Entity Came to Rescue - Leveraging Entities to Minimize Risks in Web Search

Xitong Liu, Peilin Yang and Hui Fang
University of Delaware, Newark, DE, USA
{xtliu,franklyn,hfang}@udel.edu

1 Introduction

We present the summary of our work in the TREC 2014 Web Track. We participated both the ad hoc task and risk-sensitive task and explored two entity-based approaches to evaluate the performance of leveraging entities to improve retrieval effectiveness and robustness.

Our proposed approaches are based on the integration of related entities of queries and the entity model from knowledge base to the retrieval model. The first approach is called as entity-centric query expansion, in which we integrate the related entities into the original query model to perform query expansion. Documents are then retrieved based on the expanded query model. In the second approach, we leverage the publicly available Freebase annotation on ClueWeb12 as well as Freebase API to estimate the entity model. It is called Latent Entity Space (LES), in which we model the relevance between query and document in a latent space. Each dimension of the latent space is represented by an entity and the query-document relevance is estimated based on their projections to each dimension.

The evaluation results on ad hoc task show that entities can indeed bring further improvements on the performance of Web document retrieval when combined with axiomatic retrieval model with semantic expansion, one of the state-of-the-art methods. Furthermore, results on risk-sensitive task demonstrate that our proposed model also have advantage on minimizing the retrieval risk.

2 The Freebase Knowledge Base

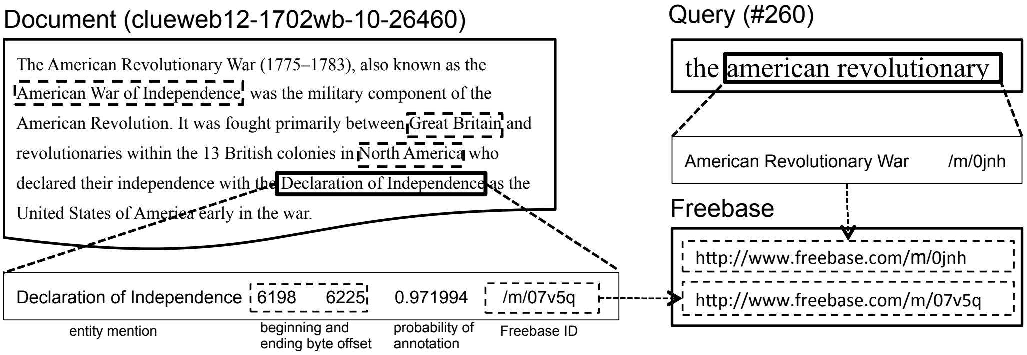
Recent study [2] revealed that nearly half of queries issued to major commercial Web search engines bear entities (e.g., person, location, organization, etc.), and there is an increasing trend for it. On the other hand, the wide existence of entities in Web documents has been known for a while, the advance of information extraction technologies recently makes it much easier to efficiently extract entities from Web-scale data than before, opening opportunities to leverage entities for many information access tasks. Clearly, understanding entities in queries and documents would bring potential benefits to the retrieval performance.

The boom of Web technology yield the birth of many well curated knowledge based including Wikipedia, DBpedia and Freebase, which provide easy interface for people to access high-quality information about entities in structured format. The rich entity information provided by knowledge bases makes it possible to be leveraged to help document retrieval. We leverage Freebase to serve as the knowledge base. The huge volume of ClueWeb12 data imposes several challenges on how to process the data including extraction of entities. Fortunately, Google performed entity extraction over the whole ClueWeb12 collection based on their in-house infrastructure and makes the entity annotation data freely available [1] to the public for research purpose. With the annotation data, we can easily fetch all the entities for a given document and link them back to Freebase through unique ID. Besides, we manually performed entity extraction over the 50 queries, as there is no freely available toolkit to perform entity extraction on extreme short text like queries with satisfying precision. Figure 1 demonstrates some example entity annotations for a document from ClueWeb12 and a query from the data of this year.

3 Retrieval Methods

3.1 Entity-Centric Query Expansion

The entity linking results on unstructured data (e.g., Web data) makes it possible to leverage the integrated information about entities from both knowledge base and Web data to improve document retrieval. We follow our previous work [3]


Document (clueweb12-1702wb-10-26460)
Query (#260)
the american revolutionary
American Revolutionary War /m/0jnh
Freebase
http://www.freebase.com/m/0jnh
http://www.freebase.com/m/07v5q
Declaration of Independence 6198 6225 0.971994 /m/07v5q
entity mention
beginning and ending byte offset
probability of annotation
Freebase ID

The American Revolutionary War (1775–1783), also known as the American War of Independence was the military component of the American Revolution. It was fought primarily between Great Britain and revolutionaries within the 13 British colonies in North America who declared their independence with the Declaration of Independence as the United States of America early in the war.

