intervals, for instance one a week. At every check point the user would see a ranking of the most confident top 1000 ranks and with the option to stop inspecting lower ranks, e.g. when precision sinks below a threshold. We simulate this scenario by scoring documents in a stream fashion and assign confidences that would represent the i'th rank.

We learn this mapping from document score to confidence rank by generating a document ranking on week-long subcorpora of the training period. In particular, we choose the weeks 2011-49 and 2012-07 (given in calendar week of the year) and generated rankings across all entities. We take the maximum of the score obtained on rank 1 as an equivalent of confidence 1000 and the minimum score obtained on rank 1000 to be equivalent to confidence 1. We project retrieval scores linearly onto this confidence interval. The stream is filtered by computing the retrieval model score under each document and project it onto the confidences.

Since we expect the different retrieval models to have drastically varying scores (which are rank equivalent to unnormalized log-probabilities) we learn a different score-confidence mapping for each method. As a result, the confidence cutoffs are not comparable across our methods.

We want to point out that no training judgments are used in our process. The heuristic only requires two weeks of the training corpus to identify the range of scores.

3.5 Indexing

Our retrieval models are memory-less, they do not learn over time and predictions from the previous time interval do not affect the predictions of the next. We parallelize the document filtering by creating several indexes of week-long segments of the stream. We use galago 3.4 for indexing with the indexing parameters listed in Figure 7.

4 Linking back to Entities

We anticipate that the information retrieval methods may have problems distinguishing mentions of the query entity from entities with similar names. We explore the utility of our entity linking tool[1] to refine the document scores produced by the SDM method. Due to time constraints for the submission deadline, we simulate the method on the two top scoring documents per week.

4.1 Entity Linking

Our entity linking method first detects named entities in the retrieved document (using Factorie's NLP Pipeline[2]). For each mention we issue a query against a search index of Wikipedia articles, which includes structured information such as linked articles and anchor text. The query is a combination of the mention and the name variants from the coreference resolution. For each mention, the top 50 Wikipedia entities are taken as candidates to be re-ranked with supervised learning-to-rank method (using a boosted decision tree (Friedman, 2001)). Features for the supervision include different kinds of simlarity between mention string and Wikipedia title, surrounding named entities to Wikipedia neighbors, as well as terms from the context and the Wikipedia article. Optionally, NIL classification is applied. The method is detailed in (Dalton and Dietz, 2013b), with the retrieval method based on query and name variants ("QV"), features for the learning-to-rank method and NIL classification. For every mention in the document we keep the 50 retrieved candidate entities around with supervised re-ranking score and NIL prediction.

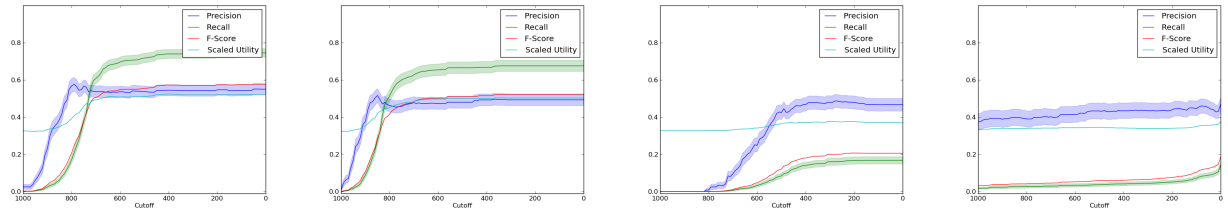
4.2 Deriving Document Score

Next we inspect all entity links in the document for links towards the query entity. We evaluate the following heuristics for deriving a score for the document:

- T2ELMax / "link": Maximum re-ranking score of the query entity for any mention, independent of the rank (inspecting all 50 candidates).
- T2ELMax_1 / "link NIL": Maximum re-ranking score of the query entity for any mention, independent of the rank, as long as it is not classified as NIL.
- T2ELMax_TO / "link Top": Maximum re-ranking score of the query entity for any mention, only if the query entity is the top ranked entity.
- T2ELMax_TO_1 / "link Top NIL": Maximum re-ranking score of the query entity for any mention, only if the query entity is the top ranked entity and not classified as NIL.
- t2LinkProb / "link LM": Probability under a multinomial distribution over linked Wikipedia entities; Distribution is build from top ranked (non-NIL) links per mention in the fashion of a language model.

