Entity Query Feature Expansion using Knowledge Base Links

Jeffrey Dalton, Laura Dietz, James Allan
Center for Intelligent Information Retrieval
School of Computer Science
University of Massachusetts Amherst
Amherst, Massachusetts
{jdalton, dietz, allan}@cs.umass.edu

ABSTRACT

Recent advances in automatic entity linking and knowledge base construction have resulted in entity annotations for document and query collections. For example, annotations of entities from large general purpose knowledge bases, such as Freebase and the Google Knowledge Graph. Understanding how to leverage these entity annotations of text to improve ad hoc document retrieval is an open research area. Query expansion is a commonly used technique to improve retrieval effectiveness. Most previous query expansion approaches focus on text, mainly using unigram concepts. In this paper, we propose a new technique, called entity query feature expansion (EQFE) which enriches the query with features from entities and their links to knowledge bases, including structured attributes and text. We experiment using both explicit query entity annotations and latent entities. We evaluate our technique on TREC text collections automatically annotated with knowledge base entity links, including the Google Freebase Annotations (FACC1) data. We find that entity-based feature expansion results in significant improvements in retrieval effectiveness over state-of-the-art text expansion approaches.

Categories and Subject Descriptors

H.3.3 [**Selection Process**]: [Information Search and Retrieval]

Keywords

Entities; Ontologies; Information Retrieval; Information Extraction

1. INTRODUCTION

Today's commercial web search engines are increasingly incorporating entity data from structured knowledge bases into search results. Google uses data from their Knowledge Graph and Google Plus, Yahoo! has Web Of Objects, Bing incorporates Facebook and Satori entities, and Facebook searches over entities with Graph Search. However, the majority of content created on the web remains unstructured text in the form of web pages, blogs, and microblog posts. For many search tasks, these documents will continue to be the main source of content for users. In this work, we address the task of ad hoc document retrieval leveraging entity links to knowledge bases in order to improve the understanding and representation of text documents and queries. We demonstrate that this gain in semantic understanding results in significant improvements in retrieval effectiveness.

We bridge the gap between entities and text using automatic information extraction to identify entities and link them to a knowledge base. The task of 'entity linking' to a knowledge base has received significant attention, with one major venue being the Text Analysis Conference (TAC) Knowledge Base Population (KBP) Entity Linking Task [17]. In this task traditional named entities (people, geo-political entities, and organizations) are linked to a knowledge base derived from Wikipedia. Beyond TAC, there is increasing interest in more general concept entities, with the task of 'wikifying' [28, 16, 19] documents by linking them to Wikipedia. Beyond information extraction, content owners are augmenting HTML markup with embedded structured data through standardized markup efforts such as schema.org. A study from 2012 showed that 30% of web documents contain embedded structured data in RDFa or Microformats [26].

Google recently released the FACC1 dataset [15] for the TREC ClueWeb09 and ClueWeb12 web collections. The dataset contains automatically extracted entity mentions from web documents that are linkable to the Freebase knowledge base [6]. Freebase is a publicly available general purpose knowledge base with over 42 million entities and over 2.3 billion facts.[1] The FACC1 dataset is the first publicly available web-scale collection of entity linked documents. In addition to annotated documents, the FACC1 data also contains explicit manual annotations for the TREC web track queries. We present one of the first published experiments using this data for retrieval.

For this work, we define an *entity* broadly to be a thing or concept that exists in the world or fiction, such as a person, a battle, a film, or a color. We focus primarily on entities that are linked to two existing publicly available knowledge bases, Wikipedia and Freebase. We use a combination of both of these knowledge bases because they provide complementary information. Wikipedia provides rich text and link associations. Freebase provides a significantly larger database of concepts, many of whom may not meet Wikipedia's standards for notability, with structured data in RDF, including categories and types.

Our work addresses two fundamental research areas using entity annotations for ad hoc retrieval. The first is the representation of both queries and documents with linked entities. What features, if any, improve retrieval effectiveness? The second is inferring latent

[1] As of January 27, 2014 according to Freebase.com

SIGIR'14, July 6–11, 2014, Gold Coast, Queensland, Australia.

ACM 978-1-4503-2257-7/14/07 ...$15.00.
http://dx.doi.org/10.1145/2600428.2609628.

entities (and more importantly, features of entities and terms) for an information need.

The FACC1 annotations include entity annotations for queries. However, these annotations are limited to entities that are explicitly mentioned, where we hypothesize that many more latent entities are relevant to the users' information need. For example, consider the TREC query about [Barack Obama family tree]. There are explicit query entities, [Barack_Obama] and [Family_Tree]. There are also relevant latent entities such as [Ann_Dunham], [Michelle_Obama], [Barack_Obama_Sr], [Ireland], [Kenya], [DNA], and others.

One issue is that explicit entity mentions have the same fundamental problems of query-document mismatch as words. For example, a document on the topic of Obama's family history may not explicitly refer to a [Family_Tree], but may refer to other related entities, such as a [Family_Crest] and [Genealogy]. In addition, for many existing collections, no explicit entity annotations for queries exist. In both cases, it is important to infer related entities and expand the query representation.

Entities provide a wealth of rich features that can be used for representation. These include text as well as structured data. Some of the important attributes that we highlight for these experiments include: fine-grained type information (athlete, museum, restaurant), category classifications, and associations to other entities. Although we do not explore them in detail in this work we also observe that the knowledge base contains rich relational data with attributes and relations to other entities. These attributes include: gender, nationality, profession, geographical information (latitude, longitude), and temporal attributes (such as birth and death), and many more depending on the type of entity.

We hypothesize that the language in the document contexts of entity mentions differs from that found in Wikipedia or in the knowledge base description. But, mentions of the entity are also contained in text documents across the entire corpus. To address this, we propose new query-specific entity context models extracted from snippets in the feedback documents surrounding the entity's annotations. We further hypothesize that this context information will allow us to identify entities that are relevant to the query and use their presence as signals of document relevance.

To summarize, the main contributions of this work are:

- Introducing new query expansion techniques with feature-based enrichment using entity links to a knowledge base
- Demonstrating significant improvements in retrieval effectiveness when entity features are combined with existing text approaches
- Proposing a new entity modeling technique for building query-specific context models that incorporate evidence from uncertainty inherent in automatic information extraction
- Performing the first published experiments using the FACC1 Freebase annotations for ad hoc document retrieval
- Analyzing the ClueWeb09 FACC1 annotations for their use in retrieval applications
- Providing new entity-annotated query datasets for the TREC web track queries that substantially improve entity recall

The remainder of the paper is structured as follows. Section 2 provides retrieval model background. In Section 3, we introduce the new feature expansion approach and introduce the entity context feedback model (3.4). We experimentally evaluate our approach in Section 4 on standard TREC test collections including: Robust'04, ClueWeb09B, and ClueWeb12B. Connections to related work are discussed in Section 6 before concluding.

