Document Retrieval Model Through Semantic Linking

Faezeh Ensan
Ferdowsi University of Mashhad
Mashhad, Iran
ensan@um.ac.ir

Ebrahim Bagheri
Ryerson University
Toronto, Canada
bagheri@ryerson.ca

ABSTRACT

This paper addresses the task of document retrieval based on the degree of document relatedness to the meanings of a query by presenting a semantic-enabled language model. Our model relies on the use of semantic linking systems for forming a graph representation of documents and queries, where nodes represent concepts extracted from documents and edges represent semantic relatedness between concepts. Based on this graph, our model adopts a probabilistic reasoning model for calculating the conditional probability of a query concept given values assigned to document concepts. We present an integration framework for interpolating other retrieval systems with the presented model in this paper. Our empirical experiments on a number of TREC collections show that the semantic retrieval has a synergetic impact on the results obtained through state of the art keyword-based approaches, and the consideration of semantic information obtained from entity linking on queries and documents can complement and enhance the performance of other retrieval models.

Keywords

Semantic Search, Information Retrieval, Language Models, Semantic Linking, Semantic Relatedness

CCS Concepts

•Information systems → Information retrieval; Retrieval models and ranking; Language models; Similarity measures; •Computing methodologies → Semantic networks;

1. INTRODUCTION

In the recent years, the language modelling approach for information retrieval has been widely studied and applied to different retrieval tasks due to its clearly-defined statistical foundations and good empirical performance [29]. The main idea is to estimate a language model θ_d for each document d and to rank documents based on the likelihood of generating the query using the estimated language models. In other words, for ranking document d, the following scoring method is employed:

$$Score(d, q) = P(q|\theta_d)$$

where, θ_d, the language model estimated for document d, is a probability distribution over all possible query units, and $P(q|\theta_d)$ denotes the probability of the query q according to the distribution θ_d. Clearly, one of the important steps is the estimation method for finding θ_d. Basic language modeling approaches primarily define the probability distribution based on the *exact* match of terms in the query and those in the documents as well as the collection of documents [29, 22]. Methods based on exact match of words have limitations such as *vocabulary gaps* between queries and documents, where users might choose query words that are different from those used in the documents for expressing similar meanings, or use the same words to refer to different meanings.

In order to address the vocabulary gap problem, several researchers such as [12, 3] have already proposed to model documents and queries as a set of words generated from a mixture of latent topics, where a latent topic is a probability distribution over the terms or a cluster of weighted terms, or in other work where the likelihood of translating a document to a query is estimated and is used for the purpose of ranking documents [15, 13]. In contrast, the focus of our work is to explore whether information obtained through the semantic entity linking of documents and queries can enhance the process of document retrieval.

To this end and based on our empirical review of the queries and documents in several of the TREC datasets, we have observed that there are many cases where the entity-based treatment of queries and documents can have synergetic impact on the results obtained through state of the art keyword-based approaches. Based on such observations, our hypothesis is that the consideration of semantic information obtained from entity linking on queries and documents can complement and enhance the performance of keyword-based retrieval models.

In this paper, we propose the Semantics-Enabled Language Model (SELM) for retrieving documents based on the degree of relatedness of the meaning of the query and the documents. In SELM, queries and documents are modeled as a set of semantic concepts obtained from running them through a entity linking system. Concepts, which are provided by entity linking systems, correspond to entities in a semantic network. Examples of entities generated by se-

WSDM 2017, February 06-10, 2017, Cambridge, United Kingdom

DOI: http://dx.doi.org/10.1145/3018661.3018692

mantic linking systems are semantically-related Wikipedia articles or entities in the DBpedia knowledge base. The core idea of our work is to find the conditional probability of generating the concepts observed in the query given all the document concepts and the relatedness relationships between them. In the recent years, the field of automated semantic annotation of textual content for extracting concepts and entities and linking them to external knowledge bases [23, 9], as well as the field of computing semantic similarities between knowledge base entities [26, 10] have been widely studied, and promising experimental performance has been reported. The language model presented in this paper is designed to be able to work with any such semantic annotation (entity tagging) and semantic similarity estimation system.

SELM models a document as an undirected graph where each node corresponds to a concept in the document and each edge represents a relatedness relationship between two concepts. In forming the graph, we assume two concepts are related if there is an edge between them and there is no dependency between two non-neighbouring concepts. Based on this graph, we adopt a probabilistic reasoning model based on conditional random fields for calculating the conditional probability of a query concept (as the output label) given values assigned to document concepts (as input nodes). SELM uses the conditional probabilities for forming the language model.

We will show in our work that SELM is able to identify a distinct set of documents as relevant that were not retrieved by state of retrieval models. In addition, we observed that there are cases where the retrievals of keyword-based models are not included in SELM. Therefore, the integration of SELM and keyword-based models would collectively yield and retrieve a larger set of relevant results. For this reason, we explore the possibility of interpolating SELM with other retrieval models, and show in our experiments that the interpolation of semantics-enabled model will significantly enhance the performance of keyword-based models by identify relevant documents that could not have been retrieved otherwise.

2. THE SELM MODEL

Based on the language model approach to information retrieval, we assume that a query q is generated from a document d by the probabilistic model θ_d. Here we are interested in estimating $P(q|\theta_d)$ for the purpose of scoring and ranking d. SELM provides an estimation for $\theta_d = \{P(q_i|d)\}_{i \in \lfloor 1,|Q| \rfloor}$, where $P(q_i|d)$ is the probability of query q_i and Q is the set of all query units. We ensure that $\sum_{i \in \lfloor 1,|Q| \rfloor} P(q_i|d) = 1$. In estimating the probability distribution, we adopt an undirected graphical model for calculating the conditional probability of a set of target variables, given the set of observed variables. In the context of our model, concepts of the query are modelled as the target variables and concepts of the document are modelled as the set of observed variables.

Our undirected graphical model is similar to CRFs that have been previously applied to different information retrieval tasks. In work such as [19], CRFs are used for modelling sequential data. In these works, it is assumed that the output is a sequence of labels, and input variables and their dependencies form a chain. In [27, 28], CRFs are used as a method for combining a diverse set of features. In these works, CRFs are trained over training datasets. The challenging aspect of existing work is to efficiently learn appropriate weights for different feature functions based on the available training data. In this paper, we do not restrict the input document concepts to form a chain. In fact, concepts in the document can form a graph in any arbitrary shape. In addition, in this paper, we attempt to build a generative language model contrary to the most dominant application of CRFs applied to discriminative problems. In the other words, we are not interested in learning the best weights for diverse features that converge to the maximum value over a training dataset, instead, given the semantic relatedness between the observed concepts, we are interested in finding the probability that a query concept is generated from a specific document.

