Graph-based Named Entity Linking with Wikipedia

Ben Hachey[1,2], Will Radford[2,3], and James R. Curran[2,3]

[1] Department of Computing
Macquarie University
NSW 2109, Australia
ben.hachey@mq.edu.au
[2] Capital Markets CRC
55 Harrington Street
NSW 2000, Australia
[3] School of Information Technologies
University of Sydney
NSW 2006, Australia
{wradford,james}@it.usyd.edu.au

Abstract. Named entity linking (NEL) grounds entity mentions to their corresponding Wikipedia article. State-of-the-art supervised NEL systems use features over the rich Wikipedia document and link-graph structure.

Graph-based measures have been effective over WordNet for word sense disambiguation (WSD). We draw parallels between NEL and WSD, motivating our unsupervised NEL approach that exploits the Wikipedia article and category link graphs. Our system achieves 85.5% accuracy on the TAC 2010 shared task — competitive with the best supervised and unsupervised systems.

Keywords: web intelligence, text mining, integration, entity resolution, Wikipedia

1 Introduction

Named entity linking (NEL) grounds mentions of entities in text to a central knowledge base (KB). Before Wikipedia, open-domain NEL lacked the requisite wide-coverage KB [26]. For example, we can ground: *David Murray recruited Amidu Berry from Positive Black Soul* to Wikipedia articles for David Murray (saxophonist) and Positive Black Soul and mark that *Amidu Berry* does not appear in Wikipedia. NEL systems have begun to exploit Wikipedia's rich structure: category data [5, 6] and infobox data [30, 19] have complemented traditional string matching and context similarity approaches; link probabilities conditioned on mentions have been calculated across Wikipedia [23]. State-of-the-art supervised approaches [31, 9, 18] learn to rank entity candidates.

NEL is very similar to word sense disambiguation (WSD), with Wikipedia articles playing the role of WordNet [11] synsets. Yet the connection with WSD has not been exploited by existing approaches to NEL. Unlike WSD, NEL does not assume the KB is complete, requiring entity mentions without KB entries to be marked as NIL.

Both Wikipedia and WordNet provide links between KB entries: as hyperlinks to other articles and categories in Wikipedia; and as semantic relations, e.g. hypernym and meronym, in WordNet. [24, 25] exploited the graph structure of WordNet for unsupervised WSD rivalling supervised approaches.

In this paper, we adapt these successful approaches for WSD to the NEL task by exploiting Wikipedia's link graph structure, where articles and categories are nodes. The nature and distribution of Wikipedia and WordNet links are very different, and so there is no guarantee WSD approaches will work for NEL.

We use local subgraphs containing the candidate Wikipedia articles for all entity mentions in the document. We include nodes for any intervening article and/or category on short paths connecting entity mention nodes. We explore subgraph selection parameters including inter-article versus category links, maximum path length, and link directionality. We compare two graph measures: Degree centrality [24] and PageRank [4] for ranking candidates. The graph scores are combined with cosine similarity calculated from the paragraph around the entity mention and the candidate Wikipedia article.

We follow the standard Text Analysis Conference (TAC) NEL evaluation methodology. Our best graph-based model achieves 85.5% accuracy on the TAC 2010 shared task. This is competitive to the top-reported TAC 2010 supervised (86.8%) and unsupervised (85.8%) systems. This demonstrates that NEL is indeed similar to WSD and can be performed using graph-based approaches.

2 Background

NEL is similar to the widely-studied problem of word sense disambiguation (WSD), where the goal is to identify the sense of a word given the context. For example, the word *bank* in the sentence *I put money in the bank* is more likely to refer to a financial institution than a river side. WSD is often performed with respect to WordNet [11], a lexical database that maps words to synonym sets (synsets).

In NEL, the same entity can similarly be referred to using different mentions strings; and a single mention string can ambiguously refer to multiple entities. But, due to the lack of a comparably comprehensive *sense* inventory for entities, most previous work focuses on identifying and characterising entity mentions [14, 29], or on clustering mentions within documents [15, 28] and across documents [1, 8]

Recently, Wikipedia has emerged as a wide-coverage KB of collective knowledge about notable entities, leading to exploration of NEL over arbitrary named entity types [5, 6]. Unfortunately, previous evaluations have focused on final accuracy only [20]. Here, we break NEL into three components for systematic comparison: *extraction* of entities and their contexts; *search* generates candidate entity nodes; and *disambiguation* selects the best candidate or NIL. We explore search and disambiguation independently, holding the other components constant to isolate their effect.

Search for WSD assumes that WordNet is a complete lexical resource and consists of a lexical lookup to find the possible synsets for a given word. The same approach is taken in a number of Wikipedia linking approaches [22, 23, 17, 12]. However, this does not provide a mechanism for dealing with entities that are not present in the database. Search for NEL requires a noisier candidate generation process, often using fuzzy matching to improve recall [30, 18]. Finally, NEL does not assume the KB is complete, requiring entity mentions without KB entries to be marked as NIL [20].