2. BACKGROUND

2.1 Notation

We distinguish notationally between random variables in upper case (e.g. E) and possible assignments (e) in lower case. We denote count statistics of a configuration e in a sequence of as e_i.

2.2 Log-linear Models

Graphical models [20], such as Markov Random Fields (MRF), are a popular tool in both information extraction and information retrieval. Dependencies between two (or more) variables (e.g. Queries and Documents) are encoded by factor functions that assign a non-negative score to each combination of variable settings. Regarding the factor function in log space allows for arbitrary scores.

Factor functions (or similarity functions) between two variables are indicated by ϕ (e.g. $\phi(Q, W)$) which is assumed to be of log-linear form. This means that ϕ is determined by an inner product of weight vector θ and feature vector $\boldsymbol{f}$ in log-space.

2.3 Retrieval Models

The query likelihood (QL) retrieval model can be represented as a factor between a multi-word query, and a document represented as a bag of words as $\phi(Q, D) = \prod_{w_i \in Q} \phi(w_i, D)$.

Within this framework, we adopt both the QL model and the widely used Sequential Dependence Model (SDM) [24], which incorporates word unigrams, adjacent word bigrams, and adjacent word proximity. The feature function used to match words, W to a document is a Dirichlet smoothed probability:

$$\phi(W, D) = \log \frac{\#(W, D) + \mu \frac{\#(W,C)}{|C|}}{|D| + \mu} \tag{1}$$

This approach generalizes to bigrams "W_1, W_2" and unordered term proximity. Furthermore, we can apply it to different kinds of vocabularies, such as entity identifiers or categories with appropriate redefinition of the document length $|D|$ and collection statistics.

2.4 Query Expansion

One commonly used query expansion model is the Relevance Model (RM) [22]. It is a pseudo-relevance feedback approach that uses retrieved documents to estimate the query topic. Relevant expansion terms are extracted and used in combination with the original query (the RM3 variant). We use this as our baseline text-based expansion model. Beyond unigrams, Metzler and Croft propose a generalized model, Latent Concept Expansion (LCE) [25], which models arbitrary expansion vocabularies such as words, entity identifiers, types or categories [25].

In both relevance modeling and LCE, the formulation is similar. Assuming that the retrieval score represents the probability of the document under the query, e.g. $p(D|Q)$, document-wise multinomial distributions over a vocabulary $p(V|D)$ are combined via a mixture model.

$$p(V|Q) = \sum_d p(V|d)p(D = d|Q) \tag{2}$$

Hyperparameters of this approach are the number of expansion documents, number of expansion features, and a balance parameter for weighting the original query against the expanded query, which are further weighted according to $P(V|Q)$.

The document probability, $p(D = d|Q)$ is typically derived from the retrieval score $s(d)$ by exponentiation and re-normalization over the domain of expansion documents. The document specific

Table 1: Example expansion terms for the query "Obama Family Tree"

Words	Entity ID	Wiki Categories	Freebase Type
family	Barack_Obama	cat:first_families_u.s.	/people/family
tree	Michelle_Obama	cat:political_families_u.s.	/book/book_subject
genealogy	Family_Tree	cat:bush_family	/location/country
surname	Family_Crest	cat:american_families_english	/film/film_subject
history	Barack_Hussein_Obama_Sr	cat:american_families_german	/base/presidentialpets/first_family
crest	Family_History	cat:business_families_u.s.	/base/webisphere/topic

```
#combine(
  #sdm(  obama family tree )
  #sdm( [Barack_Obama] [Family_Tree] )
  #sdm( {US President} {Politician})
  #sdm( [Michelle_Obama] [Ireland] [Kenya])
)
```

Figure 1: Example expansion of query C09-1 with entities [] and Freebase types {}.

distribution of features is derived under the multinomial assumption by $p(V|d) = \frac{\#(V \in d)}{\sum_{V'} \#(V' \in d)}$.

3. ENTITY QUERY FEATURE MODEL

In this section we introduce the representation of queries and documents using linked entities and provide background on the models we use throughout this work.

In a preprocessing step, documents in the collection are annotated with entity links. Entity links establish a bidirectional reference from words to entities in the KB, and indirectly to Freebase types and Wikipedia categories and further related entities in the knowledge base (called neighbors, henceforth). We index these different vocabulary representations for each document in different fields. Our retrieval engine supports proximity and approximate matches with respect to each of the vocabularies.

The goal is to derive expansions across the different kinds of vocabularies such as words W, entities E, types T, and categories C to retrieve annotated documents with the goal of maximizing document retrieval effectiveness.

Figure 1 details expansions for the ClueWeb09B query 1 "obama family tree" for the words, entities and Freebase types. The first three entries constitute words, entities and types directly mentioned in the query, where the last entry includes other relevant entities. A sample of the expansion terms from our system on this query are given in Table 1.

Expansions in different vocabularies can be derived through multiple options. Entity linking the query provides very precise indicators, but may also miss many of the relevant entities. Alternative expansion entities can be found using pseudo-relevance feedback on the document collection containing entity annotations or alternatively by issuing the query against an index of the knowledge base and considering top-ranked entries. Figure 2 gives an overview of all possibilities studied in this work, which we detail in this section.