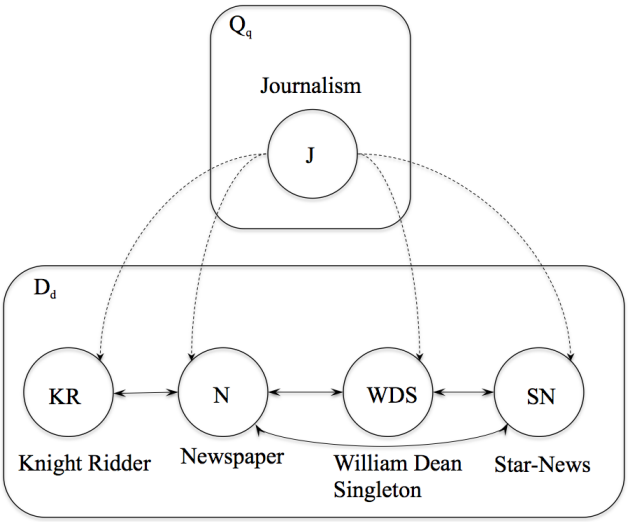


Figure 1: Sample query and document relationship model.

2.1 Illustrative Example

As an illustrative example, consider the query $q = \{$ Journalism$\}$ and the document d that is composed of the following paragraph, which is selected from Document LA082290-0094 of TREC CD5:

> Singleton, [...], bought the Star-News for $55 million from the Knight-Ridder newspaper chain in June 1989.

Figure 1 shows the representation of the query and the document based on their concepts and semantic relatedness relationships. As seen in the figure, four concepts 'Knight Ridder', 'William Dean Singleton ', 'Newspaper', and 'Star-News' have been spotted in the document. Also, the concept 'Journalism' has been found in the query. Dashed lines show semantic relatedness between the query concept and document concepts and solid lines represent semantic relatedness between document concepts. In this figure, concepts correspond to the Wikipedia articles with the same names and semantic relatedness are found using a semantic analysis system that estimates relatedness between Wikipedia entries.

This example highlights two main challenges of representing documents and queries based on their semantic concepts, which we address as follows:

1. Contrary to the bag of words model, where the prob-

ability of generating a query term given a document is estimated based on its occurrence in the document and in the collection, here we need to model semantic relatedness between query concepts and document concepts. We represent relatedness relations as probability dependencies. In our model, two semantically-related concepts are modelled as dependent neighbours and two not-semantically-related concepts are modelled as non-neighbouring nodes, which are independent given all other concepts. For forming this graph, our model relies on semantic analysis systems that measure semantic relatedness between concepts in documents. These systems usually provides semantic relatedness score for pairs of concepts, where those with a score more than a specific threshold are considered as semantically-related.

2. For a document of size n concepts, finding all semantic relatedness relationships is of order $O(n!)$. Given that such relatedness relations represent probability dependencies, finding the probability distribution over documents is quite complex and hardly possible for a big corpus. Our approach addresses this problem by avoiding finding the distribution over the input variables, hence it is a good choice for estimating the probability of output variables (query concepts), without worrying about the joint distribution of input variables (document concepts).

2.2 Proposed Model

Let $G = (V, E)$ be an undirected graph, where $V = D \bigcup Q$ and D is a set of document variables whose values are observed for any input document and Q is a set of query variables whose values need to be predicted by the model. Document and query variables correspond to concepts found and annotated in documents and queries, respectively. Document and query variables take binary values of (0,1), where the value of 1 indicates that the corresponding concept exists in a given document or query. The random variables are connected by undirected weighted edges, E, showing their degree of semantic relatedness. We denote an assignment to D by D_d, and an assignment to Q by Q_q. According to this model, a query concept Q_{q_j} is an assignment to Q in which the values of all variables except the j^{th} variable is zero. The value of the j^{th} element is 1. In this work, we assume that query concepts have no dependencies to each other. Hence, for a query $q = \{q_1, ..q_n\}$, $P(q|d)$=$\prod_{j=1}^{n} P(q_j|d)$. There are seminal works in the literature that consider dependencies between query terms in retrieval models [21]. Nonetheless, analyzing dependencies between query concepts is not the subject of this work and we leave it for future work.

In order to generate a ranking score for documents given a query term q_j, a scoring function needs to be defined based on the interpolation of two probabilities: the probability of the query given the document expressed as $P_{selm}(Q_{q_j}|D_d)$, and the probability of the query given the collection of all documents denoted by $P(Q_{q_j}|Col)$. The scoring function is formulated as:

$$\begin{aligned} Score_{selm}(d,q) &= P(Q_q|D_d) \\ &\simeq \sum_{j=1}^{|q|} logP(Q_{q_j}|D_d) \end{aligned} \quad (1)$$

where according to the Jelinek-Mercer [34] interpolation function, we have:

$$P(Q_{q_j}|D_d) = \begin{cases} (1-\lambda)P_{selm}(Q_{q_j}|D_d) + \lambda P(Q_{q_j}|Col) & similar\ concept\ found \\ \lambda P(Q_{q_j}|Col) & Otherwise \end{cases} \quad (2)$$

Based on this model, we wish to find $P_{selm}(Q_{q_j}|D_d)$, the probability of a given query concept based on a given document. According to [16], we have:

$$P_{selm}(Q_{q_j}|D_d) = \frac{1}{Z(D_d)} exp(\sum_{i=1}^{i=k} f_i(C_i, q_j, D_d)) \quad (3)$$

where $C_i \subseteq V$ is a clique over G and $C_i \not\subset D$, f_i is a feature function defined over C_i. $Z(d)$ is a normalization factor and is defined as:

$$Z(D_d) = \sum_j exp(\sum_{i=1}^{i=k} f_i(C_i, Q_{q_j}, D_d)) \quad (4)$$

Q has $|Q|$ different assignments in each of which a node has a value of 1 and the others have the value of 0. The partition function Z is the sum of the non-normalized probability for all of $|Q|$ possible query concepts. Based on our definition of feature functions, which we will introduce in the following paragraph, the value of $f_i(C_i, Q_{q_j}, D_d)$ is zero for those concepts in Q that are not semantically related to concepts of d. Given d has n concepts and each of them are maximally related to m query concepts, Z can be computed by the summation of at most $n \times m$ non-normalized probabilities.