Figure 1: Example Freebase annotations on ClueWeb12 (Not all entity annotations are displayed).

to exploit the entity linking information to find related entities and integrate them back for document retrieval through query expansion. Formally, we have:

$$S(Q,D) \propto \sum_{w} \big((1-\lambda)p(w|\theta_q) + \lambda p(w|\theta_{ER})\big) \log p(w|\theta_d) \tag{1}$$

where θ_{ER} is the estimated expansion model from related entities and can be estimated in two approaches: (1) entity name based; (2) entity relation based. More details about how to find the related entities and how to estimate entity expansion model can be found in our previous work [3].

3.2 Latent Entity Space

The relevance between document d and query q is estimated based on the probability $p(\mathcal{R}=1|q,d)$, where $\mathcal{R}$ is a binary random variable denoting the relevance. We propose to model the it using a *latent entity space*. Each dimension is represented by an entity, and a query is generated from a mixture of all the dimensions. Thus, we can factor the log-odds ratio $p(\mathcal{R}=1|q,d)$ as follows:

$$p(\mathcal{R}=1|q,d) \overset{\text{rank}}{=} \log \frac{p(d,q|\mathcal{R}=1)}{p(d|\mathcal{R}=0)p(q|\mathcal{R}=0)} \overset{\text{rank}}{=} \log \frac{\sum_{e\in\mathcal{E}} p(d,q|e,\mathcal{R}=1)p(e|\mathcal{R}=1)}{p(d|\mathcal{R}=0)} \overset{\text{rank}}{=} \sum_{e\in\mathcal{E}} p(q|d,e,\mathcal{R}=1)\cdot p(e|d,\mathcal{R}=1).$$

As it is not practical to estimate the joint conditional probability $p(q|d,e,\mathcal{R}=1)$ directly, we use the linear interpolation of $p(q|e,\mathcal{R}=1)$ and $p(q|d,\mathcal{R}=1)$ to estimate it, and obtain:

$$p(\mathcal{R}=1|q,d) \overset{\text{rank}}{=} \lambda \sum_{e\in\mathcal{E}} \underbrace{p(q|e,\mathcal{R}=1)}_{\text{query projection}} \cdot \underbrace{p(e|d,\mathcal{R}=1)}_{\text{document projection}} + (1-\lambda)p(q|d,\mathcal{R}=1). \tag{2}$$

where λ balances the importance of two probabilities. The first component essentially is LES. For a given document d, we first choose an entity $e \in \mathcal{E}$ to represent one semantic aspect of d with probability $p(e|d,\mathcal{R}=1)$, and then generate the query q conditioned on e with probability $p(q|e,\mathcal{R}=1)$. The second component $p(q|d,\mathcal{R}=1)$ is the query likelihood and can be estimated by existing language modeling based approaches. $p(e|d,\mathcal{R}=1)$ can be interpreted as the projection of d on the dimension of e in the latent space, and we leverage KL-divergence to estimate it:

$$p(e|d,\mathcal{R}=1) = p(e|\theta_d,\mathcal{R}=1) \propto -D_{KL}(\theta_e||\theta_d),$$

where θ_e denotes the profile model of e, θ_d can be obtained through maximum likelihood estimation.

$p(q|e,\mathcal{R}=1)$ can be interpreted as the probability that q is generated from the profile model of e (i.e., θ_e). It actually serves as the weight of dimension represented by e in the latent space. We propose to estimate it based on the similarity between entities in query (i.e., $e_q \in q$) and the target entity e:

$$p(q|e,\mathcal{R}=1) = \sum_{e_q\in E(q)} p(e_q|e,\mathcal{R}=1) \propto \sum_{e_q\in E(q)} sim(\theta_{e_q},\theta_e), \tag{3}$$

where $E(q)$ is the set of all entities in q and θ_{e_q} denotes the profile model of e_q, $sim(\theta_{e_q},\theta_e)$ represents the similarity between θ_{e_q} and θ_e Since both θ_{e_q} and θ_e are of the same type, we choose cosine similarity, a pairwise symmetric distance-based measure to estimate $sim(\theta_{e_q},\theta_e)$.