3.1 Annotated Query

The query, Q, is given as a sequence of keywords $w_1 w_2, ...w_{|Q|}$. Aside from representing the query Q by their query word representation W, we can annotate the query in the same way we preprocess

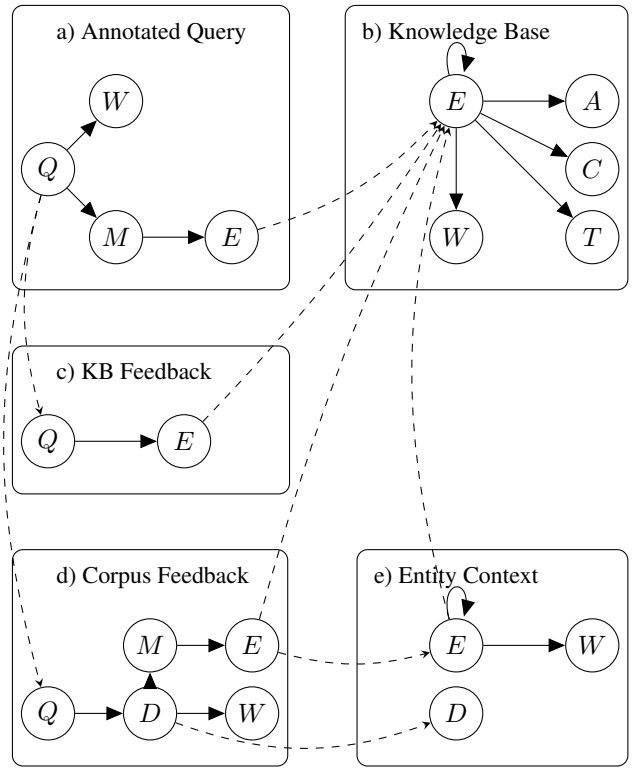


Figure 2: Overview over feature sources.

the documents before indexing. This provides annotations for all entity mentions M in the query terms together with a link to the KB entity E (cf. Figure 2a)

Resolution through the entity provides further information about its type, category and name alias information. We can additionally infer indirectly related entities by following hyperlinks on Wikipedia articles or exploiting Freebase relations (cf. Figure 2b).

For instance, terms on the entity's Wikipedia article provide a resource for related words W. These are derived through a hierarchical multinomial model by integrating over mentions and entities

$$f_{\text{ExplWiki}}(Q, W) = \sum_M \left(\sum_E p(W|E) p(E|M) \right) p(M|Q)$$

In the equation, $p(M|Q)$ is a uniform distribution over annotated mentions and $p(E|M)$ is the entity disambiguation confidence and $p(W|E)$ refers to the language model of the entity's article.

In addition to words, we access different alternative names A in the knowledge base through the entity link. Our knowledge base contains name aliases of different confidences, e.g. title versus anchor text, which we take into account through the multinomial distribution $p(A|E)$.

3.2 KB Feedback

An alternative route can be taken by issuing the query against a search index containing all knowledge base entries (cf. Figure 2c). The ranking of articles can be interpreted as a distribution over entities, encoded in the feature $\boldsymbol{f}_{\text{KB}}(Q, E)$ which is obtained by exponentiating and renormalizing the retrieval score $s_Q(E)$ of the entity under the query.

$$\boldsymbol{f}_{\text{KB}}(Q, E) = \frac{1}{Z} \exp s_Q(E)$$

Here, Z ensures normalization across the feedback entities.

For instance we can derive a distribution over words W from the highest ranked entities. This has been found to be effective in the related work [3, 34]. Further vocabularies over name aliases, related entities, types and categories can be derived as explained above.

3.3 Corpus Feedback

We can also apply a variation on pseudo-relevance feedback which we extend to document annotations (cf. Figure 2d). The unaltered relevance model provides feature $\boldsymbol{f}_{\text{RM}}(Q, W)$ by integrating $p(D|Q)$ and $p(W|D)$ over documents in the feedback set.

In a similar fashion we can derive a distribution over all mentions M, denoted by the feature $\boldsymbol{f}_{\text{RM}}(Q, M)$. Mentions include both the string that is linked to an entity as well as unlinked Named Entity Spans (NERs). Even if these mentions M cannot be linked to the knowledge base, they provide useful entity-like information for expansion, as used by Callan et al. [7].

For linked mentions, the entity link disambiguation probability gives an alternative indicator for relevant entities E.

$$\boldsymbol{f}_{\text{RM}}(Q, E) = \sum_D \left(\sum_M p(E|M) p(M|D) \right) p(D|Q)$$

The disambiguation confidence distribution $p(E|M)$ has an effect in cases where multiple entities have a high disambiguation probability for a given mention. In the experimental section we explore options ranging from noisy indicators to high-precision links, such as using only the highest ranked entity or additionally applying a NIL classification. In these conservative options we define $p(E|M) = 1$ for the most confident (and non-NIL) linked entity E and 0 otherwise.

From the distribution over entities, we can follow the connection to the knowledge base (cf. Figure 2b) and derive distribution over name aliases, types, categories, and neighbor entities.

3.4 Entity Context Model

The corpus feedback provides distributions over entities. However, it is likely that relevant entities are referred to in a query-specific way which differs from the global term distribution in the knowledge base. For instance in the query "obama family tree" we expect the entity [Barack_Obama] to be referred to by personal names and less via his political function. Also, some related entities (in particular family members) are more important than others.

Our goal is to develop query-specific distributions over name aliases and related entities by inspecting the local context surrounding entity annotations for co-occurrences of entities with words and pairs of entities. In our experiments, we create three versions of each entity's query-specific context model, varying the size of the context snippets: 8 words on either side of a mention, 50 words on either side, or one sentence, where sentence boundaries are determined by a sentence-splitter.

From each feedback document D and each annotated entity mention, M, we build entity context snippets using the only contextual window around the annotation. For each entity, E, we aggregate all snippets by weighting them by the document retrieval probability $p(D|Q)$.

The entity context model for a given entity, E, provides a distribution over words, W, which is used for the context model feature $\boldsymbol{f}_{\text{ECM}}(E, W)$. Likewise, the entity context model provides a distribution over co-occurring neighboring entities E' as $\boldsymbol{f}_{\text{ECM}}(E, E')$. And by following the link to the knowledge base, features over co-occurring name aliases, types, and categories.

3.5 Learning Feature Weights

So far we introduced several features $\boldsymbol{f}$ for query expansion with words, entities, mentions, types, categories, and neighbors using various options to traverse available information sources, each representing a path in Figure 2.

The large number of features renders grid-tuning approaches infeasible. We exploit that our model falls into the family of log-linear models and can therefore be efficiently estimated with a learning-to-rank approach.

For every feature, $\boldsymbol{f}$, we build the expansion model induced by this feature only. For example from $\boldsymbol{f}_{\text{RM}}(Q, E)$ we build an expansion model over entities $p_{RM}(E)$ by normalizing across all entity candidates E'.

$$p_{RM}(E) = \frac{\boldsymbol{f}_{\text{RM}}(Q, E)}{\sum_{E'} \boldsymbol{f}_{\text{RM}}(Q, E')}$$

For every document d in our training set, we compute the retrieval score under the RM1 expansion model $p_{RM}(E)$ using only the k highest ranked entities weighted by their respective probability under the expansion model.

