Across-Document Neighborhood Expansion: UMass at TAC KBP 2012 Entity Linking

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

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

Abstract

Last year's competition demonstrated that the NER context contains important information that should not be ignored in entity linking. State-of-the-art approaches anchor on unambiguous entities, look for overlap in categories, or approximate a joint model of candidate assignments, *after* Wikipedia candidates have been selected. Current candidate approaches, such as anchor text maps, are effective but may lead to very large candidate sets to be examined. UMass has two objectives for our TAC submission. First, we use cross-document context information to perform entity neighborhood expansion and estimate the importance of entity context using corpus-wide information. Second, we use probabilistic information retrieval that incorporates the neighborhood information to generate a ranked candidate set in a single step. The result is a small candidate set that even for less than 50 candidates contains the true answer in 95% of the cases, allowing for computationally intensive inference in the next phase. It turns out that our best performing run simply predicts the top candidate of the unsupervised candidate ranking, outperforming more than half of the contestants.

1 Introduction

A typical TAC KBP 2011 entity linking system has five steps: 1) query expansion, 2) candidate generation, 3) candidate ranking, 4) NIL detection, and 5) NIL clustering. The goal of the first two steps is to achieve a high-recall set of Wikipedia entities. However, if the query mention is highly ambiguous, the set of candidates can be very large with potentially thousands of candidates to rank. Given a candidate set, the most effective models use the surrounding entities in the document as disambiguating evidence (Monahan et al., 2011; Cucerzan, 2011; Ratinov et al., 2011). Our system differs from the traditional approach by considering the surrounding entities already in the candidate generation phase.

A danger of using contextual NER spans from the neighborhood is that the context contains spurious and misleading NER spans. In some cases, the document even focuses on a different subject. For example, consider a document about Australia with the sentence "ABC shot the TV drama Lost in Australia." where the task is to link "ABC" to the entity American Broadcast Central. In this example, the neighboring NER span "Australia" may lead to the incorrect conclusion that "ABC" refers to Australian Broadcasting Corporation. In contrast, the named entity "Lost" as well as the phrase "shot TV drama" provide helpful disambiguation context.

The goal of Neighborhood Expansion is to identify NER spans which are helpful for disambiguation, before candidate sets are retrieved. We use pseudo-relevance feedback (Xu and Croft, 1996), a technique from information retrieval, to find documents from the TAC source corpus that help to determine a reliable set of contextual NER spans.

Notice, that this task different from measuring ambiguity (Monahan et al., 2011): An NER span can be unambiguous, such as "Australia," but still be misleading context for disambiguation.

Outline. After considering MRF-based retrieval models in Section 2, we introduce Neighborhood Expansion which is based on pseudo-relevance feedback in Section 3. We detail the entity linking system in Section 4 and give results on evaluation data from 2011 and 2012 in Section 5.

2 Probabilistic Retrieval

To efficiently identify relevant Wikipedia and TAC source document, we build upon the Markov Random Field model for Information Retrieval (Lavrenko and Croft, 2001; Metzler and Croft, 2005). The query model scores the documents in the corpus using a log-linear weighted combination of language model probabilities of multiword concepts. The probabilities themselves can be gov-

```
#combine:0=(λ_T + λ_V):1=λ_S:2=λ_E(
  #combine:0=λ_T:1=λ_V(
    #seqdep(t)
    #combine(#seqdep(v_0) ... #seqdep(v_V))
  )
  #combine(#seqdep(s_0), ..., #seqdep(s_S))
  #combine:0 = φ_0^E : ... k : φ_k^E(
    #seqdep(e_0), ..., #seqdep(e_k)
  )
)
```

Figure 1: Query for retrieving relevant stream documents in Galago query syntax.

erned by a query model, allowing for arbitrary composition of unigram and sequential dependence models.