Based on the query term independence assumption, there is no edge between the $|Q|$ query nodes. Hence, a C_i has exactly one node from Q. Considering this fact, we may have three types of features: 1) Features defined over document concepts, 2) Featured defined over a set that includes one query concept and an arbitrary number of document concepts, and finally 3) Features defined over a pair of a query concept and a document concept. The first set of features appear both in the non-normalized probability and Z in Equation 3, therefore, they will cancel each other out in the normalized probability. Therefore, we do not need to consider them for estimating the score measure. In this paper we also avoid calculating the second possible set of features because of its induced complexity and instead, we focus on the third set of features. It means that in our example in Figure 1, we do not define a feature over the set of {'Knight Ridder', 'Newspaper', 'Journalism }. Instead, we define two features {'Newspaper', 'Journalism } and {'Knight Ridder', 'Journalism }. Based on our assumptions, each C_i is a two-node clique that has one node from Q and one node from D that are connected through an edge, expressing that two corresponding concepts are semantically related to each other. Given $C_i = (x, y)$, $x \in D$, a node in the document space, $y \in Q$, a node in the Query space, and the value of x and y are assigned by d and q_j, the feature function f_i is defined as follows:

$$f_i(C_i, q_j, D_d) = \begin{cases} SemRel(x,y) & x_d = y_{q_j} = 1 \\ 0 & Otherwise \end{cases} \quad (5)$$

where $SemRel(x, y)$ is the value of semantic relatedness between two concepts associated with x and y. Now, the

probability of $P(Q_{q_j}|Col)$ is defined based on the document probabilities and collection statistics as follows:

$$P(Q_{q_j}|Col) = \frac{\sum_{d_i \in Col} P_{selm}(Q_{q_j}|D_d)}{|Col|} \quad (6)$$

It is now possible to form an integration of the proposed SELM model with traditional keyword-based language models.

2.3 Integration of SELM and Keyword-Based Retrieval Systems

As we will show later in the experimental results section, while SELM and other retrieval models can produce overlapping results, in many cases a subset of their relevant and correct results are distinct and non-overlapping. For this reason, the interpolation of these models can benefit from the correctly retrieved documents of each model and hence show improved performance. For integrating semantic-based and other language models we adopt the approach proposed in [4] with some modifications:

$$Score(d,q) = \lambda_{KW} Score_{KW}(d,q) + \lambda_{selm} Score_{selm}(d,q) \quad (7)$$

where $Score$ shows a normalized score [18] and $Score_{KW}(d,q)$ is a score obtained from a keyword-based language model. We use the EM algorithm to estimate mixture weights. For each query q, $\theta_q = \{\lambda_{\theta_{KW}}, \lambda_{\theta_{selm}}\}$, we have:

$$\theta_q^* = arg\max_{\theta_q} \log\left(\sum_{i=1}^{i=N} \lambda_{\theta_{KW}} Score_{KW}(d,q) + \lambda_{\theta_{selm}} Score_{selm}(d,q)\right) \quad (8)$$

where N is the total number of documents and $\lambda_{\theta_{KW}} + \lambda_{\theta_{selm}} = 1$. In order to estimate λ, the mixture weight for a given query q is computed as follows:

$$\lambda_{\theta_{KW}}^t = \frac{1}{N}\sum_{i=1}^{i=N} \frac{\lambda_{\theta_{KW}}^{t-1} Score_{KW}(d_i,q)}{\lambda_{\theta_{KW}}^{t-1} Score_{KW}(d_i,q) + \lambda_{\theta_{selm}}^{t-1} Score_{selm}(d_i,q)} \quad (9)$$

The mixture weight is calculated for each query separately, making it possible to assign different weights to semantic and keywords based models in retrieving different queries. To terminate the EM iterations, we set a threshold such that changes less than the threshold will stop the EM algorithm. In our experiments, we find that EM converges quickly usually converging in less than 5 iterations.

Before proceeding to the experimental results, let us provide an example of the impact of SELM and its interpolation with keyword-based models. As the example, for the Trec topic 340: 'Land Mine Ban', the state of the art techniques such as [21] would not be able to retrieve documents that do not explicitly include the keywords such as *land, land mine*, or *ban* but are relevant to the query from a content perspective, e.g., FBIS3-44701 is ranked 398 by [21] because it does not have the explicit query keywords while it is a relevant document to the query in the gold standard. However, SELM retrieves this document and ranks it in the first position. The interpolation of SELM+SDM proves to be effective in that this relevant document is ranked in position 9.

3. EXPERIMENTS

In this section, we describe experiments for analyzing the performance of the SELM Model and its impact on the performance of other existing retrieval methods.

3.1 Experimental Setup

For experiments, we adopted three widely used document collections: 1) TREC Robust04, 2) ClueWeb09-B (TREC Category B, which is the first 50 million English pages of the ClueWeb09 corpora), and 3) ClueWeb12-B (the TREC 2013 Category B subset of the ClueWeb12 corpora). Table 1 summarizes the datasets and the queries that were used in our experiments. Given the fact that there are no public entity annotations for the TREC Robust04 dataset, we performed automated annotation on this dataset. We chose to perform the annotations using the TAGME annotation engine. The choice of this annotation engine was motivated by a recent study reported in [6] that showed that TAGME was the best performing annotation system on a variety of document types such as Web pages and Tweets, which has publicly accessible RESTful API and available open source code. As a part of its results, TAGME provides a confidence value for each retrieved concept. We use TAGME's recommended confidence value of 0.1 for pruning unreliable annotations. For both ClueWeb09-B, and ClueWeb12-B, we use the same annotation engine to annotate the documents as was done for Robust04. As suggested in [7] due to limited computational resources, we do not entity link all documents in the document collections. Instead, we pool the top one hundred documents from all of the baseline text retrieval runs. The top 100 documents retrieved from all of our baselines along with their annotations as well as their runs and their evaluation metric results are made publicly accessible[1].

Collection	Documents	Topics
Robust04	528,155	301-450, 601-700
FACC1-09	50,220,423	1-200
ClueWeb09-B	50,220,423	1-200
ClueWeb12-B	52,343,021	1-50

Table 1: The TREC collections used in our experiments.