Following this procedure, each feature function $\boldsymbol{f}(Q, V)$ is converted for all vocabularies V into a feature vector for documents d in the training set. We use a log-linear learning-to-rank approach to optimize the retrieval effectiveness for the target metric under this feature function. This provides the parameter vector $\boldsymbol{\theta}$ which corresponds to the weights for each expansion model, when retrieving rankings for test queries.

By incorporating the retrieval score from the original query Q as a special feature function, this also determines the RM3 balance weight on the original query with respect to all expansion models.

4. EXPERIMENTAL SETUP

This section details the tools and datasets used for our experiments. The retrieval experiments described in this section are implemented using Galago[2], an open source search engine. The structured query language supports exact matching, phrases, and proximity matches needed for our retrieval models. A summary of the document collections used in these experiments is presented in Table 2. The corpora include both newswire (Robust04) and web pages (ClueWeb). During indexing and retrieval, both documents and query words are

[2] `http://www.lemurproject.org/galago.php`

Table 2: Test Collections Statistics.

Name	Documents	Topic Numbers
Robust04	528,155	301-450, 601-700
ClueWeb09-B	50,220,423	1-200
ClueWeb12-B	52,343,021	1-50

stemmed using the Krovetz stemmer [21]. Stopword removal is performed on word features using the INQUERY 418 word stop list. For the web collections, the stopword list is augmented with a small collection of web-specific terms, including "com", "html", and "www". We use title queries which contain only a few keywords.

Across all collections retrieval and feedback model parameters are learned or tuned using 5-fold cross-validation. Instead of selecting a single number of feedback documents or entities, we include expansion feature models with different hyperparameters and learn a weighted combination of these along with other features. We include expansion features from one, ten, and twenty feedback entities and documents. We optimize parameters θ with a coordinate-ascent learning algorithm provided in the open source learning-to-rank-framework RankLib.[3] Parameters are optimized for mean average precision (MAP) effectiveness directly.

Retrieval effectiveness is evaluated with standard measures, including mean average precision (MAP) at 1000. Because several of our collections focus on web search, where precision at the early ranks is important, we also report normalized discounted cumulative gain (NCGD@20) and expected reciprocal rank (ERR@20).

We now describe the aspects of the entities in documents and queries for each collection in more detail.

4.1 Knowledge Base

We use a combination of Wikipedia and Freebase as knowledge bases in these experiments. Many of the Freebase entities are contained in Wikipedia. We use the Freebase schema to map between the two knowledge bases (using attributes: /wikipedia/en_title and /wikipedia/en). These knowledge resources provide the entity features used for query expansion from linked entities described in Section 3. Our Wikipedia collection is derived from a Freebase-generated dump of the English Wikipedia from January 2012, which contains over 3.8 million articles. For each Wikipedia entity we extract an entity representation consisting of its article text, canonical name, categories, and a distribution over aliases from redirects, Wikipedia-internal anchor text, and web anchor text from the Google cross-lingual dictionary [31]. In these experiments we also use a subset of the Freebase data: machine identifiers (MIDs), types, and aliases.

4.2 Robust'04

No publicly available entity annotations exist for Robust04 queries or documents. We do not exploit explicit entity annotations in queries, reducing the model in 3.1 to only the words in the title query. For document analysis, we use the extraction tools in the Factorie [23] NLP library. We use Factorie to perform tokenization, sentence segmentation, named entity recognition, part-of-speech tagging, dependency parsing, and entity mention finding. The entity mentions detected by Factorie are linked to the knowledge base using our state-of-the-art entity linking system, KB Bridge [11], which is trained on the TAC KBP entity linking data from 2009-2012. For each mention, the entity linker provides a distribution over the top fifty most probable entities. Based on the TAC evaluation data, the linker has an F1 score of approximately 80-85%. We note that this entity linker is trained to detect and link traditional named entities (people, organization, and geo-political entities) and may not detect or link conceptual entities. Because of limited resources we do not entity link all documents in the Robust04 collection. Instead, we pool the top one hundred documents from all of the baseline text retrieval runs. For our resulting experiments we perform re-ranking on this pooled set of documents using the entity linking features. We use the top documents as sources for extracting both text and entity features.

[3] http://people.cs.umass.edu/~vdang/ranklib.html

4.3 ClueWeb09 and ClueWeb12

We perform experiments using two web datasets from the TREC web track. The first is the ClueWeb09 dataset. For the queries, we use the title queries, but some entity annotations are derived from the descriptions. The Google FACC1 data provides explicit entity annotations for the web track queries (2009-2012) queries, created by automatically entity linking and manually correcting entities in the text of the topic descriptions. We found these to be missing significant numbers of entities and so manually revised these annotations to improve recall and fix several annotation errors. We discuss these revisions in Section 5.3. For the documents, we use the ClueWeb09 Category-B collection, which consists of 50 million pages, including Wikipedia. For ClueWeb09-B, we apply spam filtering with a threshold of 60, using the Waterloo spam scores [9].

We use the ClueWeb12 collection with the TREC web tack 2013 queries, using only the titles. Similar to Robust04, there are no explicit entity annotations. We do not apply spam filtering on the ClueWeb12 documents because hard spam filtering was shown to hurt all the baseline retrieval runs.

5. EXPERIMENTAL EVALUATION

The effectiveness of our query feature expansion is compared with state-of-the-art word-based retrieval and expansion models. Our baseline retrieval model is the Sequential Dependence Model (SDM) [24]. We also compare to two baseline expansion models. The first is an external feedback model, which uses the Wikipedia knowledge base as a text collection and extracts terms from the top ranked article, which we call WikiRM1. Models similar to WikiRM1 were shown to be effective for these collections in previous work [3, 34]. The second baseline uses collection ranking from the SDM model and builds a collection relevance model, which we call SDM-RM3. For ClueWeb12 we also report an official baseline using Indri's query likelihood model (Indri-QL).

