Ranking Related Entities: Components and Analyses (Abstract) [*]

Marc Bron
m.m.bron@uva.nl

Krisztian Balog
k.balog@uva.nl

Maarten de Rijke
derijke@uva.nl

ISLA, University of Amsterdam
Science Park 904, 1098 XH Amsterdam

ABSTRACT

Related entity finding is the task of returning a ranked list of homepages of relevant entities of a specified type that need to engage in a given relationship with a given source entity. We propose a framework for addressing this task and perform a detailed analysis of four core components; co-occurrence models, type filtering, context modeling, and homepage finding. Results show that pure co-occurrence is useful to select initial candidates, that type filtering is an instrument for tuning towards either recall or precision, and that context models successfully promote entities engaged in the right relation with the source entity. Our method achieves very high recall scores on the end-to-end task and is able to incorporate additional heuristics that lead to state-of-the-art performance.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.3 Information Search and Retrieval

General Terms

Algorithms, Experimentation, Measurement, Performance

Keywords

Entity search, Language modeling, Wikipedia

1. INTRODUCTION

Over the past decade, increasing attention has been devoted to retrieval technology aimed at identifying entities relevant to an information need. The TREC 2009 Entity track introduced the *related entity finding* (REF) task: given a source entity, a relation and a target type, identify homepages of target entities that enjoy the specified relation with the source entity and that satisfy the target type constraint [1]. E.g., for a source entity ("Michael Schumacher"), a relation ("His teammates while he was racing in Formula 1") and a target type ("people") return entities such as "Eddie Irvine" and "Felipe Massa." To address the REF task we consider an entity finding system architecture as shown in Fig. 1. The first component is a co-occurrence-based model that selects candidate entities. While a co-occurrence-based model can be effective in identifying the potential set of related entities, it fails to rank them effectively. Our failure analysis reveals two types of error that affect precision: (1) entities of the wrong type pollute the ranking and (2) entities are retrieved that are associated with the source entity without engaging in the right relation with it. To address (1), we add type filtering based on category information in Wikipedia. To correct for (2), we complement the pipeline with contextual information, represented as statistical language models derived from documents in which the source and target entities co-occur.

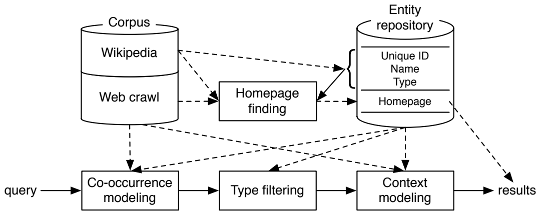

Figure 1: Components of our REF system.

2. APPROACH

The goal of the REF task is to return a ranked list of relevant entities e for a query consisting of a source entity (E), target type (T) and a relation (R) [1]. We formalize REF as the task of estimating the probability $P(e|E,T,R)$. As this probability is difficult to estimate directly we apply Bayes' Theorem and rewrite $P(e|E,T,R)$. After dropping the denominator as it does not influence the ranking of entities, we derive the following ranking formula:

$$\begin{aligned} & P(E,T,R|e) \cdot P(e) \\ & \quad \propto P(E,R|e) \cdot P(T|e) \cdot P(e) \qquad (1) \\ & \quad = P(E,R,e) \cdot P(T|e) = P(R|E,e) \cdot P(E,e) \cdot P(T|e) \\ & \quad \overset{\text{rank}}{=} P(R|E,e) \cdot P(e|E) \cdot P(T|e) \end{aligned}$$

In (1) we assume that type T is independent of source entity E and relation R. We rewrite $P(E,R|e)$ to $P(R|E,e)$ so that it expresses the probability that R is generated by two (co-occurring) entities (e and E). Finally, we rewrite $P(E,e)$ to $P(e|E) \cdot P(E)$ and drop $P(E)$ as it is assumed uniform. We are left with the following components: (i) pure co-occurrence model ($P(e|E)$), (ii) type filtering ($P(T|e)$) and (iii) contextual information ($P(R|E,e)$).

Co-Occurrence Modeling. The pure co-occurrence component ($P(e|E)$) expresses the association between entities based on occurrences in documents, independent of context (i.e., document content). To express the strength of co-occurrence between e and

[*] The full version of this paper appeared in *CIKM 2010*.

E we use a function $\mathrm{cooc}(e, E)$ and estimate $P(e|E)$ as follows:

$$P(e|E) = \frac{\mathrm{cooc}(e, E)}{\sum_{e'} \mathrm{cooc}(e', E)}.$$

We consider two settings of $\mathrm{cooc}(e, E)$: (i) as maximum likelihood estimate and (ii) χ^2 hypothesis test [3].

Type Filtering. In order to perform type filtering we exploit category information available in Wikipedia. We map each of the target types ($T \in \{PER, ORG, PROD\}$) to a set of Wikipedia categories ($\mathrm{cat}(T)$) and create a similar mapping from entities to categories ($\mathrm{cat}(e)$). The former is created manually, while the latter is granted to us in the form of page-category assignments in Wikipedia. Note that we only consider entities that have a Wikipedia page. With these two mappings we estimate $P(T|e)$ as follows:

$$P(T|e) = \begin{cases} 1 & \text{if } \mathrm{cat}(e) \cap \mathrm{cat}^{L_n}(T) \neq \emptyset \\ 0 & \text{otherwise.} \end{cases}$$

We expand the set of categories assigned to each target entity type T, hence write $\mathrm{cat}^{L_n}(T)$, where L_n is the chosen level of expansion and a parameter to be determined empirically.