For indexing concepts identified in each document, we use their corresponding ConceptIDs, an integer number corresponding to the ID of a Wikipedia entry, as the key in Lucene. In terms of the required semantic relatedness values, we use TAGME relatedness service to compute pairwise concept semantic relatedness. The indexing step of our implementation has a second stage in which the normalization factor Z (Equation 4) is calculated and stored for each document. The normalization factor is calculated based on the degree of semantic relatedness between concepts of a document and all of the concepts of the collection. We use Jelinek-Mercer [34], the linear interpolation of the document

[1]https://github.com/SemanticLM/SELM

Collection	Model	MAP	Δ%	*p-value*	P@20	Δ%	*p-value*	nDCG@20	Δ%	*p-value*
Robust04	SDM	0.2615			0.3715			0.4235		
	SELM+SDM	0.2858†	+9.2	0.0001	0.3811	+2.5	0.1419	0.4405†	+4	0.0136
	RM3	0.2937			0.388			0.4341		
	SELM+RM3	0.31†	+5.5	0.0003	0.3986	+2.6	0.0577	0.4501†	+3.6	0.0061
	EQFE	0.3278			0.3797			0.4237		
	SELM+EQFE	0.3382†	+3	0.0197	0.3902	+2.7	0.1465	0.4353	+2.7	0.1233
ClueWeb09-B	SDM	0.1143			0.3412			0.21467		
	SELM+SDM	0.1183†	+3.4	0.0156	0.3495	+2.4	0.7	0.22793†	+6.1	0.006
	RM3	0.12			0.3447			0.22108		
	SELM+RM3	0.123	+2.5	0.0699	0.3477	+0.8	0.6	0.23411†	+5.9	0.006
	EQFE	0.1096			0.3184			0.2119		
	SELM+EQFE	0.117†	+6.7	0.0004	0.3298†	+3.5	0.0475	0.23078†	+8.9	0.0004
ClueWeb12-B	SDM	0.0421			0.209			0.12679		
	SELM+SDM	0.0446†	+5.1	0.002	0.221†	+5.7	0.0019	0.13407†	+5.6	0.0025
	RM3	0.0359			0.189			0.11098		
	SELM+RM3	0.038†	+5.5	0.0122	0.204†	+7.9	0.0001	0.11776†	+6.1	0.0042
	EQFE	0.0469			0.232			0.14633		
	SELM+EQFE	0.0493	+4.8	0.0535	0.234	+0.8	0.5	0.14981	+2.3	0.2

Table 2: Evaluation results for the interpolation of SELM with the three baseline methods. Statistical significance shown by †. Relative difference percentage and p-values from paired t-test shown as Δ% and *p-value*.

language model and the collection language model, with coefficient λ set to 0.1.

The queries that were used in the experiments are the title field of 250 Trec topics for Robust04, 200 Trec Web track topics for ClueWeb09-B, and 50 Web track topics for ClueWeb12-B. In our model, both queries and documents are required to be modeled as a set of concepts. For ClueWeb09-B queries, we use the Google FACC1 data that provides explicit annotations for the Web track queries. These annotations include descriptions and sub-topics, from which we use the description annotations. For Robust04 and ClueWeb12-B queries, there are no publicly available annotations. For our experiments, we employe TAGME with its confidence value set to 0.25 for annotating queries. We found a number of missing entities and also annotation errors in the results. As an example, Topic 654, 'same-sex schools', was annotated as 'Homosexuality', and 'Catholic School', which are definitely inconsistent. We manually revised these annotations to fix several errors. In this case, our revised annotations was the concept 'Single-sex education' for the topic number 654. All query annotations made by TAGME and also revisions are publicly available in the earlier mentioned Git repo.

In SELM, each query concept has a similarity threshold $0 < \alpha < 1$, such that all similarities less than α are pruned (i.e., concepts with similarities less than α, considered as unrelated to the query concepts). α is determined using 10-fold cross-validation and is optimized for Mean Average Precision (MAP) effectiveness.

3.2 Baselines

For the sake of comparison, we chose the Sequential Dependence Model (SDM)[21], which is a state-of-the-art retrieval model based on Markov Random Field that assumes dependencies between query terms. In addition, we compare SELM with two query expansion models: a variant of Relevance Model (RM3) [17], and Entity Query Feature Expansion (EQFE) [7]. RM3 extracts the relevant terms and uses them in a combination with the original query. RM3 is known to improve the retrieval performance over methods that do not use expansion terms. EQFE is an expansion method that enriches the query with features extracted from entities found in queries, entity links to knowledge bases, and the entity context. It has already been shown [7] that EQFE improves retrieval performance significantly over the state-of-the-art methods. In this paper, and to keep our experiment comparable to these methods, we used the parameter settings reported in [17, 7] for the baseline methods. SELM is interpolated with these three baseline systems based on Equation 7 in order to form three variations, referred to as SELM+SDM, SELM+RM3, and SELM+EQFE.

3.3 Results

In this section we report the performance of SELM and its interpolation with baseline methods. For each collection, we report the Mean Average Precision (MAP), Precision at rank 20 (P@20), and normalized Discounted Cumulative Gain at rank 20 (nDCG@20). The statistical significance of differences in the performance of SELM models with respect to other retrieval methods is determined using a Paired t-test with a confidence level of 5%. For evaluating ClueWeb09-B and ClueWeb12-B, the relevance judgments of the whole corpus have been used.

3.3.1 SELM Interpolation Effectiveness

Table 2 presents the evaluation results on the three datasets. The interpolation of SELM with all baselines improves their performance. SELM+SDM outperforms SDM significantly across two measures MAP and nDCG@20 on all datasets (up to +9.2% MAP and +6.1% nDCG@20). Also, SELM+SDM improves P@20 compared to SDM over Robust04, ClueWeb09-B and outperforms SDM significantly over ClueWeb12-B (up to +5.7% P@20). SELM+RM3 outperforms RM3 across all measures on all datasets (up to +5.5% MAP, +6.1% nDCG@20, and +7.9% P@20). The improvements are statistically significant on P@20 over ClueWeb12-B, MAP over Robust04 and ClueWeb12-B, and on nDCG@20 on all datasets. SELM+EQFE outperforms EQFE on all metrics for all datasets and the observed improvements are statistically significant for ClueWeb09-B.

