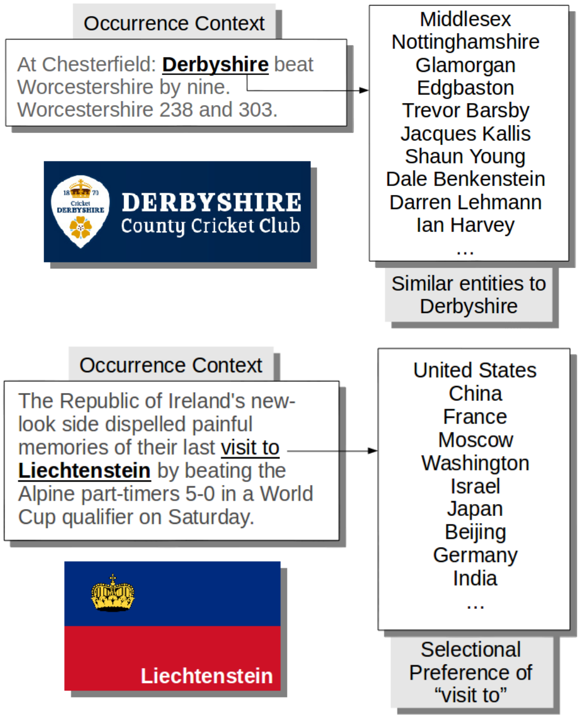
Alleviating Poor Context with Background Knowledge for Named Entity Disambiguation

Ander Barrena and **Aitor Soroa** and **Eneko Agirre**
IXA NLP Group
UPV/EHU University of the Basque Country
Donostia, Basque Country
ander.barrena,a.soroa,e.agirre@ehu.eus

Abstract

Named Entity Disambiguation (NED) algorithms disambiguate mentions of named entities with respect to a knowledge-base, but sometimes the context might be poor or misleading. In this paper we introduce the acquisition of two kinds of background information to alleviate that problem: entity similarity and selectional preferences for syntactic positions. We show, using a generative Näive Bayes model for NED, that the additional sources of context are complementary, and improve results in the CoNLL 2003 and TAC KBP DEL 2014 datasets, yielding the third best and the best results, respectively. We provide examples and analysis which show the value of the acquired background information.



At Chesterfield: **Derbyshire** beat Worcestershire by nine. Worcestershire 238 and 303.

The Republic of Ireland's new-look side dispelled painful memories of their last visit to **Liechtenstein** by beating the Alpine part-timers 5-0 in a World Cup qualifier on Saturday.

Figure 1: Two examples where NED systems fail, motivating our two background models: similar entities (top) and selectional preferences (bottom). The logos correspond to the gold label.

1 Introduction

The goal of Named Entity Disambiguation (NED) is to link each mention of named entities in a document to a knowledge-base of instances. The task is also known as Entity Linking or Entity Resolution (Bunescu and Pasca, 2006; McNamee and Dang, 2009; Hachey et al., 2012). NED is confounded by the ambiguity of named entity mentions. For instance, according to Wikipedia, *Liechtenstein* can refer to the micro-state, several towns, two castles or a national football team, among other instances. Another ambiguous entity is *Derbyshire* which can refer to a county in England or a cricket team. Most NED research use knowledge-bases derived or closely related to Wikipedia.

For a given mention in context, NED systems (Hachey et al., 2012; Lazic et al., 2015) typically rely on two models: (1) a mention module returns possible entities which can be referred to by the mention, ordered by prior probabilities; (2) a context model orders the entities according to the context of the mention, using features extracted from annotated training data. In addition, some systems check whether the entity is coherent with the rest of entities mentioned in the document, although (Lazic et al., 2015) shows that the coherence module is not required for top performance.

Figure 1 shows two real examples from the development dataset which contains text from News, where the clues in the context are too weak or misleading. In fact, two mentions in those examples (*Derbyshire* in the first and *Liechtenstein* in the second) are wrongly disambiguated by a bag-of-words context model.