Table 1. Comparison of TAC data sets.

	TAC 2009 test		TAC 2010 train		TAC 2010 test	
N	3,904		1,500		2,250	
KB	1,675	(43%)	1,074	(72%)	1,020	(45%)
NIL	2,229	(57%)	426	(28%)	1,230	(55%)
PER	627	(16%)	500	(33%)	751	(33%)
ORG	2710	(69%)	500	(33%)	750	(33%)
GPE	567	(15%)	500	(33%)	749	(33%)
News	3904	(100%)	783	(52%)	1500	(67%)
Web	0	(0%)	717	(48%)	750	(33%)

Simple disambiguators based on cosine similarity between mention contexts and article text were highly successful in NEL evaluations [30]. However, other successful systems have exploited Wikipedia's rich structure: Bunescu and Paşca and Cucerzan use page categories as features [5, 6]; Dredze et al. use key-value information from infoboxes [9]; and Milne and Witten calculate probabilities of pages given aliases across Wikipedia [23]. Simple features based on common links between articles have also been used recently [17, 10, 18, 27]. However, the graph-based techniques we explore here have not been used for NEL disambiguation.

Further parallels between NEL and WSD can be drawn in terms of their graph structure. WordNet includes links between synsets that represent semantic relations (e.g., hyponymy, meronymy, antonymy). This graph has been used successfully to incorporate global information into unsupervised WSD rivalling supervised approaches (Section 6). While WordNet and Wikipedia have been found to have similar small-world graph structures [13], the inter-article links in Wikipedia are unconstrained in terms of the relation types they represent.

Based on this comparison, we explore whether successful graph-based approaches from the WSD literature [24, 25] are appropriate for NEL with respect to Wikipedia. We also report results for implementations of seminal approaches from the literature [5, 6], providing a comparison to more recent approaches for the first time.

3 Evaluation Data and Methodology

The Text Analysis Conference Knowledge Base Population (TAC-KBP) shared tasks have established common datasets that emphasise ambiguous queries, and formalise NIL linking for queries not referring to a KB node [20]. TAC queries consist of an entity mention string (e.g., *Mr. Murray*) and a source document containing it. The gold standard is a reference to a TAC KB node or NIL if there is no corresponding node in the KB. The data sets are summarised in Table 1. There are several notable differences. First, TAC 2010 training data is highly skewed towards non-NIL queries at 72%. Second, the TAC 2009 data is highly skewed towards ORG entities (69%). Third, the TAC 2010 data sets include web documents as well as newswire. We use the TAC 2010 training data as our training set, the TAC 2009 data as our development set (Section 4 and 5), and the TAC 2010 test data as our final test set (Section 7).

Table 2. Notation for searcher analysis measures.

N	Number of queries in data set
$\mathcal{G}$	Gold standard annotations for data set ($\lvert\mathcal{G}\rvert = N$)
$\mathcal{G}_i$	Gold standard for query i (KB ID or NIL)
$\mathcal{C}$	Candidate sets from system output ($\lvert\mathcal{C}\rvert = N$)
$\mathcal{C}_i$	Candidate set for query i
$\mathcal{C}_{i,j}$	Candidate at rank j for query i (where $\mathcal{C}_i \neq \emptyset$)

The TAC KB used for all experiments is derived from articles in the October 2008 Wikipedia dump that have infoboxes. It includes approximately 200,000 PER nodes, 200,000 GPE nodes, 60,000 ORG nodes and more than 300,000 miscellaneous/non-entity nodes. We also use a more recent Wikipedia dump (29 July 2010) for extracting aliases (e.g., titles of redirect pages, link anchor text) and building graphs. This contains 3,398,404 articles. After disambiguation, any articles that can not be mapped to the subset of articles in the TAC KB are marked as NIL.

We use the following evaluation measures, defined using the notation in Table 2. The first is the official TAC measure for evaluation of end-to-end systems. TAC also reports KB accuracy ($A_{\mathcal{C}}$) and NIL accuracy ($A_{\emptyset}$), which are actually recall scores for non-NIL and NIL respectively. The remaining measures are used here to analyse performance of the different search strategies for generating candidate entities.