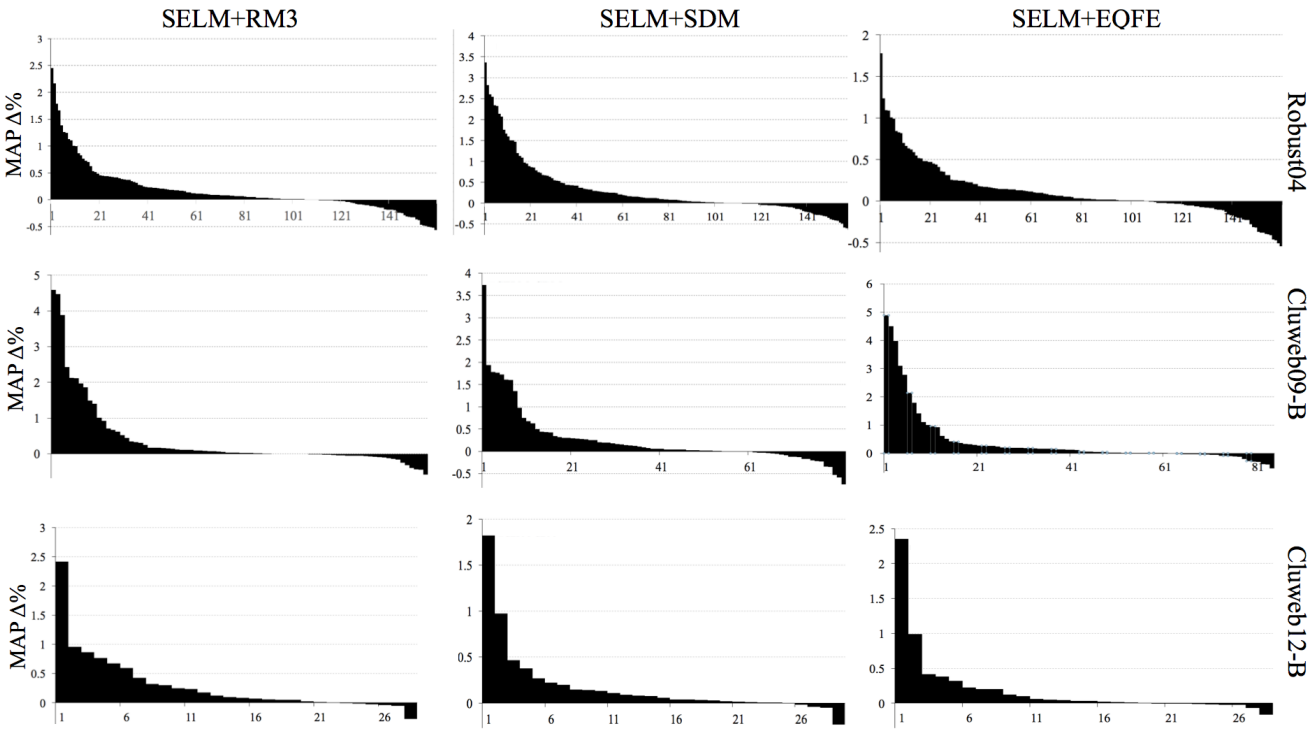


Figure 2: MAP Δ% of interpolated SELM & baselines (e.g., SELM+SDM vs SDM). Positives show improvement over baseline.

3.3.2 *Success/Failure Analysis*

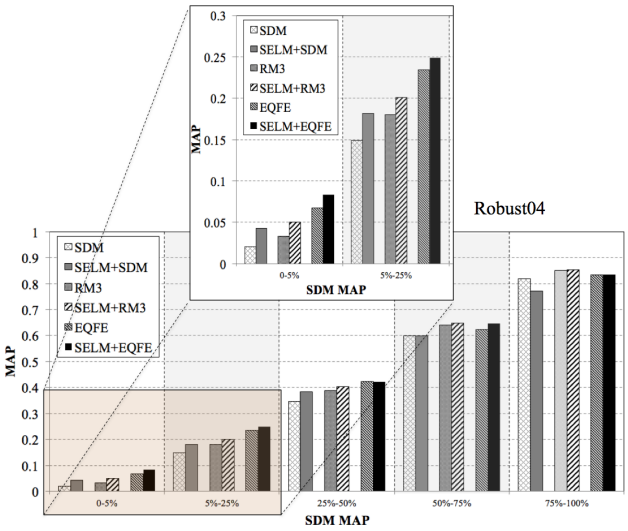
Figure 2 provides analysis of queries whose effectiveness are improved/hurt by the variants of the SELM method. In these figures, the relative percentage improvement of MAP for SELM+SDM over SDM, SELM+RM3 over RM3, and SELM+EQFE over EQFE is reported. Given the fact that SELM returns no results for queries with no concepts, we only consider the queries that have at least one concept annotation, which is equal to 163 queries for Robust04, 94 and 34 for ClueWeb09-B and ClueWeb12-B, respectively. As outlined in Table 3, out of the 163 queries for the Robust04 dataset, SELM+SDM helps 115, SELM+RM3 helps 113, and SELM+EQFE helps 107 of the queries. In ClueWeb09-B and for the 94 queries, SELM+SDM helps 59, SELM+RM3 helps 50, and SELM+EQFE helps 62 queries. For ClueWeb12-B and the associated 34 queries, SELM+SDM helps 25, SELM+RM3 helps 23, and SELM+EQFE helps 19 queries. All the help/hurts were determined by comparing the relative difference percentage of MAP of an interpolated SELM method compared to its respective baseline. SELM+SDM is the method that has seen a high improvement in terms of the number of helped queries. The reason can be due to the fact that SDM, contrary to RM3 and EFQE, is a method that has not been augmented by expansions from documents or knowledge base data and links. Hence it can benefit the most when combined with the semantic perspective that is offered by SELM.

We also analyze SELM variants with regards to their effect on a range of easy to difficult queries. For this analysis, we divide queries into buckets of MAP ranges according to their MAP from the SDM baseline. Queries that have larger SDM MAPs are considered to be easier queries compared to the ones that have a lower SDM MAP, which are those that we will consider to be more difficult. Figure 3 illustrates this analysis. The figure has three parts for each of the document collections. In the figure, a SELM variant is paired with its associated baseline, e.g. SELM+SDM and SDM, to show how much improvement was obtained as a result of the interpolation. In addition, we have provided a zoomed-in view of the results for the most difficult queries in order to be able to clearly depict the improvement made on such queries. This analysis shows that SELM is effective in improving the more difficult queries. For Robust04, all queries except the easiest queries (queries whose SDM MAP are between 75% and 100%) are improved by all SELM interpolated methods compared to their respective baselines. In ClueWeb09-B all difficult queries (MAP<50%) have been improved and the specially more difficult queries (MAP<25% as shown in the zoom) have received noticeable improvement. SELM performed well on the ClueWeb12-B collection, where all of the queries, specially the difficult queries, were improved.