We include four types a of concepts with corresponding weights λ_A in the query: the mention text t, a set of name variants $\vec{v}$, context sentences $\vec{s}$, and a set of neighboring NER spans $\vec{e}$. For each document d in the collection, the score $f(d)$ is given by the proportionality in Equation 1, with type-based weights λ_T, λ_V, λ_S, and λ_E, concept-based weights $\vec{\phi}$, and ψ which is a real-valued log-score of the concept under the document's language model.

$$f(d) \propto \exp \left\{ \sum_{a \in \{t,v,s,e\}} \lambda_A \frac{1}{|\vec{a}|} \sum_i \phi_i^A \psi(d, a_i) \right\} \quad (1)$$

Concept-based weights $\vec{\phi}$ which are assumed to be uniform if omitted, and are re-normalized to form a multinomial distribution.

In this work, we use sequential dependence language models (Metzler and Croft, 2005) for ψ, which incorporate word, phrase, and proximity from adjacent concept words.

To execute the queries, we use the open source retrieval engine Galago (Strohman, 2007),[1] which is part of the Lemur project. The model from Equation 1 can be expressed using the Galago query language as specified in Figure 1.

3 Neighborhood Expansion with Pseudo-Relevance Feedback

In this section we describe our models for constructing reliable NER context for a query mention using pseudo-relevance feedback.

3.1 Query Document Analysis

We analyze the enclosing query document in order to identify three sources of information: a) name variants, b) contextual sentences, and c) the NER neighborhood. The query document is analyzed with NLP packages from UMass's factorie (McCallum et al., 2009) and Stanford CoreNLP (Finkel et al., 2005) to identify NER spans, within-document coreference chains, and sentence boundaries.

We extract name variants from the coreference chains, dropping mentions that do not include noun phrases. Because coreference systems are usually designed for high precision settings, we found them to be often too restrictive to capture all name variants. Therefore, we further include NER spans and capitalized word sequences that contain the query string (ignoring capitalization and punctuation for the matching).

For a fixed number of mentions (preferring strict matches) the surrounding sentence is taken into account. After removing stopwords, casing and punctuation they represent non-NER context such as verbs, adjectives, and multi-word phrases.

NER spans are sorted by proximity in character offsets to the query mention or one of its coreferent mentions and take the k closest as the NER neighborhood for the query.

3.2 Neighboorhood Weighting

Our intuition is that an ideal entity candidate would include all of the identified sources. However, a preliminary study has shown that directly adding the k closest NER spans leads to worse results on average. This is because the surrounding NER spans are not always reliable disambiguation context. As mentioned before, unambiguous spans are not necessarily reliable for disambiguation. Rather, an NER span is reliable if it occurs frequently in the context of the query mention, across other documents.

We identify the reliability of NERs with pseudo-relevance feedback (Xu and Croft, 1996; Metzler and Croft, 2007): We retrieve TAC source documents that maximize a combined score of query mention, name variances, and contextual sentences using the Galago retrieval engine.

The approach is based on the assumption that these pseudo-relevant documents are actually about the target entity. If an NER in the query document is unreliable, it will only be contained in few or none of the pseudo-relevant documents. If it represents reliable disambiguation context, it shall occur in many documents of the retrieved set.

Pseudo-relevant documents are retrieved by the search query given in Figure 1, with the modification that NER

[1] http://www.lemurproject.org/galago.php

spans e are not included.

For each pseudo-relevant document d, the probability that it is relevant to the TAC query is quantified by the retrieval probability $p(d|t, \vec{v}, \vec{s})$. We introduce a Bernoulli variable, which expresses whether the document d includes a given NER span e. For each NER span e, the probability ϕ_e^E of it being reliable is obtained by marginalizing over the retrieved set of pseudo-relevant documents D.

$$\phi_e^E = p(e|t, \vec{v}, \vec{s}) \propto \sum_{d \in D} p(e \in d|d) \cdot p(d|t, \vec{v}, \vec{s}) \quad (2)$$

In other words, the reliability of an NER span is expressed by accumulating retrieval probabilities of documents that contain the span.

3.3 Pseudo-Neighborhood

As an alternative for just re-weighing NER spans with pseudo-relevance, we experiment with including new NER spans from the pseudo-relevant documents a prerequisite to the weighting scheme of Equation 2.