Accuracy: percentage of correctly linked queries (i.e., queries where the top-ranked candidate/NIL is the same as the gold standard annotation).

$$A = \frac{|\{\mathcal{C}_{i,0}|\mathcal{C}_{i,0} = \mathcal{G}_i\}|}{N} \tag{1}$$

Candidate Count: mean cardinality of the candidate sets. Fewer candidates mean easier disambiguation.

$$\langle C \rangle = \frac{\sum_i |\mathcal{C}_i|}{N} \tag{2}$$

Candidate Precision: mean percentage of non-empty candidate sets containing the correct entity.

$$P_{\mathcal{C}} = \frac{|\{\mathcal{C}_i|\mathcal{C}_i \neq \emptyset \wedge \mathcal{G}_i \in \mathcal{C}_i\}|}{|\{\mathcal{C}_i|\mathcal{C}_i \neq \emptyset\}|} \tag{3}$$

Candidate Recall: mean percentage of KB queries where the candidate set includes the correct candidate.

$$R_{\mathcal{C}} = \frac{|\{\mathcal{C}_i|\mathcal{G}_i \neq \text{NIL} \wedge \mathcal{G}_i \in \mathcal{C}_i\}|}{|\{\mathcal{G}_i|\mathcal{G}_i \neq \text{NIL}\}|} \tag{4}$$

Nil Precision: mean percentage of NIL queries with *correctly empty* candidate sets.

$$P_{\emptyset} = \frac{|\{\mathcal{C}_i|\mathcal{C}_i = \emptyset \wedge \mathcal{G}_i = \text{NIL}\}|}{|\{\mathcal{C}_i|\mathcal{C}_i = \emptyset\}|} \tag{5}$$

Nil Recall: mean percentage of NIL queries for which the candidate set is empty. A high NilRecall rate is valuable because it is difficult for disambiguators to determine whether queries are NIL-linked when candidates are returned.

$$R_{\emptyset} = \frac{|\{\mathcal{C}_i|\mathcal{G}_i = \text{NIL} \wedge \mathcal{C}_i = \emptyset\}|}{|\{\mathcal{G}_i|\mathcal{G}_i = \text{NIL}\}|} \tag{6}$$

Table 3. High-precision search (TAC 2009 data).

Alias Source	$\langle C \rangle$	P_C^∞	R_C^∞	$P_\emptyset$	$R_\emptyset$
Title+Redirect	30.0	83.8	58.6	76.6	93.9
− Hatnote	18.9	**89.0**	39.3	68.7	97.5

Extraction The extraction component for preprocessing query source documents is held constant for the experiments here. We first extract article text, discarding markup and non-visible content if they are formatted using a markup language. The C&C tagger [7] is used to extract named entity mentions from the text. Finally, coreference chains are formed using an implementation of a high-precision algorithm from Cucerzan [6]. This assumes that longer mentions (e.g., *David Murray*) are generally more specific than shorter mentions (e.g., *Murray*) and will thus be easier to disambiguate. A match is accepted when the short mention is the beginning or end of the long mention and the long mention has no more than three additional tokens. Mentions entirely in uppercase may be acronyms (e.g., *DM*). In this case, a match is accepted when the acronym letters correspond to the token-initial characters of a longer mention.

4 Search Experiments

Table 3 contains analysis of a search strategy that is tuned for the highest possible precision at the expense of recall. The first row corresponds to exact matching against Wikipedia article titles and redirect page titles. The second row uses the same search but removes any results that have hatnote templates — i.e., a Wikipedia link to another article or disambiguation page describing entities with the same name. Removing these pages results in a 5 point increase in candidate precision at the cost of 19 points candidate recall. The imbalanced effect is due to the fact that pages with hatnote templates are more popular than their non-hatnoted counterparts.

This strategy is designed for backoff approaches, where preference is given to high-precision matches before resorting to noisier approaches tuned for recall. It is also useful for identifying minimally ambiguous entities in a document, which we use as reliable context articles in building local networks for graph-based disambiguation. All graph results in the following sections use the hatnote-filtered search for this purpose. This is comparable to the approach in [18], which only uses context entity mentions that are close to the query, have candidate page probability in Wikipedia of ≥ 0.01 given the mention string, and frequency in Wikipedia ≥ 4.

Analysis of Implemented Searchers Table 4 contains search results for our implementations of prominent approaches from the literature on the TAC 2009 data. The first row corresponds to our Bunescu and Paşca [5] searcher, which uses exact matching over Wikipedia page titles, titles of redirect pages and titles of disambiguation pages. The second row corresponds to our Cucerzan [6] searcher, which forms coreference chains (Section 3) creates a query containing all mentions in the chain. The query is searched using an exact-match lookup against article titles, redirect page titles, disambiguation page titles and bold terms from article first paragraphs. The third row corresponds to

Table 4. Performance of searchers from the literature.

Searcher	$\langle C \rangle$	P_C^∞	R_C^∞	$P_\emptyset$	$R_\emptyset$
Bunescu and Paşca	3.60	56.3	77.0	86.6	62.7
Cucerzan	2.42	59.7	79.8	88.4	66.0
Varma et al.	2.99	59.8	81.2	90.9	66.4
Tuned backoff	3.84	58.2	**87.3**	92.9	57.9

Table 5. Effect of searchers on graph-based reranking.