In the first example, the context is very poor, and the system returns the *county* instead of the *cricket team*. In order to disambiguate it correctly one needs to be aware that *Derbyshire*, when occurring on News, is most notably associated with cricket. This background information can be acquired from large News corpora such as Reuters (Lewis et al., 2004), using distributional methods to construct a list of closely associated entities (Mikolov et al., 2013). Figure 1 shows entities which are distributionally similar to *Derbyshire*, ordered by similarity strength. Although the list might say nothing to someone not acquainted with cricket, all entities in the list are strongly related to cricket: Middlesex used to be a county in the UK that gives name to a cricket club, Nottinghamshire is a county hosting two powerful cricket and football teams, Edgbaston is a suburban area and a cricket ground, the most notable team to carry the name Glamorgan is Glamorgan County Cricket Club, Trevor Barsby is a cricketer, as are all other people in the distributional context. When using these similar entities as context, our system does return the correct entity for this mention.

In the second example, the words in the context lead the model to return the *football team* for *Liechtenstein*, instead of the *country*, without being aware that the nominal event "visit to" prefers locations arguments. This kind of background information, known as selections preferences, can be easily acquired from corpora (Erk, 2007). Figure 1 shows the most frequent entities found as arguments of "visit to" in the Reuters corpus. When using these filler entities as context, the context model does return the correct entity for this mention.

In this article we explore the addition of two kinds of background information induced from corpora to the usual context of occurrence: (1) given a mention we use distributionally similar entities as additional context; (2) given a mention and the syntactic dependencies in the context sentence, we use the selectional preferences of those syntactic dependencies as additional context. We test their contribution separately and combined, showing that they introduce complementary information.

Our contributions are the following: (1) we introduce novel background information to provide additional disambiguation context for NED; (2) we integrate this information in a Bayesian generative NED model; (3) we show that similar entities are useful when no textual context is present; (4) we show that selectional preferences are useful when limited context is present; (5) both kinds of background information help improve results of a NED system, yielding the state-of-the-art in the TAC KBP DEL 2014 dataset and getting the third best results in the CoNLL 2003 dataset; (6) we release both resources for free to facilitate reproducibility. [1]

The paper is structured as follows. We first introduce the method to acquire background information, followed by the NED system. Section 4 presents the evaluation datasets, Section 5 the development experiments and Section 6 the overall results. They are followed by related work, error analysis and the conclusions section.

2 Acquiring background information

We built our two background information resources from the Reuters corpus (Lewis et al., 2004), which comprises 250K documents. We chose this corpus because it is the one used to select the documents annotated in one of our gold standards (cf. Section 4). The documents in this corpus are tagged with categories, which we used to explore the influence of domains.

The documents were processed using a publicly available NLP pipeline, Ixa-pipes,[2] including tokenization, lematization, dependency tagging and NERC.

2.1 Similar entity mentions

Distributional similarity is known to provide useful information regarding words that have similar co-occurrences. We used the popular word2vec[3] tool to produce vector representations for named entities in the Reuters corpus. In order to build a resource that yields similar entity mentions, we took all entity-mentions detected by the NERC tool and, if they were multi word entities, joined them into a single token replacing spaces with underscores, and appended a tag to each of them. We run word2vec with default parameters on the preprocessed corpus. We only keep the vectors for named entities, but note that the corpus contains

[1] `http://ixa2.si.ehu.es/anderbarrena/2016ACL_files.zip`

[2] `http://ixa2.si.ehu.es/ixa-pipes/`

[3] `https://code.google.com/archive/p/word2vec/`

both named entities and other words, as they are needed to properly model co-occurrences.

Given a named entity mention, we are thus able to retrieve the named entity mentions which are most similar in the distributional vector space. All in all, we built vectors for 95K named entity mentions. Figure 1 shows the ten most similar named entities for *Derbyshire* according to the vectors learned from the Reuters corpus. These similar mentions can be seen as a way to encode some notion of a topic-related most frequent sense prior.

2.2 Selectional Preferences

Selectional preferences model the intuition that arguments of predicates impose semantic constraints (or preferences) on the possible fillers for that argument position (Resnik, 1996). In this work, we use the simplest model, where the selectional preference for an argument position is given by the frequency-weighted list of fillers (Erk, 2007).