All retrieved pseudo-relevant documents are analyzed with NLP methods as described in Subsection 3.1. The name variants $\vec{v}$ identified from the query document are used to search for potential coreferent mentions in the pseudo-relevant documents—we call them pseudo-coreferent mentions.

For each pseudo-relevant document, a set of k NER spans closest to any pseudo-coreferent mention is extracted. The union of closest NER spans across all pseudo-relevant documents and the query document is used as input to the reliability analysis described in Subsection 3.2. Finally, the k most reliable NER spans are retained and used in the following.

4 Entity Linking System

Given the pre-requisites from the neighborhood expansion, we are in the position to retrieve candidate entities for the TAC query using a Galago index of Wikipedia, and apply further re-ranking and NIL handling.

4.1 Candidate Entity Retrieval

We issue the candidate generation query that includes sequential dependence sub-models for the query string t, name variants $\vec{v}$, contextual sentences $\vec{s}$, as well as k most reliable NER spans $\vec{e}$ with its reliability probabilities $\vec{\phi}^E$. Further, different query concept types are weighted by settings of λ.

The resulting retrieval query is given in Galago query syntax in Figure 1. The query is scored against each entity's article full text with title, Freebase names, redirects, as well as anchor text from within Wikipedia and from the web.

To prioritize name matches over contextual information we set $\lambda_T + \lambda_V > \lambda_S + \lambda_E$. Since the weighting cannot guarantee that only articles with matching name variants are returned, we alternatively explore a two-pass alternative where first candidate entities are retrieved with the name variants model, which then are re-ranked with the full query.

4.2 Supervised Re-ranking

The candidate entity set is re-ranked with the supervised learning to rank framework, RankLib.[2] Features represent the similarity between a TAC query and a candidate entity based on string similarity of names, similarity of term vectors, name confidence based on ambiguity of anchor texts. For a full list of features, see Tables 3 and 4.

The ranker is trained in a supervised manner on TAC data from 2010. We omitted data from 2009 as it demonstrated a negative effect on the ranking performance as tested on 2011 queries.

In a preliminary study we evaluated various learning to rank rank models including LambdaRank. Our final model is based on generalized linear models, optimized with coordinate ascent and random re-starts.

4.3 NIL Classification and Clustering

As it is not the main focus of our work, we use simple heuristics for handling query mentions that are not included in the TAC knowledge base.

We allow the candidate entity set to contain any Wikipedia entity including many recent entities as well as U.S. states that are not contained in the TAC knowledge base. We link a query mention to NIL, if one of the following conditions hold. a) An empty candidate set is retrieved. b) The ranking score of the top ranked entity is below a threshold. c) The top ranked entity is not contained in the TAC knowledge base.

The NIL-threshold of the ranking score is trained on TAC data from 2011.

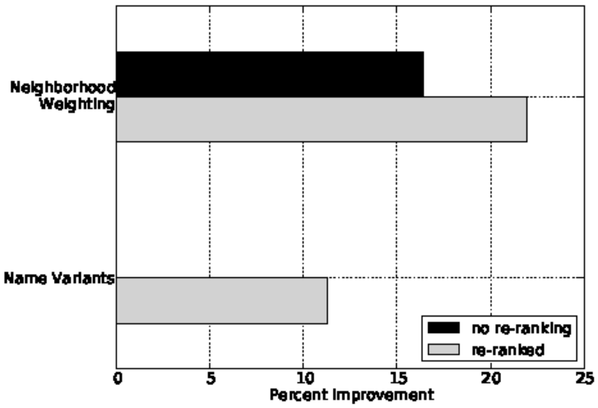
All query mentions that are predicted as NIL are clustered, either by the Wikipedia entity (in the case of c) or by identical surface forms.