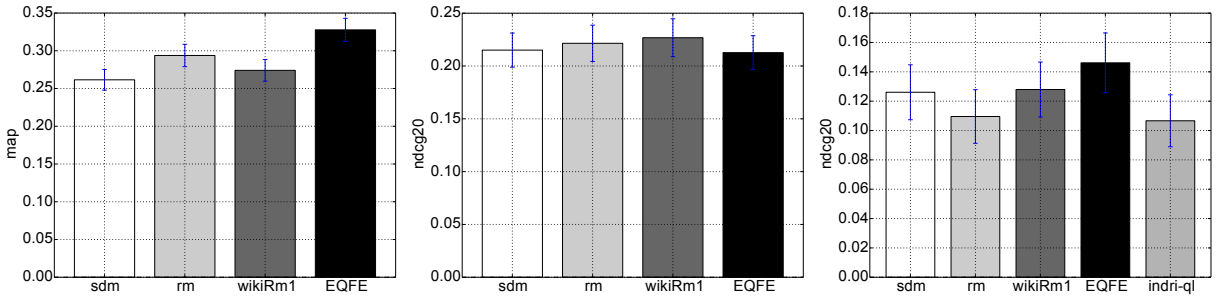
5.1 Overall Performance of EQFE

The overall retrieval effectiveness across different methods and collections is presented in Table 3 and Figure 3. Our EQFE model is the best performer on MAP for Robust04 and best on NDCG@20, ERR@20 and MAP on the ClueWeb12B collection. A paired t-test with α-level 5% indicates that the improvement of EFQE over SDM is statistically significant for both. For ClueWeb09B, the EQFE numbers are slightly worse, but no significant difference is detected among the competing methods. The helps/hurts analysis reveals that EQFE helps a few more times than it hurts in ClueWeb09B. (cf. 4).

In order to analyze whether the EQFE method particularly improves difficult or easy queries, we sub-divide each set into percentiles according to the SDM baseline. In Figure 4 the queries are organized from most difficult to easiest. The 5% of the hardest queries are represented by the left-most cluster of columns, the 5% of the easiest queries in the right-most cluster of columns, the mid-

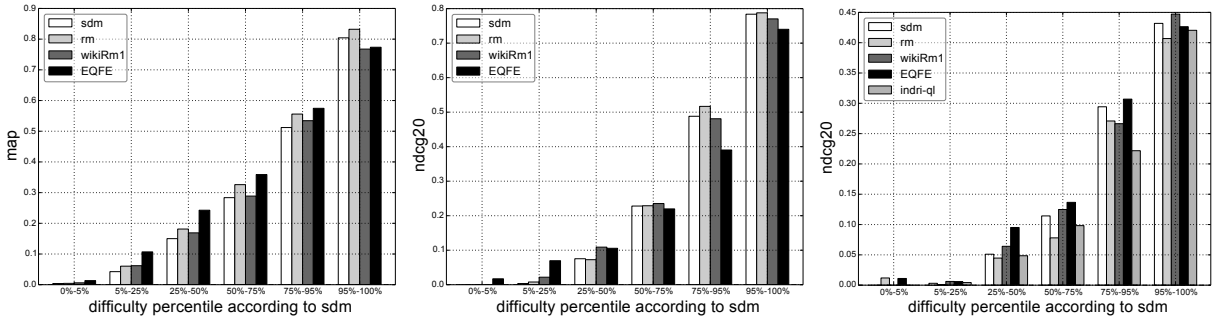
Table 3: Summary of results comparing EQFE for <title> queries across the three test collections.

	Robust04			ClueWeb09B			ClueWeb12B		
Model	MAP	P@20	NDCG@20	MAP	ERR@20	NDCG@20	MAP	ERR@20	NDCG@20
SDM	26.15	37.52	42.37	11.43	13.63	21.40	4.18	9.15	12.61
WikiRM1	27.41	37.71	42.81	11.39	15.29	22.56	4.00	9.31	12.80
SDM-RM3	29.38	38.82	43.44	11.43	13.63	21.40	3.53	7.61	11.00
EQFE	32.77	38.00	42.40	11.00	14.00	21.12	4.67	10.00	14.61



(a) Robust04 (b) ClueWeb09B (c) ClueWeb12B

Figure 3: Mean retrieval effectiveness with standard error bars.



(a) Robust04 (b) ClueWeb09B (c) ClueWeb12B

Figure 4: Mean retrieval effectiveness across different query-difficulties, measured according to the percentile of the SDM method.

Table 4: Queries EFQE helped versus hurt over SDM baseline.

	Queries Helped	Queries Hurt
Robust04	173	47
ClueWeb09B	68	65
ClueWeb12B	26	8

dle half is represented in two middle clusters (labeled "25%-50%" and "50%-75%").

This analysis shows that EQFE especially improves hard queries. For Robust04 and ClueWeb12B EQFE outperforms all methods, except for the top 5% of the easiest queries (cf. 4a and 4c). For ClueWeb09B all queries in the difficult bottom half (cf. 4b) are improved. We want to point out that we achieve this result despite having on average 7 unjudged documents in the top 20 and 2.5 unjudged documents in the top 10 (in both the "5%-25%" and "25%-50%" cluster), which are counted as negatives in the analysis.

The WikiRM1 method, which is the most similar expansion method to EQFE, demonstrates the opposite characteristic, outperforming EQFE only on "easiest" percentiles.

5.2 Feature-by-Feature Study

We study the contribution of each of the features by re-ranking the pooled documents according to the feature score alone and measuring the retrieval effectiveness in MAP. The results for each collection are shown in Figure 5. It shows a subset of the top expansion features. The label on the x-axis has three attributes of the entity features: the vocabulary type, feedback source, and number of expansion terms. The vocabulary types are (A,E,C,W,M, and T from Figure 2). The source is the original query (Q), query annotation ($query\ ann$), corpus feedback (doc), corpus feedback using entity features ($doc - ent$), knowledge base feedback (kb), and entity context model feedback (ecm). The last number in the description usually indicates the number of feedback terms (1, 5, 10, 20, and 50). For ecm it indicates the size of the context model window. We note that for several classes of features have similar names. These are variations of the same expansion feature. For example, the most confident entity (t1), the most confident entity whose score is above the NIL threshold (t1nn), or any entity above the NIL threshold (all).

Entity identifiers E are top features across all collections, but every collection prefers entities expanded by a different paradigm: For Robust from corpus feedback, for ClueWeb09B from the entity context model, and for ClueWeb12B from knowledge base expansion with five entities.

For the Robust04 collection, our study confirms that query keywords are highly indicative of relevance and accordingly words from corpus feedback are strong features. This is in not the case for the ClueWeb collections.

For both ClueWeb collections, the entity context model with window size 8 performs well. Further, name aliases from both corpus feedback and from entity context models are highly effective, even where the entity identifiers themselves are not. We believe this is because the recall of the entity identifiers in the FACC1 data is limited. Here the name aliases bridge this annotation gap.

We note that certain vocabularies such as categories and types do not perform well on their own, but likely help in combination with other features.