We notice that the estimation of both $p(q|e,\mathcal{R}=1)$ and $p(e|d,\mathcal{R}=1)$ require θ_e, the entity profile model. We proposed two approaches to estimate it:

Run	ERR-IA@10	ERR-IA@20	nDCG@20	ERR@20
RM	0.50414	0.51304	0.24286	0.15296
TR	0.53177	0.54235	0.25979	0.18872
median	-	0.57472	0.25489	0.16679
UDInfoWebAX	0.60154	0.60756	0.30655	0.20704
UDInfoWebENT	0.62148	0.62771	0.30736	0.20203
UDInfoWebLES	**0.68243**	**0.68809**	**0.32295**	**0.22700**

Table 1: Results of submitted runs in ad hoc task. **RM** and **TR** are the results of official runs from Indri and Terrier, respectively. **median** is the mean of per-topic median for all submitted runs.

- **Build entity profiles from scratch**: One entity may be mentioned in multiple documents, each of which carries some information of the entity. By pooling all the information together, we aim to get the full picture of the entity like solving the jig-saw puzzle. Specifically, we adopt language modeling to estimate θ_e as follows:

$$p(w|\theta_e) = \frac{1}{|\mathcal{C}(e)|} \sum_{c(e) \in \mathcal{C}(e)} p(w|c(e)) = \frac{1}{|\mathcal{C}(e)|} \sum_{c(e) \in \mathcal{C}(e)} \frac{n(w, c(e))}{\sum_{w'} n(w', c(e))}$$

  where $c(e)$ is a context of e from a document and $\mathcal{C}(e)$ is the set of all contexts in which e occurs, including a sequence of σ terms before and after e. σ is set to 40 in our experimental setup.

- **Leverage existing knowledge bases**: Knowledge bases provide a portal to access full spectrum of information about entities. For each entity mapped to Freebase, we leveraged the Freebase API to fetch the description field (`/common/topic/description`) and apply maximum likelihood estimation to get the entity profile as it provides much richer textual information than other fields.

4 Experiment Results

4.1 Ad hoc task

We submitted three runs to the ad hoc task, summarized as follows:

1. **UDInfoWebAX**: Axiomatic approach with semantic term expansion [4]. The related terms are selected from Web-based working set. It performs empirically well on the 2013 Web track data and serves as a strong baseline.
2. **UDInfoWebENT**: Entity-centric query expansion, with expansion model estimated from entity name based approach. The original query model θ_q in Equation (1) is estimated by **UDInfoWebAX**.
3. **UDInfoWebLES**: The latent entity space method. The entity models are estimated from Freebase profile and the query likelihood $p(q|d, \mathcal{R} = 1)$ in Equation (2) is estimated from **UDInfoWebAX**. It is selected as the top-ranked submission.

The parameters for all the submitted runs are trained on the 2013 data. We use Indri with default language model to retrieve 15,000 top ranked documents for each query and apply Waterloo spam filter to remove documents with spam ranking percentile scores less than 70 to build the test collection. Evaluation results are summarized in Table 1. We observe that **UDInfoWebAX** performs much better than **RM**, **TR** and **median**, which is consistent with observations on 2013 data, and it is already a very strong baseline in term space. Moreover, **UDInfoWebLES** shows superior performance over **UDInfoWebAX**, particularly in ERR-IA@10 and ERR-IA@20, demonstrating the effectiveness of latent entity space model as it could capture additional semantic relevance in entity space which are missed by existing term space based approaches. Besides, **UDInfoWebENT** could still bring additional improvements to **UDInfoWebAX**. In conclusion, entities could bring additional benefits to ad hoc Web document retrieval.

4.2 Risk-sensitive task

We choose latent entity space model for the risk-sensitive task as it is selected as the top-ranked submission. We observe that the interpolation parameter λ in Equation (2) provides a natural approach to balance the risk and gain between latent entity space model and query likelihood. By increasing λ, we are giving more weight to the relevance score estimated by latent entity space model, but running at the risk of introducing more uncertainty. In contrast, by decreasing λ, we are more conservative and give less weight to latent entity space model. We optimize the

Run	**UDInfoWebRiskRM**		**UDInfoWebRiskTR**		**UDInfoWebRiskAX**	
Baseline	ERR-IA@10	ERR-IA@20	ERR-IA@10	ERR-IA@20	ERR-IA@10	ERR-IA@20
RM	-0.19617	-0.19334	-0.23410	-0.23106	**-0.18199**	**-0.17925**
TR	-0.24442	-0.24824	-0.25888	-0.25787	**-0.20209**	**-0.20063**
UDInfoWebAX	-0.23415	-0.22984	-0.26263	-0.25323	**-0.19444**	**-0.18426**
UDInfoWebENT	-0.28181	-0.28286	-0.32310	-0.31867	**-0.25924**	**-0.25192**
UDInfoWebLES	-0.30078	-0.29851	-0.28808	-0.28225	**-0.17853**	**-0.17384**

Table 2: Results of submitted runs in risk-sensitive task ($\alpha = 5$).