5 Experimental Results

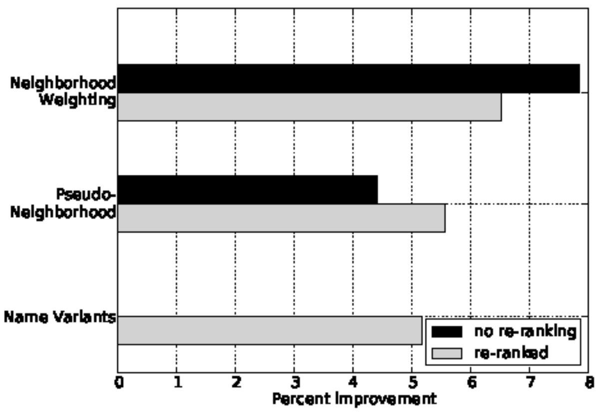
5.1 Corpus Preprocessing

In order to efficiently support the queries above, we index a Wikipedia dump with Galago. Basis is a Freebase Wikipedia Extraction (WEX) dump of English Wikipedia from January 2012 which provides the Wikipedia page in machine-readable XML format and relational data in tabular format. The Freebase dump contains 5,841,791 entries. We filter out non-article entries, such as category

[2] `http://www.cs.umass.edu/~vdang/ranklib.html`



(a) 2011 evaluation data.



(b) 2012 evaluation data.

Figure 2: Performance improvement in B^3-F1 over name-variants baseline (without re-ranking).

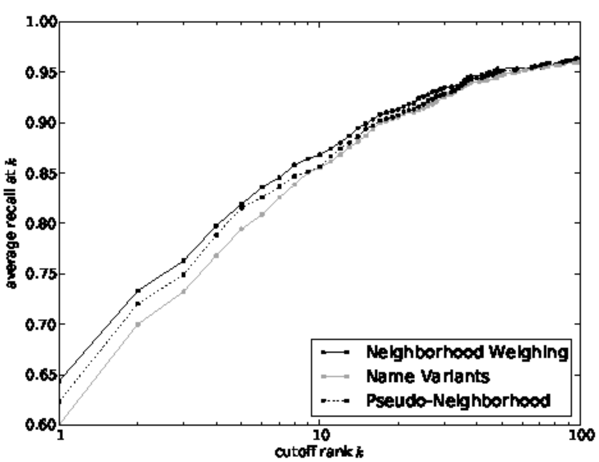


Figure 3: Candidate retrieval performance measured in in average recall at different cutoff ranks on data from 2012. The results are presented on a lin-log scale.

pages. The resulting index contains 3,811,076 articles and over 60 billion words.

The goal is to create an index with fields for anchor text (within Wikipedia as well as from the web), Wikipedia categories, Freebase names, Freebase types, redirects, article titles, and full-text for each article. Most of this information is contained in the WEX dump. We also incorporate external web anchor text to Wikipedia entries using the the Google Cross-Wiki dictionary, which contains 3 billion links and 297 million associations from 175 million unique anchor text strings.

The anchor extraction from the WEX dump is performed using the SPARK parallel processing framework,[3] which allows fast in-memory computation over large scale data in a cluster. The final merge of full-text and WEX meta-data with Google Cross-Wiki dictionary is performed using Hadoop MapReduce using the PIG parallel processing language.

In order to support the pseudo-relevance retrieval, the TAC source corpus is lightly preprocessed and indexed with Galago.

5.2 Parameters

For Neighborhood Expansion, we use 5 contextual sentences (limited to 200 characters), and up to $k = 10$ contextual NER spans. 30 pseudo-relevant documents are retrieved using a weighing parameters $\lambda_T = 0.4$, $\lambda_V = 0.4$, $\lambda_S = 0.2$, $\lambda_E = 0$.

For the candidate retrieval model, up to 100 candidate entities are retrieved using $\lambda_T = 0.35$, $\lambda_V = 0.35$, $\lambda_S = 0.1$, $\lambda_E = 0.2$.

For all sequential dependence models, we use weights 0.29 for unigrams, 0.21 for ordered window, and 0.5 for unordered window. For the Wikipedia index we use Dirichlet smoothing value of 96400, for TAC Source index a smoothing value of 2000.

5.3 Submitted Runs

We submitted six runs to the TAC KBP English monolingual Entity Linking Task testing the two neighborhood expansion techniques. The run for Neighborhood Weighting use the two-pass variant, ensuring that the entities include name variants, where the Pseudo-Neighborhood approach includes the experimental version that does not filter the candidates by name variants.