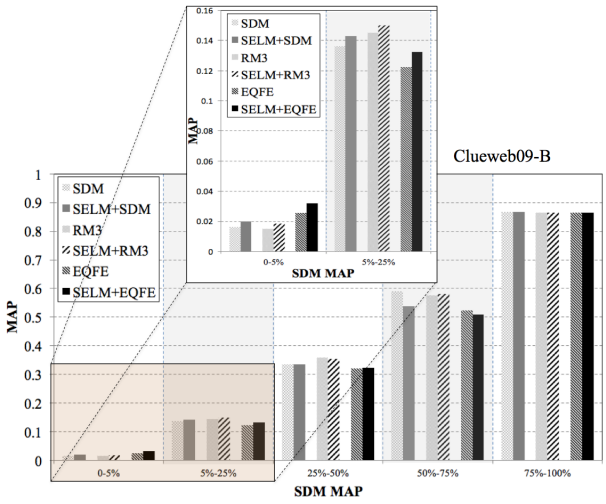
	R'04	CW'09	CW'12
SELM+SDM vs SDM	115	59	25
SELM+RM3 vs RM3	113	50	23
SELM+EQFE vs EQFE	107	62	19

Table 3: The number of queries helped by SELM variants.

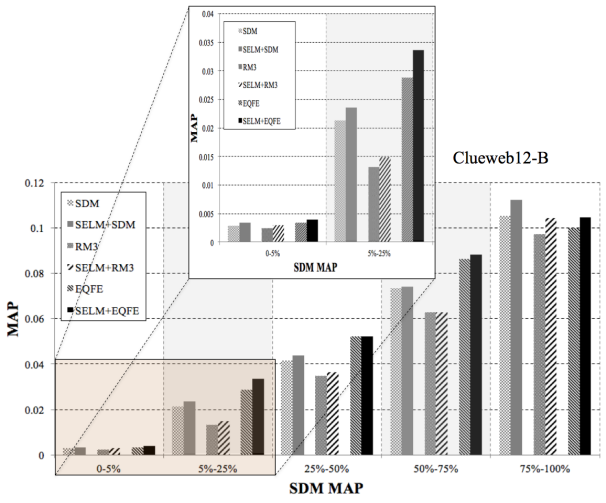
Table 4 shows that the interpolation of SELM with baselines outperforms the baselines across all measures for the most difficult queries. We considered queries whose MAP value for the SDM baseline is less than 0.05 to be the *most difficult* queries. As an instance, web query #92 ('the wall'), is a difficult query for the keyword-based system (SDM MAP= 0.0009). Keyword based query expansion cannot help much (RM3 MAP= 0.0009). Even EQFE (EQFE MAP= 0.0008),



(a)



(b)



(c)

Figure 3: Mean retrieval effectiveness across different query-difficulties, measured according to the percentile of SDM.

which uses semantic knowledge for query expansion is far behind SELM (SELM+SDM MAP= 0.0234, SELM+RM3 MAP= 0.0231, and SELM+EQFE MAP= 0.14). SELM works with query annotations (The Pink Floyd album named 'the wall') for retrieval, which helps SELM to search within documents that have concepts related to music, rock bands, and Pick Floyd. Hence, SELM has a better chance of finding related documents. On the other hand, SELM faces difficulties when dealing with search queries that are annotated with general concepts. As an example, none of the SELM interpolations produce effective results for the web query #44('map of the united states'). This is because the query is annotated with one concept only, i.e. United States, which is a very general concept with relationships to a lot of unrelated entities irrelevant to the topic of the query. As another example, web query #142 ('Illinois state tax') produces poor results when processed by SELM variants. This query is annotated with only one concept (Illinois), which is a general concept with a lot of diverse relationships, and at the same time does not cover the main topic of the query, which is taxes. We hypothesize that more effective query annotation techniques that are able to find both *relevant* and *specific* concepts that relate to the core topic of the query will help improve SELM. There is a progressive body of work in the literature that focus on query analysis and segmentation [11, 24]. We leave verification of this hypothesis and application of the query analysis literature to our future work.

3.3.3 *Analysis of Interpolation Success*

The main premise of our work was that the semantics-enabled model would retrieve documents that would not be otherwise retrieved by the other models. This has been empirically shown in Figure 4. The three sub-figures show the comparative analysis of the retrieval of distinct relevant documents retrieved by SELM compared to the other methods. As seen, for all three datasets, SELM retrieves a significant number of relevant documents that are missed by the other methods (shown in the Venn diagrams). The bar charts show the number of distinct non-overlapping relevant documents retrieved by SELM that have not been observed in any of the other approaches within the top-10 results (the x-axis shows queries and is ordered descendingly.). This shows how SELM is effective in the retrieval process and why its integration improves the overall performance.

4. RELATED WORK

The followings are the closely related research directions to our work.

Knowledge based retrieval: Exploiting general or domain-specific knowledge in retrieval has been extensively studied in the literature. Vallet et al. [32] propose using knowledge that is formally represented in domain ontologies for enhancing domain specific search. In their approach, a free text query is translated to RDQL, a query language for RDF, and posed over a formally represented domain ontology. A related document is one that is annotated with instances of the result tuples. The amount and quality of information that is modeled within the ontology limits the performance of ontology-based retrieval systems. On the other hand, Wikipedia and Freebase are two comprehensive sources of general world knowledge that are used as alternatives to domain-specific ontologies. In [31], the authors