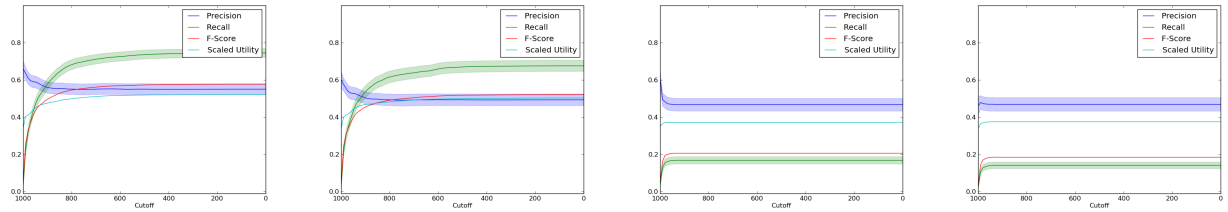
[1] code available at http://ciir.cs.umass.edu/~jdalton/kbbridge/

[2] http://factorie.cs.umass.edu/



(a) SDM (b) RN (c) Entity Link (max score) (d) Entity Link LM

Figure 1: P/R/F over confidence cutoffs.



(a) SDM (b) RN (c) Entity Link (max score) (d) Entity Link LM

Figure 2: P/R/F over rank cutoffs.

Only the last heuristic incorporates the frequency with which the target entity is mentioned in the document.

The resulting log-scores range in [-15, +15] and are inverted and linearly projected onto the [1, 1000] confidence interval.

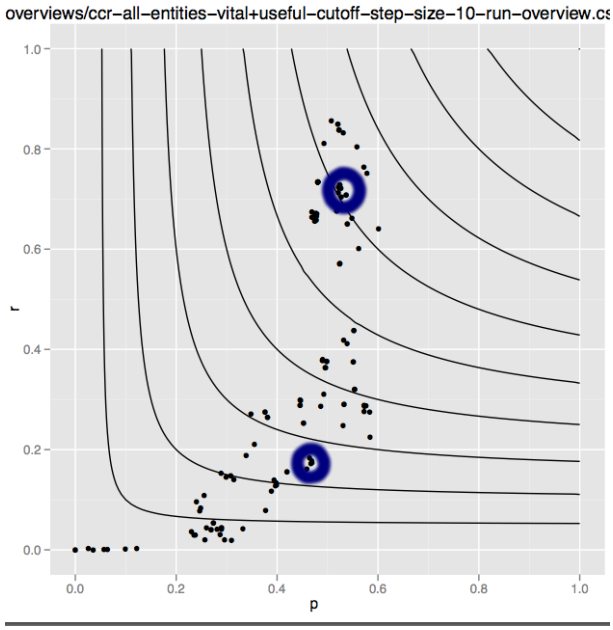


Figure 3: UMass_CIIR runs in comparison. Blue circle on top marks the "sdm" method; the blue circle at the bottom marks the "entity link LM" method.

5 Results

We participate in the CCR task, with the goal of predicting documents in the "useful" class.

Figure 3 (top) marks our best IR run the "sdm" method, indicating that our methods perform competitively. The bottom circle marks all our entity linking runs, which (as expected) have a much lower recall, since entity links are only annotated for the the top two documents per week from the "sdm" method.

As our submission did not focus on twitter entities, nor the novelty aspects, we analyze our results with the scoring script options 'Wikipedia-only', 'vital+useful', and 'require-positives'. Figures 1a-d display the Precision/Recall/F1/SU over cutoff ranks with standard error tubes. The presented runs comprise the method "sdm", expanding with entity names ("rn"), entity linking with max score ("link"), and the entity linking language model approach ("link LM"). For comparison, 4a displays the precision of all our runs in one plot. The "sdm" method reachest a maximum of 0.57, with a decent recall of 0.75. Expansion with names and terms from Wikipedia yields better precision on high cutoffs. The entity link language model starts with a high precision that increases slightly.