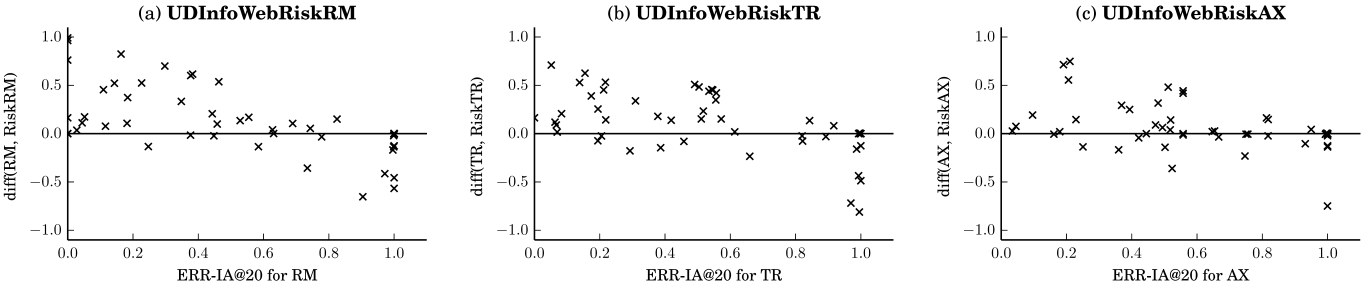


Figure 2: Impacts of latent entity space model on different baselines (ERR-IA@20).

parameter with $\alpha = 5$ against three baselines: relevance model from Indri, Terrier and **UDInfoWebAX**, denote them as **UDInfoWebRiskRM**, **UDInfoWebRiskTR** and **UDInfoWebRiskAX** respectively. Results are summarized in Table 2. For each submitted run, the risk sensitive measure for the two official baselines and our three submitted runs to ad hoc task are reported. We observe that **UDInfoWebRiskAX** could always outperform other two runs when compared with all the five baselines, implying that latent entity space model works best when combined with **UDInfoWebAX** on minimizing risk. The λ in **UDInfoWebRiskAX** is set to 0.7 based on training data, while the λ in **UDInfoWebLES** is set to 0.4. It suggests that latent entity space model is more robust than axiomatic approach and should be favored more to minimize risk. To further investigate the impacts of our latent entity space model on different baselines, we plot the distribution of all the queries with regard to the impacts on the performance when it is applied to the query, as illustrated in Figure 2. The x-axis represents the performance of three baselines for each query in ERR-IA@20 and y-axis represents the difference after LES is applied. Points above the $y = 0$ bar means LES improved over the baseline, while points below the $y = 0$ bar means LES hurt the performance. Clearly, latent entity model could improve most of the hard queries while hurting a few easy queries.

5 Conclusion

We report our methods and experimental results on TREC 2014 Web track in this paper. We explored two entity based approaches to integrate entity to improve the performance of Web document retrieval. Experimental results demonstrate that entities could improve retrieval performance in terms of both effectiveness and robustness, in particular for the latent entity space model. We plan to investigate more approaches to explore the potentials of entities for Web document retrieval in the future work.

References

[1] E. Gabrilovich, M. Ringgaard, and A. Subramanya. FACC1: Freebase annotation of ClueWeb corpora, Version 1 (Release date 2013-06-26, Format version 1, Correction level 0). http://lemurproject.org/clueweb09/FACC1/, June 2013.

[2] T. Lin, P. Pantel, M. Gamon, A. Kannan, and A. Fuxman. Active Objects: Actions for Entity-Centric Search. In *WWW*, pages 589–598, 2012.

[3] X. Liu, F. Chen, H. Fang, and M. Wang. Exploiting Entity Relationship for Query Expansion in Enterprise Search. *Information Retrieval*, 17(3):265–294, 2014.

[4] P. Yang and H. Fang. Evaluating the Effectiveness of Axiomatic Approaches in Web Track. In *Proceedings of TREC*, 2013.