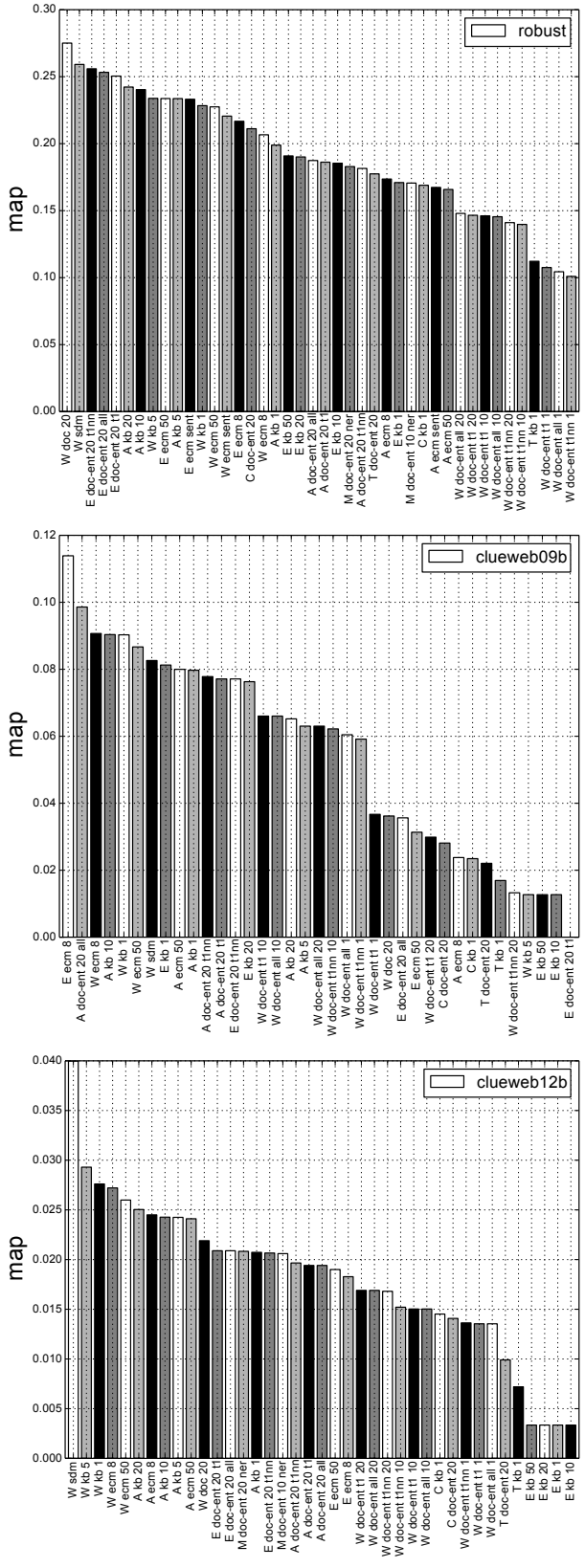


Figure 5: Features sorted by retrieval effectiveness on its own.

Table 5: Different classes of entities are more prevalent in different data set. Number of queries that mention each entity class.

Dataset	Overall	Freebase	NER	PER/ORG/LOC
Robust04	249	243	85	49
Clueweb09	200	191	108	80
ClueWeb12	50	48	26	16

Table 6: Mean Average Precision over subsets of Robust04 queries that mention entities of respective classes.

Method	Overall	Freebase	NER	PER/ORG/LOC
SDM	26.15	26.61	31.11	27.72
MSE	30.49	31.02	36.45	31.98
EQFE	32.77	33.33	38.28	33.31

5.3 Error Analysis of ClueWeb09

We now perform an analysis of the ClueWeb09 results to better understand why EQFE using entity feature expansion does not significantly improve the results. This case is particularly surprising because it is the only dataset where explicit entity query annotations are available.

We first examine the FACC1 query annotations. The FACC1 dataset contains entity annotations for 94 of the 200 queries. Upon inspecting the annotations, we found that despite manual labeling, many entities were not annotated. The queries were manually re-annotated, resulting in 191 of the 200 queries containing an entity. The revised annotations which will be made publicly available on our website.[4] We used the revised query entity annotations for our experiments on ClueWeb09B.

The remaining queries without entity links are interesting. Several contain entities that are not noteworthy enough to be included in existing public knowledge bases, including:"jax chemical company", "fickle creek farm", "sit and reach test", and "universal animal cuts". The remaining are not entity-centric without clearly defined concepts: "getting organized" and "interview thank you".

However, even after coverage of queries is improved, the feature does not appear to help overall. To better understand the contribution (or lack thereof) of explicit entities in the query, we evaluated a query model that uses only entity identifiers to retrieve documents. Surprisingly, the entities alone have poor effectiveness, with a MAP of 0.048, an NDCG@20 of 0.162, and an ERR@20 of 0.123. This is less than half the effectiveness of the SDM baseline. We observe that 72.5% of the documents returned using the entity model are unjudged. The retrieved results differ significantly from the pool of judged documents. Further assessments are required to assess the model effectiveness. Beyond unjudged documents, we also examine the potential for explicit entities by analyzing the relevance judgments.

We analyze the potential for explicit entities using all of the judged documents for the queries. We find that 37.4% of the relevant documents in ClueWeb09B do not contain an explicit query entity. The largest source of missing entities in documents are those in Wikipedia. Missing entity links for Wikipedia accounts for 24.6% of the documents. The FACC1 annotations do not contain annotations for the majority of Wikipedia articles in ClueWeb09B. Of the relevant documents that contain at least one entity, 43% of these contain at least one mention of an explicit query entity. This indicates that 57% of the remaining relevant documents do not contain the explicit query entity and cannot be matched using this feature alone. The reasons for the mismatch is an area for future work. It is caused by both missing entity links as well as fundamental query-document mismatch.

5.4 Entity Analysis of queries

In this section we further study the entity characteristics of these datasets. How common are different classes of entities in the queries? We manually classify the presence of entities in the queries for all of the datasets.

The queries are labeled with three classes of entities. The first is the most general, whether the entity occurs in Freebase. The second whether it contains a named entity that would be detected by a typical entity recognition (NER) system. The last class of entities is narrower and is restricted to people, organizations, and locations. For each query we classify whether or not an entity of a particular class appears in the query. We do not examine the number of entities in the query or the centrality of the entity to the query.