Searcher	A_C	$A_\emptyset$	A
Cucerzan	72.7	83.4	**78.8**
Backoff	74.4	80.5	77.9

our Varma et al. [30] searcher, which replaces acronyms with full-forms where possible and employs a backoff search strategy that favours high-precision matching against page titles that map to the KB over alias search. Search includes exact match against article, redirect and disambiguation titles as well as disambiguation hatnotes and bold terms in the first paragraph of an article. Finally, the last row corresponds to a high recall searcher, which is discussed in more detail below.

The implemented Cucerzan and Varma et al. searchers are clearly the winners here. They both achieve candidate precision of approximately 60% at candidate recall of 80%. This suggests that coreference and acronym handling are important. In terms of candidate count, the Cucerzan searcher performs slightly better. It returns a candidate set size that, on average, contains 0.57 fewer items. This corresponds to a reduction in ambiguity of 19% with respect to the Varma et al. searcher.

The last row of Table 4 corresponds to a backoff approach that aims to further increase candidate recall without sacrificing too much in terms of ambiguity. This first performs a hatnote-filtered high-precision match (Table 3). If this produces no results, it resorts to exact match over all Wikipedia alias sources (i.e., article titles, redirect titles, link anchor text, apposition-stripped titles, bold words in article first paragraphs, titles of pages with hatnotes, disambiguation titles, bold disambiguation terms, and titles of redirects pointing to disambiguation pages). Finally, it resorts to character n-gram match. Based on development experiments (TAC 2009 data), n-gram matches are only accepted if they have a Solr score $\geq$ three and an edit distance ≥ 0.7 times the number of characters in the term. The last row in Table 4 shows that this increases candidate recall by 6 but also results in the highest ambiguity with a candidate count of 3.84.

Effect of Searchers on Disambiguation Table 5 contains a comparison of high recall (Backoff) and high precision (Cucerzan) searchers for graph-based reranking of cosine scores on the TAC 2009 data. The high-precision searcher is 1.1 points better in overall accuracy. In the rest of the paper, we use this searcher to generate candidates for the Cosine, Milne and Witten, Cucerzan and graph-based disambiguation approaches. This serves to focus the remaining experiments on the relative effect of these disambiguators given the same search candidates as input.

5 Existing Approaches to Disambiguation

A number of disambiguators from the literature are used to benchmark our graph-based approaches. Bunescu and Paşca [5] and Cucerzan [6] are the first reported NEL approaches. Milne and Witten [23] is the seminal system from a closely task that treats Wikipedia as a complete lexical resource and formalises linking as a WSD-style task. Since this system does not have a search phase as defined here, we use the Cucerzan searcher for comparability to that system and our graph-based approaches. Finally, Varma et al. [30] and Lehmann et al. [18] are included since they were the top systems at TAC 2009 and TAC 2010 respectively.

The Bunescu and Paşca disambiguator uses a Support Vector Machine (SVM) ranking model, using the SVM$^{\text{light}}$ toolkit.[4] Two types of features are used. The first is the real-valued cosine similarity between the query context and the text of the candidate entity page. The second is generated by creating a 2-tuple for each combination of candidate categories and context words. Based on development experiments (TAC 2009 data), we use great and great-great grandparent categories. Following Bunescu and Paşca, categories have to occur 200 times or more and context words are those that occur within a 55-token context window of the entity mention.

Cucerzan disambiguates the query mention with respect to document-level vectors derived by summing over candidate entity vectors. Candidate vectors include the article's categories (e.g., American jazz composers)[5] and its contexts (anchors of links in the first paragraph and of reciprocal links anywhere in the article). Candidate vectors are scored by taking the scalar product with the document-level vector, minus a penalty to subtract the candidate's contribution to the document-level vector. The resulting calculation is equivalent to an un-normalised cosine, which has the effect of giving more weight to candidates with more categories and/or contexts.

The Varma et al. approach ranks candidates based on the textual similarity between the query context and the text of the candidate page, using the cosine measure. Here, the query context is the full paragraph surrounding the query mention, where paragraphs are easily identified by double-newline delimiters in the TAC source documents.

The Milne and Witten disambiguator uses a C4.5 decision tree to combine three sources of information: commonness (probability of an article given an alias), relatedness (set overlap of in-links for the query candidate and the unambiguous articles discovered from the local context of the query document) and context quality (sum of the weights that were previously assigned to each unambiguous article from the local context). For the comparison here, we adapted the Wikipedia Miner version 1.1[6] disambiguator to take a list of mentions and their candidate sets. The data and models are based on a different version of Wikipedia, but we ensure that the disambiguation considers only candidates that we have passed in to avoid returning a candidate that does not exist in our version of Wikipedia.

[4] `http://www.cs.cornell.edu/People/tj/svm_light/svm_rank.html`

[5] Following Cucerzan, we exclude categories that we do not deem to be useful semantic types for NEL. We filter categories whose name contains a key word or its plural (`article, page, date, year, birth, death, living, century, acronym, stub`) a four-digit number (i.e., a year), or was `Exclude in print`.