The results are evaluated in comparison to a baseline that retrieves candidates based on name variants only, i.e. using the query from Figure 1 without $\vec{s}$ and $\vec{e}$.

For each of the three approaches we submit one run that uses the full process including the supervised reranker and another run that uses the top 1 of the candidate retrieval directly. NIL classification and clustering is used in all cases.

[3] http://www.spark-project.org/

5.4 Entity Linking Performance

Performance on the official evaluation metric B^3+F1 is given in Table 1 as well as in improvement over the name-variants baseline in Figure 2. As our research did not focus on NIL clustering we also evaluate in terms of micro-average precision with similar findings.

The performance of the neighborhood weighting approach is consistently better than the baseline, achieving 16% improvement over the baseline in terms of B^3+F1 on 2011 data. In last year's competition this would have placed UMass on rank 6 (the baseline would have been beyond rank 27). The supervised re-ranker improved the results by another 5% (a total 22% over the baseline), placing UMass on rank 4 with an B^3+F1 score of 0.789.

Unfortunately on 2012, the supervised re-ranker (which was trained on 2010 data) did not yield consistent improvements, actually decreasing performance by up to 2%. Ignoring the re-ranked runs, Neighborhood Weighting gave 8% improvement over the name-variants baseline; the Pseudo-Neighborhood approach yields 4%.

As the supervised re-ranking score hurt the prediction performance by 2%, we suspect that it also affected the NIL classification in a negative way. We evaluate the performance we could have obtained if we would have submitted runs for other NIL thresholds as well in Table 1. In comparison to the trained NIL threshold, the ideal NIL threshold would have improved the Neighborhood Weighting approach by 7% and the Pseudo-Neighborhood approach by 4%.

Expansion with Neighborhood Weighting (without re-ranking) is our strongest run, which yields 5% improvement over the median among contestants in terms of B^3+F1; Expansion with Pseudo-Neighborhood still improves 3.7% over the median. With an ideal NIL threshold, the Neighborhood Weighting approach yields a B^3+F1 of 0.603 represents an 13% improvement over the median contestant.

5.5 Candidate Generation Performance

Our contribution on Neighborhood Expansion is aimed at improving the set of candidate entities, which are used as input to a refinement process. The goal is to maximize the number of true entities at high ranks. Our declared goal is to achieve 95% recall.

We evaluate the retrieval performance of the candidate ranking in terms of mean reciprocal rank (MRR) and recall of true entities in the candidate set for different cutoff ranks averaged over all "in KB" queries on 2012 data.

Results are presented in Figure 3 and Table 2. Across all cut-off ranks k and also in terms of mean reciprocal rank, Neighborhood Weighting is consistently the best method, achieving a recall of 80% at rank 5, 90% at rank 16, and 95% at rank 45.

The recall at cutoff rank 1 is equivalent to the micro-average precision metric on the focused "in KB" query set. Out of the set of 1177 queries, the difference in successfully identified entities is +51 for Neighborhood Weighting, and +27 for Pseudo-Neighborhood over the name-variants baseline.

6 Conclusion

All the different evaluations paint the same picture: The Neighborhood Weighing, which uses across-document information to identify the reliability of NER context, is the preferred method. The candidate ranking achieved competitive results even without further supervised re-ranking. The Pseudo-Neighborhood approach, which also introduces NER spans not included in the query document, still yields consistent improvement over the name variants baseline. We suspect that noise in the pseudo-relevant document set promoted spurious NER spans, letting the performance drop below Neighborhood Weighting. Future work will be about balancing the promotion of NER spans from other documents with spans found in the query document.

We envision the retrieved candidates to be further refined with elaborate inference methods, for instance, joint entity linking methods of a set of NER spans in a model similar to (Ratinov et al., 2011). As such inference methods are also time consuming, the ability to generate a small candidate set while guaranteeing high recall gives rise to elaborate inference methods. Our Neighborhood Weighting approach achieves 90% recall with candidates sets of size 16; 95% recall with size 45.