The entity classification statistics are shown in Table 5. We observe that between 95% and 98% of the queries contain at least one mention of a Freebase entity. Many of the entities in the queries are general concepts, such as 'mammals', 'birth rates', 'organized crime', and 'dentistry'. For the web queries, approximately half the queries (54% and 52%) contain a named entity. The distribution of the types in the web queries is similar. A smaller percentage of queries for Robust04 contain named entities, only 34%. One reason for this is that web queries are more likely to contain brand names, actors, songs, and movies. Examples of these include 'Ron Howard,', 'I will survive', 'Nicolas Cage', 'Atari', 'Discovery Channel', 'ESPN', and 'Brooks Brothers'.

When the entities are restricted to people, organizations, and locations the fraction of queries containing entities decreases further. The fraction of entities that fall into this limited class is between 59% and 74% of the queries containing named entities overall. These entities belong to the "MISC" category and include diseases, songs, movies, naval vessels, drugs, nationalities, buildings, names of government projects, products, treaties, monetary currencies, and others. These appear to be common in queries and more emphasis should be placed on finer grained entity type classification.

5.5 Effectiveness on Robust04

In this section we describe an analysis of the effectiveness of the previously described entity query classes for the Robust04 dataset. We study the behavior of three retrieval models: sequential dependence model (SDM), multiple source expansion (MSE) [3], and entity-based feature expansion (EQFE). The results are shown in Table 6.

We observe that the EQFE expansion model is the best performing model across all classes of queries. We also note that queries that contain entities perform better for all retrieval models. The differences with queries containing Freebase entities are small, which is not surprising because most of the queries contain at least one entity. EQFE performs consistently better than the other models for all classes of queries.

The most interesting finding is the comparison of queries with named entities (NER). Queries containing named entities, but not restricted to PER/ORG/LOC show a difference over the other classes of queries. It demonstrates that the queries with 'MISC' entities perform better than other classes of entity queries for all models. The gains are the largest for this class of queries for EQFE compared with the baseline SDM retrieval model.

[4] http://ciir.cs.umass.edu/downloads/

6. RELATED WORK

6.1 Query Expansion

Query expansion techniques have been well studied for many models [22, 29]. Unlike most models, our approach goes beyond words or even features of words and includes features from entity links. The mostly closely related work is Latent Concept Expansion (LCE) model proposed by Metzler and Croft [25]. It builds upon the sequential dependence retrieval framework and introduces the idea of using arbitrary features for expansion. However, although a general framework is proposed they find improvements using only unigram features. Another well-known expansion model is Latent Concept Analysis from Xu and Croft [33], which selects 'concepts', limited to unigram and phrase features that co-occur near query terms in top ranked documents. The contribution of words versus phrases was not tested. In contrast, we use words, phrases, and structured entity attributes in EQFE to improve retrieval effectiveness.

6.2 Entity Retrieval

Using entities in retrieval is an area that has been well studied. In particular, the research area of retrieving entities has received significant recent attention. Entity retrieval was studied at the TREC entity retrieval track [2], at INEX with the entity ranking [12] and linked data tracks [32], the workshop on Entity Oriented and Semantic Search [1, 5], and other venues. In contrast, we focus on document retrieval leveraging entity annotations. Exploiting entity links and other types of semantic annotations is an area of open research. The workshop on Exploiting Semantic Annotations in Information Retrieval (ESAIR) [4, 18] has run over the last five years, and highlights the need for continued research in this area.

6.3 World Knowledge

Using Wikipedia as a source of world knowledge has been demonstrated to improve a variety of tasks, including retrieval. It is a common source of external query expansion [13, 3, 34]. Wikipedia entities as a basis for semantic representation demonstrated significant gains for a variety of NLP tasks. These tasks include semantic relatedness [14], document clustering [14], and entity linking [10]. We demonstrate that leveraging structured attributes of knowledge base entities similarly provides substantial gains in effectiveness for retrieval.

6.4 Entity Context Model

Building entity context models from their surrounding representation has been studied in the past. In 1994, Conrad and Utt [8] used all paragraphs in the corpus surrounding named entity mentions to represent the entity, allowing free text queries to find names associated with a query. Ten years later, Raghavan et al. [27] extended that idea to use language modeling as a representation and showed that these models could successfully be used to cluster, classify, or answer questions about entities. In these cases, the entity's context was a paragraph or a fixed number of words surrounding all mentions of the entity in the corpus. More recently, the work of Schlaefer et al. [30] expanded the representation of a Wikipedia entity using extracted "text nuggets" from the web for use in the Watson question answering system. Nuggets that were scored as relevant to the entity were used as its context, even if the nugget did not contain an actual mention.

Our entity context model (ECM) differs from existing work in three key ways. First, it uses state-of-the-art disambiguated entity links. If there are multiple ambiguous mentions of the same name, the contexts are separated based on their linked entity. Also, we do this for all types of concepts that exist in the knowledge base rather than just traditional named entities (person, organization, location).

Second, our context models are query focused. We construct an entity context model from documents retrieved in response to the query. This change is important for large corpora because for entities with multiple diverse topics a representation across the entire collection will blend these topics together and lose their distinguishing characteristics. For example, the ClueWeb09 query [obama family tree] focuses on aspects of Obama's family life and relationships to relatives, which is a relatively obscure topic when compared with more popular aspects such as "obamacare."

Finally, our approach captures not just words and phrases surrounding the mention, but structured annotations from co-occurring entities: their mentions and features of them, including types and categories. We also incorporate the uncertainty of extracted features, both the source relevance and entity link probability.

7. CONCLUSION

We have shown that features derived from linked entities can be used to improve the effectiveness of document retrieval. In qualitatively different collections (Robust04 and ClueWeb12B), the EQFE method was on average the strongest performer compared to several state-of-the-art baselines. These are some of the first reported results using the FACC1 Freebase annotations for ad hoc retrieval.

One limitation of this work is that it depends upon the success and accuracy of the entity annotations and linking. It would be useful to understand the accuracy and utility more robust detection of entities such as 'poverty', or 'term limits' rather than focusing primarily on people, organizations, and locations.

Our results are also affected by entities that are detectable but that are not in the knowledge base – e.g., 'fickle creek farms'. For these entities, there is no knowledge base entry to leverage, so the simplest solution is to consider only the unstructured word features. Lastly, we also described and successfully incorporated an entity context model that represents an entity by the language surrounding its mentions in the context of the query.

This work presents a first step leveraging large-scale knowledge resources that have become available in the last several years. We expect that as these knowledge resources mature that entity-based representations of both queries and documents will grow in importance, supporting increasingly complex information needs.