[6] `http://wikipedia-miner.sourceforge.net/` (code: 04/2009, data: 03/2009)

The Lehmann et al. disambiguator uses features based on mention and candidate name similarity, as well as context similarity. As in Milne and Witten [23], unambiguous link anchors close to the mention are used as context. High-coverage entity type matching is also used to ensure semantic compatibility between mention and candidate entities. Features indicating which searcher source retrieved the candidate let the classifier learn *which* sources to trust. A heuristic over the features is used to rank the candidates. They propose two NIL classification systems: a supervised binary logistic classifier that classifies the top ranked few candidates and an unsupervised system based on a subset of features and a minimum confidence threshold for non-NIL.

6 Graph-based Approaches to Disambiguation

In this section, we propose an unsupervised approach that operates over a link graph of Wikipedia articles for document mentions. Wikipedia's size prohibits building the full graph, instead we construct a subgraph by creating vertices for each unambiguous mention. Following [23], these are mentions for which a searcher (see Section 4) returns exactly one candidate article. We also create one vertex for each candidate article returned for the query mention. The seed subgraph is expanded by tracing length-limited paths between each vertex via other articles and categories – extra vertices, adding them as required. Once built, a graph measure is applied, and the query candidates can then be ranked by their score.

Consider Figure 1, where *Positive Black Soul* has been unambiguously matched but we do not know whether the query 'David Murray' refers to the Iron Maiden guitarist or the jazz saxophonist. The sub-graph shows that David Murray (saxophonist) is connected to Positive Black Soul, since their pages are both linked to by the article Fo Deuk Revue, an album they collaborated on. The Iron Maiden guitarist, however, does not have any connections.

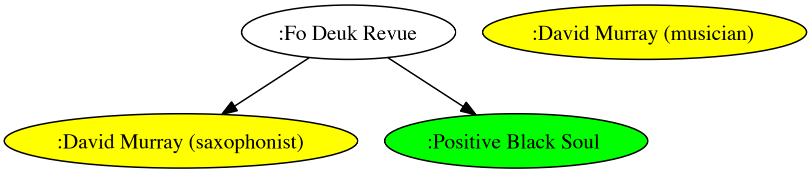

Fig. 1. Graph of 'David Murray' candidates with unambiguous match: 'Positive Black Soul'.

We are also motivated by the comparison to WSD (Section 2), which raises the question of whether graph-based approaches from the WSD field [2, 21] could lead to improvements in NEL. Navigli and Lapata [25] provide a recent comparative survey and demonstrate that unsupervised, graph-based approaches rival supervised WSD. They build graphs based on all candidate groundings for all nouns in a sentence and compare the performance of various graph measures, including degree centrality, PageRank [4], HITS [16], the key-player problem [3] and betweenness centrality. Results show that degree centrality leads to the best results. Degree centrality is also preferable in terms of runtime complexity $\mathrm{O}(n)$ compared to $\mathrm{O}(n^2)$ or worse for other algorithms.

Table 6. Characteristics of Wikipedia subgraphs.

Article	Category	$\lvert Q\rvert$	$\langle\lvert V\rvert\rangle$	$\langle\lvert E\rvert\rangle$
2	0	1,343	14,271	33,037
0	3	1,344	2,535	23,689

Based on these results, we explore the two graph measures: degree centrality and PageRank. Degree centrality is simply the number of edges terminating in a given vertex v normalised by the total number of vertices in the graph minus 1:

$$D(v) = \frac{|\{u|\,\langle u, v\rangle \in E\}|}{|V| - 1} \tag{7}$$

Degree treats all edges as being equal. PageRank, by contrast, encodes the notion that edges involving more highly linked nodes are more important. PageRank scores are calculated using a recursive Markov chain algorithm based on the following equation:

$$PR(v) = \frac{(1-\alpha)}{|V|} + \alpha \sum_{\langle u,v\rangle \in E} \frac{PR(u)}{|\{w|\,\langle u, w\rangle \in E\}|} \tag{8}$$

where α is a damping factor representing the probability at each page a random surfer will get bored and request another random page. It is set to the default of 0.85 here. The denominator in the sum is the number of links going out of vertex u.

The maximum path length for inter-article links is two, a reasonable trade-off between efficiency and detail. For category links, the maximum is three, which allows paths through up to two category pages (e.g., David Murray (saxophonist) ← American jazz composers ← American composers → Leon "Pee Wee" Whittaker).

We consider two kinds of extra vertices individually and combined: inter-article links in paragraphs and category links. For example, the David Murray (saxophonist) page has links from the pages describing albums to which he contributed, including Fo Deuk Revue. Categories collect articles on similar subjects (e.g., the David Murray page contains a American jazz composers category, which is also present on pages for other musicians including John Coltrane). Categories are not strictly hierarchical, since they can have loops. However, there is a notion of parent (e.g., American composers for American jazz composers).