We suspect that our conversion from retrieval score to confidence is flawed. To distinguish error sources, we analyze our methods via the ranking induced by confidence values. As the plots over confidence cutoffs ignore documents that were not assessed by the annotators, we omit

them from the ranking as well. Figures 2 and 4b present plots of precision at rank k, where rank 1 corresponds to cutoff 999, rank 2 to cutoff 998, etc.

We are glad to see that across all methods precision decreases over the ranks, indicating that on average, useful documents are located on higher ranks than not useful documents. The methods "sdm" and "skq" perform the best. We are surprised that the "sdm" method—which was originally our base line—outperformed all methods in terms of precision and also achieved a stunning recall of 0.72. The KB-based retrieval models perform worse in terms of precision, there the combination of both terms and names is slightly worse than expansion with either source. We find that the top 100 of all IR models contain about one third of documents that were not assessed by annotators. This bears the potential of changing the ranking among these methods.

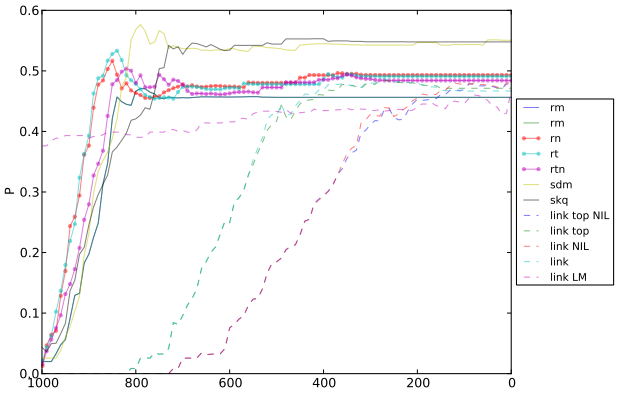
The entity linking methods were intended to increase the precision for the "sdm" method, of which two documents per week were considered. It seems that this is not the case, as the precision of 0.55 in the top 10 levels out quickly to 0.46-0.48. Error analysis revealed that although only 10% of the documents in top 100 were not assessed by annotators, we only predicted on average 20 documents per entity—a number that is way too conservative to be useful in practice. Furthermore, for several entities no documents were predicted, which attributed scores of zero in the plotted macro-average; the corresponding micro-average is about 0.62. This explains why the entity linking heuristics that only consider query entities on top1 and/or if not NIL perform worse than their less restrictive counterparts.

The relevance model on pseudo-relevance feedback (method "rm") performs with worst precision and mediocre recall.

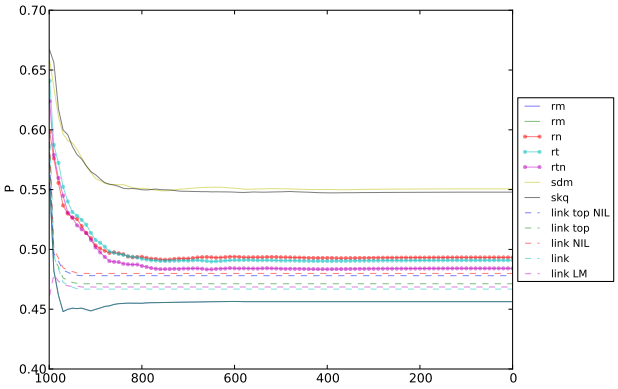
5.1 Time-aware Analysis

In our paper at the TAIA SIGIR 2013 workshop (Dietz et al., 2013) we suggested the use of time-aware evaluation paradigm as an alternative[3]. The difference is visualized in Figure 5. Merging all weekly rankings into one overall ranking by confidence score indicates the "sdm" model as a clear winner and places name alias expansion ("rn") on the second rank.