Robust04	Difficult Queries (Map: 0-5%), Number of Queries: 42								
	Map	Δ	p-value	P@20	Δ	p-value	nDCG	Δ	p-value
SDM	0.019622	111%†	0.008	0.081707	20%	0.089	0.089657	34%†	0.080
SELM+SDM	0.041539			0.09878			0.120352		
RM3	0.03099	55%†	0.030	0.070732	45%†	0.0296	0.07783	44%†	0.048
SELM+RM3	0.048041			0.102439			0.111761		
EQFE	0.063732	24%	0.07	0.095122	21%†	0.033	0.098387	29%	0.1
SELM+EQFE	0.079451			0.115854			0.127266		
ClueWeb-09	Difficult Queries (Map: 0-5%), Number of Queries: 85								
	Map	Δ	p-value	P@20	Δ	p-value	nDCG	Δ	p-value
SDM	0.016476	22%†	0.0391	0.077976	9%	0.1	0.051288	18%†	0.0355
SELM+SDM	0.020125			0.084524			0.060741		
RM3	0.015446	20%	0.05	0.069048	15%†	0.021	0.044528	22%†	0.025
SELM+RM3	0.01866			0.079762			0.054653		
EQFE	0.025824	28%†	0.0309	0.1125	7%	0.07	0.084446	18%†	0.0076
SELM+EQFE	0.032352			0.120238			0.099977		
ClueWeb-12	Difficult Queries (Map: 0-5%), Number of Queries: 34								
	Map	Δ	p-value	P@20	Δ	p-value	nDCG	Δ	p-value
SDM	0.01850	6.7%	0.1	0.0893	20%†	0.017	0.05337	6%	0.1
SELM+SDM	0.01975			0.100			0.05658		
RM3	0.01314	7%†	0.022	0.07273	11%†	0.02	0.03710	8%†	0.04
SELM+RM3	0.01410			0.08030			0.04023		
EQFE	0.02376	11%	0.2	0.12576	6%	0.09	0.0736	3.2%	0.4
SELM+EQFE	0.02576			0.13333			0.07615		

Table 4: Comparison of the retrieval models on the most difficult queries (SDM MAP$<$ 5%). † shows statistical significance using a paired t-test ($\alpha < 0.05$).

present methods for indexing documents with Wikipedia concepts and representing documents with bag of concepts. These concepts are inter-lingual, hence can be used for cross-lingual retrievals. Similarly, [8] provides a bag of concept representation for documents based on the notion of concept vectors from Explicit Semantic Analysis (ESA). This work embeds a set of feature selection methods into its retrieval process in order to handle the noisy nature of the concept representation. Both [31] and [8] use ESA representation of concepts for the purpose of concept ranking and retrieval. Contrary to [8] and [31], our work is not attached to a specific knowledge representation framework and can work with any semantic annotation and analysis system. In [33] and [7], Freebase and Wikipedia are used for expanding query terms. In these methods, object descriptions and category classifications are used among other information resources for enriching queries. We used [7] as one our baselines and compared its performance with variants of SELM.

Entity retrieval: One of the emerging research topic is retrieving entities from documents. In [14], Wikipedia is used as a pivot for searching, and Wikipedia categories and their relations are used as the main source for entity retrieval. Zhiltsov et al. [35] propose to generalize the sequential dependence model for structured documents such as DBPedia. In their model, a mixture of language models is employed for retrieving entities that are represented in a five-field scheme, which is designed for DBpedia entities. In [25], user logs are analyzed for finding implicit user feedback in the the context of entity search. The other impressive works in this area include but are not limited to [1, 5]. The main focus of all these works, which is returning an entity or a list of entities for user queries, is different from the research goal of our work, which is the utilization of knowledge represented in knowledge bases, such as Wikipedia, for document retrieval.

Semantic Annotation and Entity Linking The process of identifying key phrases in texts and queries and linking them to entities of a knowledge base, is an active research field that has witnessed a good progress both in industry and academia in recent years. In [30], a comprehensive review and analysis of the main entity linking systems and their applications is presented. Cornolti et al. [6] provide a benchmarking framework for comparing annotation systems and analyse some of the popular systems such as TagMe [9], Wikipedia-miner [23], and DBpedia Spotlight [20]. In [2], a probabilistic model for linking queries to entities in a knowledge base is proposed that uses anchor text within Wikipedia and also analyzes user clicks on the result pages of web queries. The main focus of this research is time and space efficiency, and it claims that it reaches its goal by processing queries much faster than existing systems by using state-of-the-art hashing and encoding techniques. Our work in this paper benefited from progress in semantic linking systems, as it builds a concept representation of documents and queries as its initial phase.

5. CONCLUDING REMARKS

In this paper, we have proposed a semantics-enabled language model that represents documents and queries through a graph of concepts, where the relatedness of a query to a given document is calculated based on the semantic relatedness of their concepts. Our empirical evaluations show that our proposed model can complement and enhance the performance of keyword-based systems and its interpolation with other retrieval models can significantly improve their performance from the perspective of various IR measures.

As future work, we are interested in exploring two distinct directions. First, we are currently operating on the query term independence hypothesis, which may not be always realistic. We will be considering the possibility of incorporating query term dependence as a part of SELM. Furthermore, our empirical studies showed that while SELM variants are able to improve specific queries, they will not perform better than the baseline when the queries are annotated with