As a by-product, the Neighborhood Weighting identifies spurious and misleading NER spans. Omitting those from joint entity linking models, e.g. (Ratinov et al., 2011; Cucerzan, 2011) has the potential to further improve the overall results.

Acknowledgements

This work was supported in part by the Center for Intelligent Information Retrieval, in part by NSF grant #IIS-0910884, in part by UPenn NSF medium IIS-0803847, in part under subcontract #19-000208 from SRI International, prime contractor to DARPA contract #HR0011-12-C-0016. The University of Massachusetts also gratefully acknowledges the support of Defense Advanced Research Projects Agency (DARPA) Machine Reading Program under Air Force Research Laboratory (AFRL) prime contract no. FA8750-09-C-0181. Any opinions, findings, and conclusion or recommendations expressed in this material are the authors' and do not necessarily reflect the view of DARPA, AFRL, or the US government.

		2012, trained NIL threshold		2012, ideal NIL threshold		2011	
Approach	Run	B^3+F1	micro-avg Precision	B^3+F1	micro-avg Precision	B^3+F1	micro-avg Precision
Neighborhood Weighting	2	0.563	0.626	0.603	0.677	0.753	0.792
Re-ranked Neighborhood Weighting	1	0.556	0.615	0.577	0.648	0.789	0.823
Pseudo-Neighborhood	6	0.545	0.612	0.588	0.67		
Re-ranked Pseudo-Neighborhood	4	0.551	0.611	0.573	0.646		
Name variants	5	0.522	0.591	0.566	0.652	0.647	0.765
Re-ranked Name Variants	3	0.549	0.611	0.568	0.646	0.72	0.82
Median Performance		0.536	0.601				
Top Performance		0.73	0.766				

Table 1: Performance on the Entity Linking task.

	MRR	Avg Recall@1	Avg Recall@5	Avg Recall@20	Avg Recall@100
Neighborhood Weighting	0.752	0.644	0.819	0.913	0.962
Pseudo-Neighborhood	0.734	0.624	0.816	0.907	0.964
Name variants	0.716	0.601	0.794	0.906	0.962

Table 2: Candidate Retrieval Performance on 2012 data.

References

S. Cucerzan. 2011. Tac entity linking by performing full-document entity extraction and disambiguation. *Proceedings of the Text Analysis Conference*.

J.R. Finkel, T. Grenager, and C. Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics.

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

Andrew McCallum, Karl Schultz, and Sameer Singh. 2009. Factorie: Probabilistic programming via imperatively defined factor graphs. In *In Advances in Neural Information Processing Systems 22*, pages 1249–1257.

Donald Metzler and W. Bruce Croft. 2005. 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.

D. Metzler and W.B. Croft. 2007. Latent concept expansion using markov random fields. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*.

S. Monahan, J. Lehmann, T. Nyberg, J. Plymale, and A. Jung. 2011. Cross-lingual cross-document coreference with entity linking. *Proceedings of the Text Analysis Conference*.

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

T. Strohman. 2007. *Efficient processing of complex features for information retrieval*. Ph.D. thesis, University of Massachusetts Amherst.

J. Xu and W.B. Croft. 1996. Query expansion using local and global document analysis. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 4–11. ACM.