In contrast, evaluating performance on a weekly basis demonstrates that name alias expansion is consistently worse than "sdm", while the entity linking methods outperform "sdm" on many weeks and are especially strong around ETR day 100. Macro-averaging across weeks confirms the similar performance of "sdm" (MAP 0.031)



(a) Precision over confidence cutoffs.



(b) Precision over rank cutoffs.

Figure 4: Precision of all submitted runs.

and all entity linking methods (MAP 0.032) where name alias expansion "rn" (MAP 0.025) is clearly behind.

This discrepancy is in analogy to discrepancy between micro- and macro-averages: It is more difficult (and more useful) to consistently predict good results across all weeks if they vary in difficulty.

5.2 Treatment of Unjudged Documents

We have noticed a rather large discrepancy between the findings of the official TREC KBA scorer and our time-aware evaluation method and rather low MAP scores in the time-aware analysis. It turns out that the main difference is in the way documents with missing relevant/non-relevant judgments are treated. While our time-aware evaluation followed the general practice in information retrieval to count unjudged documents as true negatives, the official KBA scorer removed any unjudged documents before the evaluation.

Figure 6 depicts this effect on methods "sdm" and "link LM" with respect to Precision@10. We see that this inverts the finding. In fact, most of the documents in the high ranks (across both entities and weeks) are unjudged, where the many judged documents are found on ranks

[3]Updated code for KBA-2013 format is available at `github.com/laura-dietz/kba-y2-streameval/` with more plots and information at `ciir.cs.umass.edu/~dietz/streameval/`

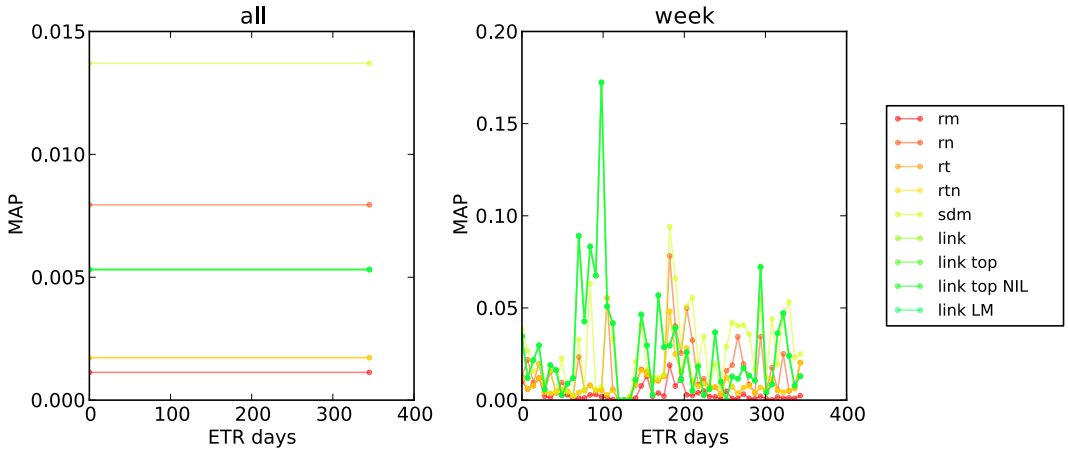
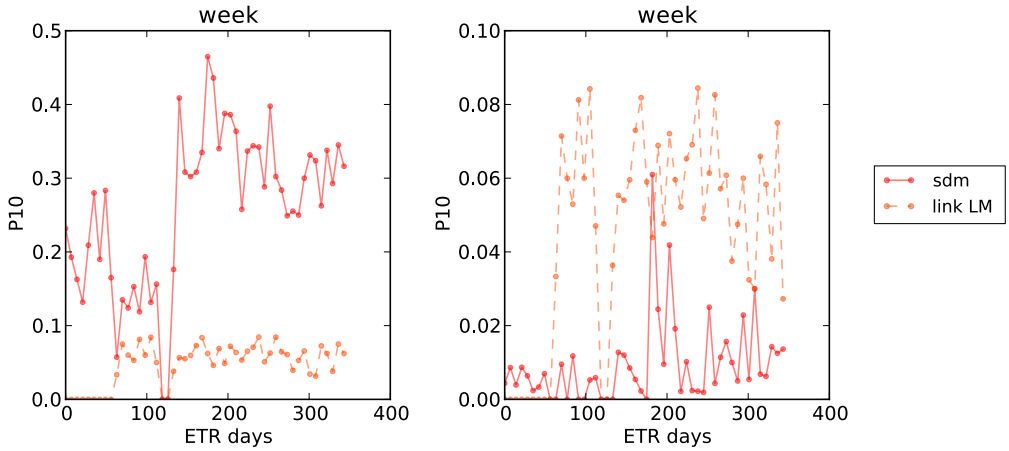