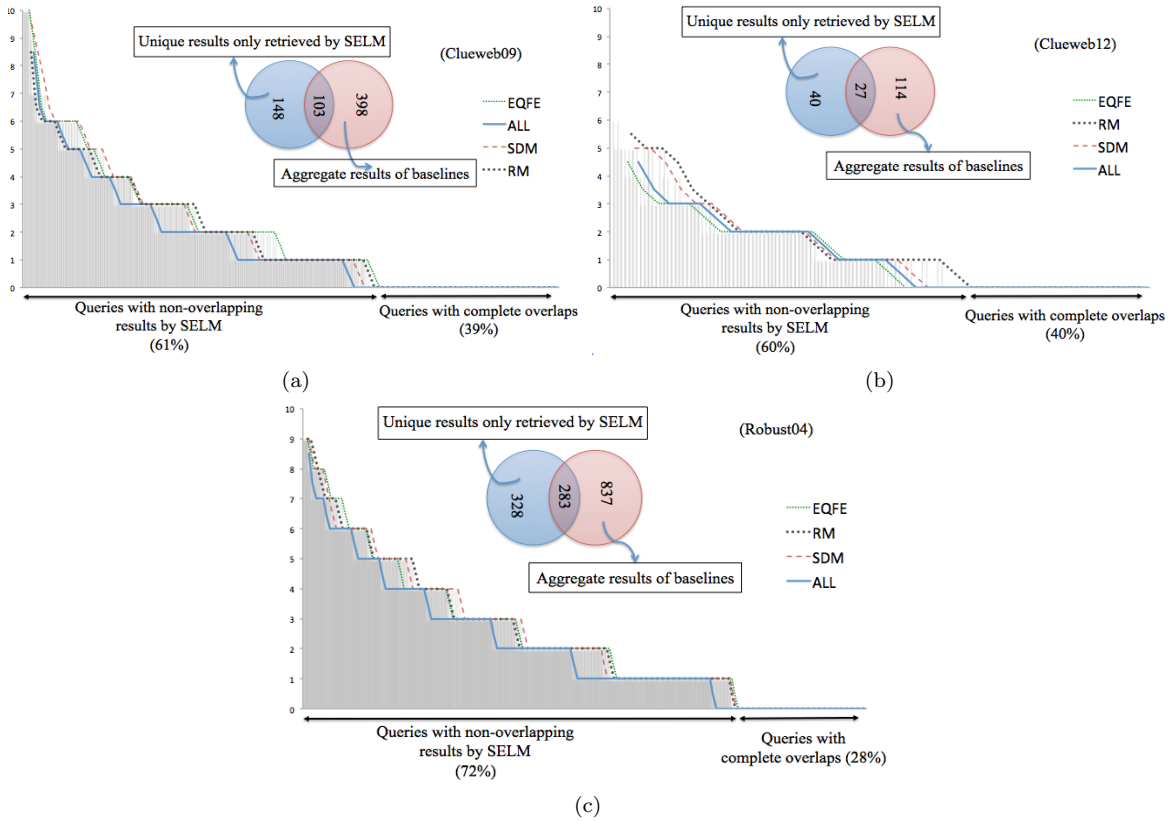


Figure 4: Comparison of the distinct results of SELM compared to the other methods.

general concepts that are not the main focus of queries. In our future work, we are interested in exploring possible ways such as applying query preprocessing, and query intent analysis to improve the performance of SELM when such queries are observed.

6. REFERENCES

[1] B. Billerbeck, G. Demartini, C. Firan, T. Iofciu, and R. Krestel. Exploiting click-through data for entity retrieval. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 803–804. ACM, 2010.

[2] R. Blanco, G. Ottaviano, and E. Meij. Fast and space-efficient entity linking for queries. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pages 179–188. ACM, 2015.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[4] G. Cao, J.-Y. Nie, and J. Bai. Integrating word relationships into language models. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 298–305. ACM, 2005.

[5] T. Cheng, X. Yan, and K. C.-C. Chang. Entityrank: searching entities directly and holistically. In *Proceedings of the 33rd international conference on Very large data bases*, pages 387–398. VLDB Endowment, 2007.

[6] M. Cornolti, P. Ferragina, and M. Ciaramita. A framework for benchmarking entity-annotation systems. In *Proceedings of the 22nd international conference on World Wide Web*, pages 249–260. ACM, 2013.

[7] J. Dalton, L. Dietz, and J. Allan. Entity query feature expansion using knowledge base links. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 365–374. ACM, 2014.

[8] O. Egozi, S. Markovitch, and E. Gabrilovich. Concept-based information retrieval using explicit semantic analysis. *ACM Transactions on Information Systems (TOIS)*, 29(2):8, 2011.

[9] P. Ferragina and U. Scaiella. Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1625–1628. ACM, 2010.

[10] 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.

[11] J. Guo, G. Xu, X. Cheng, and H. Li. Named entity recognition in query. In *Proceedings of the 32nd international ACM SIGIR conference on Research and*

development in information retrieval, pages 267–274. ACM, 2009.

[12] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM, 1999.

[13] R. Jin, A. G. Hauptmann, and C. X. Zhai. Language model for information retrieval. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 42–48. ACM, 2002.

[14] R. Kaptein, P. Serdyukov, A. De Vries, and J. Kamps. Entity ranking using wikipedia as a pivot. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 69–78. ACM, 2010.

[15] M. Karimzadehgan and C. Zhai. Estimation of statistical translation models based on mutual information for ad hoc information retrieval. In *Proceedings of the 33rd ACM SIGIR*, pages 323–330, 2010.

[16] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth ICML*, pages 282–289, 2001.

[17] V. Lavrenko and W. B. Croft. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 120–127. ACM, 2001.

[18] J. H. Lee. Analyses of multiple evidence combination. In *ACM SIGIR Forum*, volume 31, pages 267–276, 1997.

[19] A. McCallum, K. Bellare, and F. Pereira. A conditional random field for discriminatively-trained finite-state string edit distance. *arXiv:1207.1406*, 2012.

[20] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, pages 1–8, 2011.

[21] 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*, pages 472–479. ACM, 2005.

[22] D. R. Miller, T. Leek, and R. M. Schwartz. A hidden markov model information retrieval system. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 214–221. ACM, 1999.

[23] D. Milne and I. H. Witten. An open-source toolkit for mining wikipedia. *Artificial Intelligence*, 194:222–239, 2013.

[24] N. Mishra, R. Saha Roy, N. Ganguly, S. Laxman, and M. Choudhury. Unsupervised query segmentation using only query logs. In *Proceedings of the 20th international conference companion on World wide web*, pages 91–92. ACM, 2011.

[25] D. Mottin, T. Palpanas, and Y. Velegrakis. Entity ranking using click-log information. *Intelligent Data Analysis*, 17(5):837–856, 2013.

[26] Y. Ni, Q. K. Xu, F. Cao, Y. Mass, D. Sheinwald, H. J. Zhu, and S. S. Cao. Semantic documents relatedness using concept graph representation. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, pages 635–644. ACM, 2016.

[27] F. Peng and A. McCallum. Information extraction from research papers using conditional random fields. *Information processing & management*, 42(4):963–979, 2006.

[28] D. Pinto, A. McCallum, X. Wei, and W. B. Croft. Table extraction using conditional random fields. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 235–242. ACM, 2003.

[29] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281. ACM, 1998.

[30] W. Shen, J. Wang, and J. Han. Entity linking with a knowledge base: Issues, techniques, and solutions. *Knowledge and Data Engineering, IEEE Transactions on*, 27(2):443–460, 2015.

[31] P. Sorg and P. Cimiano. Exploiting wikipedia for cross-lingual and multilingual information retrieval. *Data Knowl. Eng.*, 74:26–45, Apr. 2012.

[32] D. Vallet, M. Fernández, and P. Castells. An ontology-based information retrieval model. In *The Semantic Web: Research and Applications*, pages 455–470. Springer, 2005.

[33] C. Xiong and J. Callan. Query expansion with freebase. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval*, pages 111–120, 2015.

[34] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 334–342. ACM, 2001.

[35] N. Zhiltsov, A. Kotov, and F. Nikolaev. Fielded sequential dependence model for ad-hoc entity retrieval in the web of data. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 253–262. ACM, 2015.