We extract dependency patterns as follows. After we parse Reuters with the Mate dependency parser (Bohnet, 2010) integrated in IxaPipes, we extract $(H \xrightarrow{D} C)$ dependency triples, where D is one of the Subject, Object or Modifier dependencies[4] (SBJ, OBJ, MOD, respectively), H is the head word and C the dependent word. We extract fillers in both directions, that is, the set of fillers in the dependent position $\{C : (H \xrightarrow{D} C)\}$, but also the fillers in the head position $\{H : (H \xrightarrow{D} C)\}$. Each such configuration forms a template, $(H \xrightarrow{D} *)$ and $(* \xrightarrow{D} C)$.

In addition to triples (single dependency relations) we also extracted tuples involving two dependency relations in two flavors: $(H \xrightarrow{D_1} C_1 \xrightarrow{D_2} C_2)$ and $(C_1 \xleftarrow{D_1} H \xrightarrow{D_2} C_2)$. Templates and fillers are defined as done for single dependencies, but, in this case, we extract fillers in any of the three positions and we thus have three different templates for each flavor.

As dependency parsers work at the word level, we had to post-process the output to identify whether the word involved in the dependency was part of a named entity identified by the NERC algorithm. We only keep tuples which involve at least one name entity. Some examples for the three kinds of tuples follow, including the frequency of occurrence, with entities shown in bold:

(beat $\xrightarrow{SBJ}$ **Australia**) 141
(refugee $\xrightarrow{MOD}$ **Hutu**) 1681
(visit $\xrightarrow{MOD}$ to $\xrightarrow{MOD}$ **United States**) 257
(match $\xrightarrow{MOD}$ against $\xrightarrow{MOD}$ **Manchester United**) 12
(Spokesman $\xleftarrow{SBJ}$ tell $\xrightarrow{OBJ}$ **Reuters**) 1378
(**The Middle East** $\xleftarrow{MOD}$ process $\xrightarrow{MOD}$ peace) 1126

When disambiguating a mention of a named entity, we check whether the mention occurs on a known dependency template, and we extract the most frequent fillers of that dependency template. For instance, the bottom example in Figure 1 shows how *Liechtenstein* occurs as a filler of the template (visit $\xrightarrow{MOD}$ to $\xrightarrow{MOD}$ *), and we thus extract the selectional preference for this template, which includes, in the figure 1, the ten most frequent filler entities.

We extracted more than 4.3M unique tuples from Reuters, producing 2M templates and their respective fillers. The most frequent dependency was MOD, followed by SUBJ and OBJ [5] The selectional preferences include 400K different named entities as fillers.

Note that selectional preferences are different from dependency path features. Dependency path features refer to features in the immediate context of the entity mention, and are sometimes added as additional features of supervised classifiers. Selectional preferences are learnt collecting fillers in the same dependency path, but the fillers occur elsewhere in the corpus.

3 NED system

Our disambiguation system is a Näive Bayes model as initially introduced by (Han and Sun, 2011a), but adapted to integrate the background information extracted from the Reuters corpus. The model is trained using Wikipedia,[6] which is also used to generate the entity candidates for each mention.

Following usual practice, candidate generation is performed off-line by constructing an association between strings and Wikipedia articles, which we call dictionary. The association is performed using article titles, redirections, disambiguation pages, and textual anchors. Each association is scored with the number of times the string was

[4] Labels are taken from the Penn Treebank https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

[5] 1.5M, 0.8M and 0.7M respectively

[6] We used a dump from 25-5-2011. This dump is close in time to annotations of the datasets used in the evaluation (c.f. Section 4)

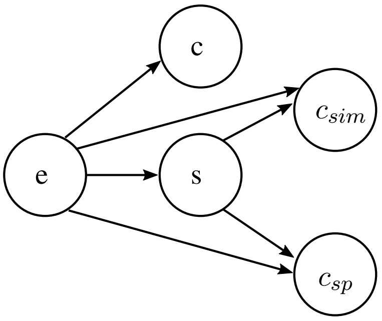


Figure 2: Dependencies among variables in our Bayesian network.

used to refer to the article (Agirre et al., 2015). We also use Wikipedia to extract training mention contexts for all possible candidate entities. Mention contexts for an entity are built by collecting a window of 50 words surrounding any hyper link pointing to that entity.

Both training and test instances are preprocessed the same way: occurrence context is tokenized, multi-words occurring in the dictionary are collapsed as a single token (longest matches are preferred). All occurrences of the same target mention in a document are disambiguated collectively, as we merge all contexts of the multiple mentions into one, following the one-entity-per-discourse hypothesis (Barrena et al., 2014).