We use the graph scores to reweight cosine-ranked candidates as follows, e.g.:

$$s_D(c) = cosine(c) * D(c) \tag{9}$$

$$s_{PR}(c) = cosine(c) * PR(c) \tag{10}$$

where c is a candidate, D is degree centrality and PR is PageRank. Unlike Navigli and Lapata [25], we use directed edges since Wikipedia does not have necessarily reciprocal links like WordNet. In development experiments (TAC 2009 data), this resulted in a negligible difference for Degree and an $Accuracy$ improvement of 0.5 for PageRank.

Table 6 contains some descriptive statistics for the extracted graphs on the development data (TAC 2009). The first and second columns contain the maximum path length for article and category links respectively. The third column ($|Q|$) contains the number

Table 7. Comparison of systems (TAC 2010 test data). Rows 15-17 are from the literature.

Row	System	$A_{\mathcal{C}}$	$A_{\emptyset}$	A
1	NIL Baseline	0.0	100.0	54.7
2	Title Baseline	37.8	98.4	70.9
3	+ Redirect Baseline	63.7	95.1	80.9
4	Cosine	72.5	88.9	81.5
5	Bunescu and Paşca	69.1	90.8	81.0
6	Milne and Witten	50.9	95.9	75.5
7	Varma et al.	70.4	91.1	81.7
8	Cucerzan	78.5	89.2	84.4
9	Article Graph, Degree	79.8	90.2	**85.5**
10	Category Graph, Degree	79.0	89.3	84.6
11	Combined, Degree	79.8	89.9	85.3
12	Article Graph, PageRank	78.0	90.0	84.6
13	Category Graph, PageRank	78.5	89.1	84.3
14	Combined, PageRank	78.4	90.0	84.8
15	TAC 2010 median	UNK	UNK	68.4
16	Lehmann unsupervised	79.2	91.2	85.8
17	TAC max (Lehmann)	80.6	92.0	86.8

of queries for which a graph was extracted (out of 3,904 total). The fourth column ($|V|$) contains the average number of vertices across these queries. And, the final column ($|E|$) contains the average number of edges. The first row corresponds to the article graph. And, the second row corresponds to the category graph, which has far fewer nodes than the article graph, but a substantially higher ratio of edges per node.

7 Linking Experiments

Table 7 contains evaluation results on the held-out TAC 2010 test set. The first three rows correspond to baseline systems that have no disambiguation component. The NIL baseline simple returns NIL for each query and achieves an overall accuracy of 54.7% due to the slightly higher proportion of NIL queries in the TAC 2010 test data. The Title baseline consists of exact match on article titles and the Title+Redirect baseline consists of exact match on both article and redirect page titles. Note that the Title+Redirect baseline is a much stronger benchmark than the median result (Row 15) reported at TAC 2010, achieving an overall accuracy 14 points higher. The fourth row corresponds to a system that uses our implementation of the Cucerzan searcher and a cosine disambiguator. This is used for search and cosine scoring in the graph-based approaches.

The fifth through eighth rows correspond to our implementation of systems from the literature. Among these, the Cucerzan [6] approach is a clear winner. At an overall accuracy of 84.4%, it is four points better than our implementations of Bunescu and Paşca [5] and Varma et al. [30] approaches. And, it is only two points shy of the top result from TAC 2010 (86.8%, row 17). The Milne and Witten approach [23] is not a fair comparison here since it was tuned for a precision-oriented linking task that is more like WSD. However, it does give an idea of what recall is sacrificed to achieve

Table 8. Overall accuracy by genre and entity type.

		News			Web	
System	ORG	GPE	PER	ORG	GPE	PER
NIL Baseline	72.6	21.0	91.0	33.2	56.6	33.1
Title Baseline	74.6	52.4	91.0	50.8	75.1	76.5
+ Redirect Baseline	76.8	66.8	**98.0**	81.2	**76.7**	86.9
Cosine	**81.6**	69.8	97.6	84.8	62.2	88.0
Bunescu and Paşca	77.2	64.4	97.2	88.8	72.7	**89.6**
Milne and Witten	78.0	56.8	97.6	71.6	67.9	74.9
Varma et al.	78.0	67.8	97.2	**90.8**	70.3	88.0
Cucerzan	79.6	80.8	97.0	82.8	73.1	88.4
Article Graph, Degree	78.2	**82.2**	97.2	84.0	73.9	88.4
Category Graph, Degree	80.0	80.8	97.4	87.6	75.9	**89.6**
Combined, Degree	80.0	81.4	97.4	86.4	75.5	88.4

high precision. With respect to our other implementations, the Bunescu and Paşca and Varma et al. approaches lead to conservative linkers (high NIL accuracy and much lower candidate accuracy) while the Cucerzan approach results in a liberal linker. This may be due in part to their respective search strategies, explored in Section 4 above. Bunescu and Paşca and Varma et al. have search strategies with relatively high ambiguity, making it more difficult for the disambiguator to identify the correct candidate. By contrast, Cucerzan has a search strategy that generates the fewest candidates on average, but achieves comparable candidate recall.