Adding Context. The $P(R|E, e)$ component is the probability that a relation is generated from ("observable in") the context of a source and candidate entity pair. We represent the relation between a pair of entities by a co-occurrence language model (θ_{Ee}), a distribution over terms taken from documents in which the source and candidate entities co-occur. By assuming independence between the terms in the relation R we arrive at the following estimation:

$$P(R|E, e) = P(R|\theta_{Ee}) = \prod_{t \in R} P(t|\theta_{Ee})^{n(t,R)}, \qquad (2)$$

where $n(t, R)$ is the number of times t occurs in R. To estimate the co-occurrence language model θ_{Ee} we aggregate term probabilities from documents in which the two entities co-occur:

$$P(t|\theta_{Ee}) = \frac{1}{|D_{Ee}|} \sum_{d \in D_{Ee}} P(t|\theta_d), \qquad (3)$$

where D_{Ee} denotes the set of documents in which E and e co-occur and $|D_{Ee}|$ is the number of these documents. $P(t|\theta_d)$ is the probability of term t within the language model of document d:

$$P(t|\theta_d) = \frac{n(t, d) + \mu \cdot P(t)}{\sum'_t n(t', d) + \mu}, \qquad (4)$$

where $n(t, d)$ is the number of times t appears in document d, $P(t)$ is the collection language model, and μ is the Dirichlet smoothing parameter, set to the average document length in the collection.

Homepage Finding. To gather possible homepage URLs we get the external links on an entity's Wikipedia page and submit the entity name as a query to an index of a large web crawl, collecting URLs of the top relevant documents. We then rank the URLs through a linear combination of their retrieval score and a score proportional to a URL's rank among the external links, with equal weights to both components.

3. EXPERIMENTS AND RESULTS

Our document collection is the ClueWeb09 Category B subset [2]. Named entity recognition is difficult to realize on a data set the size of ClueWeb. Instead we use Wikipedia as a repository of known entities. For our estimations we use the entity names as queries to an index of this collection to obtain co-occurrence counts. We perform two types of evaluation: first, on the intermediate components by comparing the entity strings to a ground truth established by extracting all primary Wikipedia pages from the TREC 2009 Entity qrels. Our second type of evaluation, is the end-to-end evaluation on the original TREC REF task. Specifically, we use R-Precision (R-prec), where R is the number of relevant entities for a topic, and recall at rank 100 (R@100). We also report on the metrics used at the TREC 2009 Entity track: P@10, nDCG@R, and the number of primary homepages retrieved (#pri). We forego significance testing as the minimal number of topics (25) recommended is not available. Table 1 shows the results when ranking without (left half) and with (right half) context; type filtering is always applied, optimized either for precision (top half) or recall (bottom half). The left half of the table shows R-precision and R@100 of the pure co-occurrence model including type filtering. We find that of the two estimators for the pure co-occurrence component χ^2 performs best in terms of precision and that MLE performs best in terms of recall. Comparing the top half with the bottom half of the table we find that the type filtering component can be used to increase either precision or recall. The highest recall is obtained by using MLE and level 6 category expansion. The right half of Table 1 shows results for the context dependent model. In both cases (optimized for precision/recall), R-precision and R@100 are improved further.

Co-occ.	Pure Co-Occurrence		Context Dependent	
	R-Prec	R@100	R-Prec	R@100
Optimized for Precision				
MLE	.1512	**.5423**	.1898	**.5423**
χ^2	**.2382**	.4891	**.2623**	.4747
Optimized for Recall				
MLE	.0799	**.5821**	.0966	**.6982**
χ^2	**.2281**	.5474	**.2399**	.5418

Table 1: Results for the pure co-occurrence and context dependent model with filtering for either precision or recall.

On the end to end evaluation when optimized for precision (P@10=.2100, nDCG@R=.1198) we improve substantially over the median results achieved at TREC 2009 (P@10=.1030, nDCG@R= .0650). When optimized for recall our model surpasses the top performing team in terms of the number of primary homepages retrieved (#pri: 171 vs. 137, out of 396). We use this as a starting point for improving precision of our model by adding additional heuristics: (i) improved type filtering by utilizing high quality type definitions in the DBpedia ontology and (ii) co-occurrence based on anchor text. Anchor based co-occurrence emphasizes candidate entities that occur on the Wikipedia page of the source entity as anchor text or vice-versa. We find that with these additional heuristics our model (P@10=.3000, nDCG@R=.1562) achieves performance comparable to the median of the top 3 at TREC (P@10=.3100, nDCG@R=.1689) in terms of precision, while maintaining exceptionally high recall scores (#pri=174/396).

Acknowledgements. Supported by the European Union's ICT Policy Support Programme as part of the Competitiveness and Innovation Framework Programme, CIP ICT-PSP under grant agreement nr 250430, by the DuOMAn project carried out within the STEVIN programme which is funded by the Dutch and Flemish Governments under project nr STE-09-12, by the Netherlands Organisation for Scientific Research (NWO) under project nrs 612.066.512, 612.061.814, 612.061.815, 640.004.802, and partially by the Center for Creation, Content and Technology (CCCT).

4. REFERENCES

[1] K. Balog, A. P. de Vries, P. Serdyukov, P. Thomas, and T. Westerveld. Overview of the TREC 2009 entity track. In *TREC '09*.

[2] ClueWeb09. The ClueWeb09 dataset, 2009. URL: `http://boston.lti.cs.cmu.edu/Data/clueweb09/`.

[3] C. D. Manning and H. Schuetze. *Foundations of Statistical Natural Language Processing*. The MIT Press, 1999.