Acknowledgements

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.

8. REFERENCES

[1] K. Balog, D. Carmel, A. P. de Vries, D. M. Herzig, P. Mika, H. Roitman, R. Schenkel, P. Serdyukov, and T. T. Duc. The first joint international workshop on entity-oriented and semantic search (JIWES). In *ACM SIGIR Forum*, volume 46, pages 87–94. ACM, 2012.

[2] K. Balog, P. Serdyukov, and A. P. de Vries. Overview of the TREC 2011 entity track. In *Proceedings of the Text REtrieval Conference (TREC)*, 2011.

[3] M. Bendersky, D. Metzler, and W. B. Croft. Effective query formulation with multiple information sources. In *Proceedings of the fifth ACM international conference on Web*

search and data mining, WSDM '12, pages 443–452, New York, NY, USA, 2012. ACM.

[4] P. Bennett, E. Gabrilovich, J. Kamps, and J. Karlgren. Sixth Workshop on Exploiting Semantic Annotations in Information Retrieval (ESAIR'13). In *In Proceedings of CIKM '13*, pages 2543–2544, New York, NY, USA, 2013. ACM.

[5] R. Blanco, H. Halpin, D. M. Herzig, P. Mika, J. Pound, H. S. Thompson, and T. T. Duc. Entity search evaluation over structured web data. In *Proceedings of the 1st international workshop on entity-oriented search workshop (SIGIR 2011), ACM, New York*, 2011.

[6] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *In Proceedings of SIGMOD '08*, pages 1247–1250, New York, NY, USA, 2008. ACM.

[7] J. P. Callan, W. B. Croft, and J. Broglio. Trec and tipster experiments with inquery. *Information Processing & Management*, 31(3):327–343, 1995.

[8] J. G. Conrad and M. H. Utt. A system for discovering relationships by feature extraction from text databases. In *SIGIR'94*, pages 260–270. Springer, 1994.

[9] G. V. Cormack, M. D. Smucker, and C. L. A. Clarke. Efficient and Effective Spam Filtering and Re-ranking for Large Web Datasets, Apr. 2010.

[10] S. Cucerzan. TAC Entity Linking by Performing Full-document Entity Extraction and Disambiguation. In *Proceedings of the Text Analysis Conference 2011*, 2011.

[11] J. Dalton and L. Dietz. A Neighborhood Relevance Model for Entity Linking. In *Proceedings of the 10th International Conference in the RIAO series (OAIR)*, RIAO '13, New York, NY, USA, May 2013. ACM.

[12] G. Demartini, T. Iofciu, and A. P. De Vries. Overview of the INEX 2009 entity ranking track. In *Focused Retrieval and Evaluation*, pages 254–264. Springer, 2010.

[13] F. Diaz and D. Metzler. Improving the estimation of relevance models using large external corpora. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 154–161, New York, NY, USA, 2006. ACM.

[14] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of the 20th international joint conference on Artifical intelligence*, IJCAI'07, pages 1606–1611, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[15] 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), June 2013.

[16] D. W. Huang, Y. Xu, A. Trotman, and S. Geva. Focused access to XML documents. chapter Overview of INEX 2007 Link the Wiki Track, pages 373–387. Springer-Verlag, Berlin, Heidelberg, 2008.

[17] H. Ji, R. Grishman, and H. Dang. Overview of the TAC2011 knowledge base population track. In *Text Analysis Conference*, 2011.

[18] J. Kamps, J. Karlgren, and R. Schenkel. Report on the Third Workshop on Exploiting Semantic Annotations in Information Retrieval (ESAIR). *SIGIR Forum*, 45(1):33–41, May 2011.

[19] R. Kaptein, P. Serdyukov, and J. Kamps. Linking wikipedia to the web. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 839–840, New York, NY, USA, 2010. ACM.

[20] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[21] R. Krovetz. Viewing morphology as an inference process. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 191–202. ACM, 1993.

[22] V. Lavrenko and W. B. Croft. Relevance-Based Language Models. In *Proceedings of the ACM SIGIR 01 conference*, pages 120–127, 2001.

[23] A. Mccallum, K. Schultz, and S. Singh. Factorie: Probabilistic programming via imperatively defined factor graphs. In *In Advances in Neural Information Processing Systems 22*, pages 1249–1257, 2009.

[24] D. Metzler and W. B. Croft. A Markov random field model for term dependencies. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '05, pages 472–479, New York, NY, USA, 2005. ACM.

[25] D. Metzler and W. B. Croft. Latent concept expansion using markov random fields. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 311–318, New York, NY, USA, 2007. ACM.

[26] P. Mika and T. Potter. Metadata statistics for a large web corpus. In *Proceedings of the Linked Data Workshop (LDOW) at the International World Wide Web Conference*, 2012.

[27] H. Raghavan, J. Allan, and A. McCallum. An exploration of entity models, collective classification and relation description. In *KDD Workshop on Link Analysis and Group Detection*, pages 1–10, 2004.

[28] L. Ratinov, D. Roth, D. Downey, and M. Anderson. Local and global algorithms for disambiguation to wikipedia. In *ACL*, 2011.

[29] J. J. Rocchio. Relevance feedback in information retrieval. In G. Salton, editor, *The SMART Retrieval System: Experiments in Automatic Document Processing*, Prentice-Hall Series in Automatic Computation, chapter 14, pages 313–323. Prentice-Hall, Englewood Cliffs NJ, 1971.

[30] N. Schlaefer, J. C. Carroll, E. Nyberg, J. Fan, W. Zadrozny, and D. Ferrucci. Statistical source expansion for question answering. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pages 345–354, New York, NY, USA, 2011. ACM.

[31] V. I. Spitkovsky and A. X. Chang. A Cross-Lingual dictionary for english wikipedia concepts. In *Conference on Language Resources and Evaluation*, 2012.

[32] Q. Wang, J. Kamps, G. R. Camps, M. Marx, A. Schuth, M. Theobald, S. Gurajada, and A. Mishra. Overview of the INEX 2012 linked data track. In *INitiative for the Evaluation of XML Retrieval (INEX)*, 2011.

[33] J. Xu and W. B. Croft. Improving the effectiveness of information retrieval with local context analysis. *ACM Trans. Inf. Syst.*, 18(1):79–112, Jan. 2000.

[34] Y. Xu, G. J. F. Jones, and B. Wang. Query Dependent Pseudo-relevance Feedback Based on Wikipedia. In *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, pages 59–66, New York, NY, USA, 2009. ACM.