Rows 9–14 correspond to our graph-based reranking approaches, parametrised by the source of their link information — articles, categories, or combined — and by the graph connectivity measure used — Degree or PageRank. In terms of overall accuracy (A), graph-based reranking leads to substantial improvements over the cosine input, ranging from 2.8 to 4. Substituting or combining link sources has little effect, though the category graph scores are slightly lower. Comparing measures, Degree achieves higher overall accuracy than PageRank, consistent with results for WSD [25].

At 85.5%, our best graph-based approach represents an improvement of 1.1 in overall accuracy with respect to the previous state of the art in unsupervised linking (Row 8) and is comparable to the best unsupervised approach at TAC 2010 (Row 16). Our approach uses a simple solution that combines highly efficient cosine scoring with an elegant approach to global disambiguation based on the Wikipedia link graph. Furthermore, our graph-based result is not far off the maximum overall accuracy reported at TAC 2010 (86.8%), a result obtained by a supervised system that relies on annotated training data to incorporate various features (Row 17).

Results by Genre and Entity Type Table 8 contains overall accuracy broken down by genre (news or web) and entity type (ORG, GPE or PER) on the TAC 2010 test data. The best score in each column is in bold. The first thing to note is that no approach is consistently best across genres and entity types. This suggests two directions for future work: 1) system combination by voting and 2) entity-specific models and tailoring approaches to individual entity types and/or genres. Next, the percentage of NIL queries (as reflected in the NIL baseline scores) varies hugely across genre and entity types. In particular, the NIL percentage in web text is much lower than in news text for ORG and PER entities, but much higher for GPE entities.

In terms of our graph-based systems, it is interesting to note the improvement over cosine is due primarily to large gains for GPE entities, where scores are more than 12 points higher on both news and web text. It is also interesting to note that the article and combined graph approaches are consistently better than or equal to our implementations of previous unsupervised strategies from Varma et al. and Cucerzan. ORG in web text is the exception, where Varma et al.'s backoff search strategy is beneficial.

8 Conclusion

Named entity linking (NEL) is similar to the widely-studied problem of word sense disambiguation (WSD), with Wikipedia articles playing the role of WordNet synsets. Our contribution is to exploit unsupervised, graph-based approaches for NEL that have previously led to results rivalling supervised approaches for WSD [25].

Analysis of NEL searchers from the literature suggests that candidate recall is important. However, increasing recall is detrimental when it results in substantially increased ambiguity, because existing disambiguators do not perform well enough yet to overcome this additional ambiguity. This is highlighted by the fact that our tuned backoff approach, which achieves an 8% increase in candidate recall, still degrades performance on the end-to-end linking task due to the consequent increase in ambiguity.

Our results on the TAC 2010 data show another similarity to the WSD task: simple baselines lead to highly competitive results. Nevertheless, the best graph-based approach leads to a substantial increase of 4% over the cosine baseline.

Our final score of 85.5% is competitive with the best reported supervised approach (86.8%) and the best unsupervised approach (85.8%) to NEL. It incorporates document-wide link graph structure in linear time. This demonstrates that despite the different types of link information, NEL is indeed similar to WSD and can be accurately and efficiently performed using graph-based approaches.

References

1. Bagga, A., Baldwin, B.: Entity-based cross-document coreferencing using the vector space model. In: Proceedings of the 17th International Conference on Computational Linguistics. pp. 79–85. Montreal, Quebec, Canada (1998)
2. Barzilay, R., Elhadad, M.: Using lexical chains for text summarization. In: Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization. pp. 10–17. Madrid, Spain (1997)
3. Borgatti, S.: Identifying sets of key players in a network. In: Proceedings of the International Conference on Integration of Knowledge Intensive Multi-Agent Systems. pp. 127–131. Cambridge, MA, USA (2003)
4. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. In: Proceedings of the 7th International Conference on the World Wide Web. pp. 107–117. Brisbane, Australia (1998)