Feature Name	Type	Description
wordMatch	name variants	Number of words occuring in both names
wordMiss	name variants	Number of words missed in the query string
substringTest	name variants	1.0 if one name is substring of the other (ignoring casing); otherwise 0.0
editDistance	name variants	Levenshtein String edit distance between query mention and Wikipedia title
tokenDice	name variants	Dice coefficient on name token sets
tokenJaccard	name variants	Jaccard index on name token sets
totalSourcesMatching	name variants	Counts matching in multiple sources, e.g. anchor text, title, freebase name, and redirect
exactMatchCount_anchor-exact	name variants	Number of Wikipedia anchor texts that matches the query string (ignoring casing and punctuation)
exactMatchBool_anchor-exact	name variants	1.0 if above score non-zero; otherwise 0.0
exactMatchCount_web_anchor-exact	name variants	Number of web anchor texts that matches the query string (ignoring casing and punctuation) according to the Google Cross-Wiki dictionary
exactMatchBool_web_anchor-exact	name variants	1.0 if above score non-zero; otherwise 0.0
exactMatchCount_fbname-exact	name variants	Number of freebase names that matches the query string (ignoring casing and punctuation)
exactMatchBool_fbname-exact	name variants	1.0 if above score non-zero; otherwise 0.0
exactMatchCount_redirect-exact	name variants	Number of redirect page titles that matches the query string (ignoring casing and punctuation)
exactMatchBool_redirect-exact	name variants	1.0 if above score non-zero; otherwise 0.0
exactMatchCount_title-exact	name variants	Number of page titles that match the the query string (ignoring casing and punctuation)
exactMatchBool_title-exact	name variants	1.0 if above score non-zero; otherwise 0.0
weakAlias	name variants	1.0 if names match according to dice, acronym, or substring test; otherwise 0.0
fieldLikelihood_anchor	name variants	Unigram Query likelihood (as unnormalized log-prob) of the query mention under the Wikipedia anchor text's language model
fieldProbability_anchor	name variants	N-gram probability of the query mention under the Wikipedia anchor text's language model
fieldLikelihood_fbname	name variants	Unigram Query likelihood (as unnormalized log-prob) of the query mention under the Freebase name dictionary's language model
fieldProbability_fbname	name variants	N-gram probability of the query mention under the Freebase name dictionary's language model
fieldLikelihood_redirect	name variants	Unigram Query likelihood (as unnormalized log-prob) of the query mention under the redirect pages' language model
fieldProbability_redirect	name variants	N-gram probability of the query mention under the redirect pages' language model
fieldLikelihood_web_anchor	name variants	Unigram Query likelihood (as unnormalized log-prob) of the query mention under the web anchor text's language model
fieldProbability_web_anchor	name variants	N-gram probability of the query mention under the web anchor text's language model
fieldLikelihood_title	name variants	Unigram Query likelihood (as unnormalized log-prob) of the query mention under the title's language model
fieldProbability_title	name variants	N-gram probability of the query mention under the title's language model
diceTestFullCharacterScore	name variants	Dice coefficient of character sets.
diceTestFullCharacter	name variants	1.0 if above score > 0.9; otherwise 0.0
diceTestAlignedCharacterScore	name variants	Maximum character dice score of left- and right aligned character sets.
diceTestAlignedCharacter	name variants	1.0 if above score > 0.9; otherwise 0.0
diceTestFullWordScore	name variants	Dice coefficient words sets; lower cased and tokenized on white space and punctuation.
diceTestFullWord	name variants	1.0 if above score > 0.9; otherwise 0.0
diceTestAlignedWordScore	name variants	Maximum character dice score of left- and right word sets; lower cased and tokenized on white space and punctuation.
diceTestAlignedWord	name variants	1.0 if above score > 0.9; otherwise 0.0

Table 3: Features of the query mention and candidate Wikipedia entity.

Feature Name	Type	Description
galagoscore	name, context words, ner	Retrieval score of this candidate, taken from the Galago candidate retrieval model.
galagoscoreNorm	name, context words, ner	Retrieval score of this candidate, normalized over all candidates in the retrieved set.
inlinks	entity	Log number of Wikipedia inlinks - a measure of popularity
stanfExternalinlinks	entity	Log number of web inlinks - a measure of popularity
linkProb	entity	If a name matches the Wikipedia anchor text, probability that the matching anchor text refers to only this entity (versus other entities)
externalLinkProb	entity	If a name matches the web anchor text, probability that the matching anchor text refers to only this entity (versus other entities)
cosineFeature-doc	document	TF-IDF weighted cosine similarity of terms between the query document and Wikipedia article.
jaccardFeature-doc	document	Jaccard coefficient of document term vectors (of query document and article)
jsdivergenceFeature-doc	document	Jensen-Shannon divergence between Dirichlet smoothed document language models (of query document and article)
klfFeature-doc	document	KL divergence of the query document's Dirichlet smoothed language model and the article's language model.

Table 4: Features of the query mention and candidate Wikipedia entity (Continued).