Figure 5: Overall MAP score (left) versus average over weekly scores (right).



(a) Unjudged ignored (official KBA scorer).

(b) Unjudged as negative (our evaluation).

Figure 6: Treatment of documents with missing judgments.

beyond 10.

The sparsity of judgments in our high ranks comes from how documents are selected for assessment by the judges. Unlike other TREC tracks, which generate pools from contributed submissions, the KBA pools are selected by a system developed by the track organizers. The official scores represent which systems would improve the pool generating method.

6 Conclusion

This paper presents the submission of the Center for Intelligent Information Retrieval at the University of Massachusetts to TREC KBA. Based on the idea of casting entity tracking as a retrieval problem we presented several methods that leverage the rich structure of knowledge base entities using IR, NLP, and supervised reranking. Information retrieval is used both to retrieve relevant documents and for entity linking—we refer to the combination as bi-directional entity linking. Even without entity-specific training, the retrieval methods give rise to a reasonable document filter. We were surprised that the sequential dependence model ("sdm"), originally intended as a baseline performs provides the highest precision, recall, and runtime performance.

However, we suspect that the "sdm" method yielded many non-relevant documents in the top ranks, which were effectively filtered out by the pool-generating method and are therefore not reflected in the evaluation score. In contrast, the entity linking method was intended to be a high-precision method selecting only up

to two documents per week and entity. Analysis in Figure 6b confirms that the entity linking method retains four times more relevant documents in the top 10 than the "sdm" method. Furthermore, our time-aware evaluation paradigm shows that sdm and entity linking are retrieving weekly rankings of equal mean-average precision.

Acknowledgment

This work was supported in part by the Center for Intelligent Information Retrieval, in part by IBM subcontract #4913003298 under DARPA prime contract #HR001-12-C-0015, and in part by NSF grant #IIS-0910884. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

References

Jeffrey Dalton and Laura Dietz. 2013a. Constructing query-specific knowledge bases. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 55–60. ACM.

Jeffrey Dalton and Laura Dietz. 2013b. A neighborhood relevance model for entity linking. In *Proceedings of the 10th Conference on Open Research Areas in Information Retrieval*, pages 149–156.

Laura Dietz, Jeffrey Dalton, and Krisztian Balog. 2013. Time-aware evaluation of cumulative citation recommendation systems. In *Proceeding of the Workshop on Time-aware Information Access*, pages 10–13.

Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, pages 1189–1232.

Victor Lavrenko and W. Bruce Croft. 2001. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '01, pages 120–127, New York, NY, USA. ACM.

```
{
    "filetype" : "xz" ,
    "parser" : {
      "externalParsers" : [
      {
        "filetype" : "xz",
        "class" : "org.lemurproject.galago.contrib.
             parse.TrecKBA2013Parser"
      }
     ]
    },
    "tokenizer" : {
        "formats" : {
         "kbadate" : "long",
         "kbastreamticks" : "long",
         "kbastreamtimestamp" : "string"
      },
        "fields" : [
         "title",
         "kbadate",
         "kbastreamticks",
         "kbastreamtimestamp",
         "kbatype"
      ]
    },
    "fieldIndexParameters" : {
      "stemmedPostings" : false
   },
    "stemmedPostings" : false
}
```

Figure 7: Index configuration parameters for Galago 3.4.