5. Bunescu, R., Paşca, M.: Using encyclopedic knowledge for named entity disambiguation. In: Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics. pp. 9–16. Trento, Italy (2006)
6. Cucerzan, S.: Large-scale named entity disambiguation based on Wikipedia data. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. pp. 708–716. Prague, Czech Republic (2007)
7. Curran, J.R., Clark, S.: Language independent NER using a maximum entropy tagger. In: Proceedings of the Seventh Conference on Natural Language Learning. pp. 164–167. Edmonton, Canada (2003)
8. de Vries, A.P., Vercoustre, A.M., Thom, J.A., Craswell, N., Lalmas, M.: Overview of the INEX 2007 entity ranking track. In: Proceedings of the 6th International Workshop of the Initiative for the Evaluation of XML Retrieval. pp. 245–251. Dagstuhl, Germany (2007)
9. Dredze, M., McNamee, P., Rao, D., Gerber, A., Finin, T.: Entity disambiguation for Knowledge Base Population. In: Proceedings of the 23rd International Conference on Computational Linguistics. pp. 277–285. Beijing, China (2010)
10. Fader, A., Soderland, S., Etzioni, O.: Scaling Wikipedia-based named entity disambiguation to arbitrary web text. In: Proceedings of the IJCAI Workshop on User-contributed Knowledge and Artificial Intelligence: An Evolving Synergy. pp. 21–26. Pasadena, CA, USA (2009)
11. Fellbaum, C. (ed.): WordNet: An Electronic Lexical Database. MIT Press, Cambridge, MA USA (1998)
12. Ferragina, P., Scaiella, U.: 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. pp. 1625–1628. Toronto, ON, Canada (2010)
13. Garoufi, K., Zesch, T., Gurevych, I.: Graph-theoretic analysis of collaborative knowledge bases in natural language processing. In: Proceedings of the 7th International Semantic Web Conference. Karlsruhe, Germany (2008)
14. Grishman, R., Sundheim, B.: Message Understanding Conference-6: a brief history. In: Proceedings of the 16th Conference on Computational Linguistics. pp. 466–471. Copenhagen, Denmark (1996)
15. Hobbs, J.R.: Pronoun resolution. Tech. rep., Department of Computer Science, City University of New York (1976)
16. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. In: Proceedings of the 9th annual ACM-SIAM Symposium on Discrete algorithms. pp. 668–677. San Francisco, CA, USA (1998)
17. Kulkarni, S., Singh, A., Ramakrishnan, G., Chakrabarti, S.: Collective annotation of wikipedia entities in web text. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 457–466. Paris, France (2009)
18. Lehmann, J., Monahan, S., Nezda, L., Jung, A., Shi, Y.: LCC approaches to knowledge base population at TAC 2010. In: Proceedings of the Text Analysis Conference. Gaithersburg, MD, USA (2010)
19. McNamee, P.: HLTCOE efforts in entity linking at TAC KBP 2010. In: Proceedings of the Text Analysis Conference. Gaithersburg, MD, USA (2010)
20. McNamee, P., Dang, H.T., Simpson, H., Schone, P., Strassel, S.M.: An evaluation of technologies for knowledge base population. In: Proceedings of the 7th International Conference on Language Resources and Evaluation. pp. 369–372. Valletta, Malta (2010)
21. Mihalcea, R.: Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing. pp. 411–418. Vancouver, BC, Canada (2005)

22. Mihalcea, R., Csomai, A.: Wikify!: linking documents to encyclopedic knowledge. In: Proceedings of the 16th ACM conference on Conference on Information and Knowledge Management. pp. 233–242. Lisbon, Portugal (2007)
23. Milne, D., Witten, I.H.: Learning to link with Wikipedia. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management. pp. 509–518. Napa Valley, CA, USA (2008)
24. Navigli, R., Lapata, M.: Graph connectivity measures for unsupervised word sense disambiguation. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence, pp. 1683–1688. Hyderabad, India (2007)
25. Navigli, R., Lapata, M.: An experimental study of graph connectivity for unsupervised word sense disambiguation. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(4), 678–692 (2010)
26. NIST: Task description for knowledge-base population at TAC 2010 (2010), accessed 20 August 2010 from `http://nlp.cs.qc.cuny.edu/kbp/2010/KBP2010_TaskDefinition.pdf`
27. Radford, W., Hachey, B., Nothman, J., Honnibal, M., Curran, J.R.: Cmcrc at tac10: Document-level entity linking with graph-based reranking. In: Proceedings of the Text Analysis Conference. Gaithersburg, MD, USA (2010)
28. Soon, W.M., Lim, D.C.Y., Ng, H.T.: A machine learning approach to coreference resolution of noun phrases. Computational Linguistics 27(4), 521–544 (2001)
29. Tjong Kim Sang, E.F.: Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In: Proceedings of the 6th Conference on Natural Language Learning. pp. 1–4. Taipei, Taiwan (2002)
30. Varma, V., Bysani, P., Reddy, K., Bharat, V., GSK, S., Kumar, K., Kovelamudi, S., N, K.K., Maganti, N.: IIIT Hyderabad at TAC 2009. In: Proceedings of the Text Analysis Conference. Gaithersburg, MD, USA (2009)
31. Zheng, Z., Li, F., Huang, M., Zhu, X.: Learning to link entities with knowledge base. In: Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics. pp. 483–491. Los Angeles, CA USA (2010)