The Näive Bayes model is depicted in Figure 2. The candidate entity e of a given mention s, which occurs within a context c, is selected according to the following formula:

$$e = \arg\max_e P(s, c, c_{\text{sp}}, c_{\text{sim}}, e) =$$
$$\arg\max_e P(e)P(s|e)P(c|e)P(c_{\text{sp}}|e, s)P(c_{\text{sim}}|e, s)$$

The formula combines evidences taken from five different probabilities: the entity prior $p(e)$, the mention probability $p(s|e)$, the textual context $p(c|s)$, the selectional preferences $P(c_{\text{sp}}|e, s)$ and the distributional similarity $P(c_{\text{sim}}|e, s)$. This formula is also referred to as the "**Full model**", as we also report results of partial models which use different combinations of the five probability estimations.

Entity prior $P(e)$ represents the popularity of entity e, and is estimated as follows:

$$P(e) \propto \frac{f(*, e) + 1}{f(*, *) + N}$$

where $f(*, e)$ is the number of times the entity e is referenced within Wikipedia, $f(*, *)$ is the total number of entity mentions and N is the number of distinct entities in Wikipedia. The estimation is smoothed using the *add-one* method.

Mention probability $P(s|e)$ represents the probability of generating the mention s given the entity e, and is estimated as follows:

$$P(s|e) \propto \theta \frac{f(s, e)}{f(*, e)} + (1 - \theta) \frac{f(s, *)}{f(*, *)}$$

where $f(s, e)$ is the number of times mention s is used to refer to entity e and $f(s, *)$ is the number of times mention s is used as anchor. We set the θ hyper-parameter to 0.9 according to developments experiments in the CoNLL testa dataset (cf. Section 5.5).

Textual context $P(c|e)$ is the probability of entity e generating the context $c = \{w_1, \ldots, w_n\}$, and is expressed as:

$$P(c|e) = \prod_{w \in c} P(w|e)^{\frac{1}{n}}$$

where $\frac{1}{n}$ is a correcting factor that compensates the effect of larger contexts having smaller probabilities. $P(w|e)$, the probability of entity e generating word w, is estimated following a bag-of-words approach:

$$P(w|e) \propto \lambda \frac{c(w, e)}{c(*, e)} + (1 - \lambda) \frac{f(w, *)}{f(*, *)}$$

where $c(w, e)$ is the number of times word w appears in the mention contexts of entity e, and $c(*, e)$ is the total number of words in the mention contexts. The term in the right is a smoothing term, calculated as the likelihood of word w being used as an anchor in Wikipedia. λ is set to 0.9 according to development experiments done in CoNLL testa.

Distributional Similarity $P(c_{\text{sim}}|e, s)$ is the probability of generating a set of similar entity mentions given an entity mention pair. This probability is calculated and estimated in exactly the same way as the textual context above, but replacing the mention context c with the mentions of the 30 most similar entities for s (cf. Section 2.1).

Selectional Preferences $P(c_{\text{sp}}|e, s)$ is the probability of generating a set of fillers c_{sp} given an entity and mention pair. The probability is again analogous to the previous ones, but using the filler entities of the selectional preferences of s instead of the context c (cf. Section 2.2). In our experiments, we select the 30 most frequent fillers for each selectional preferences, concatenating the filler list when more than one selectional preference is applied.

3.1 Ensemble model

In addition to the Full model, we created an ensemble system that combines the probabilities described above using a weighting schema, which we call "**Full weighted model**". In particular, we add an exponent coefficient to the probabilities, thus allowing to control the contribution of each model.

$$\arg\max_{e} P(e)^{\alpha} P(s|e)^{\beta}$$
$$P(c|e)^{\gamma} P(c_{\text{sp}}|e,s)^{\delta} P(c_{\text{sim}}|e,s)^{\omega}$$

We performed an exhaustive grid search in the interval $(0, 1)$ for each of the weights, using a step size of 0.05, and discarding the combinations whose sum is not one. Evaluation of each combination was performed in the CoNLL testa development set, and the best combination was applied in the test sets.[7]

4 Evaluation Datasets

The evaluation has been performed on one of the most popular datasets, the CoNLL 2003 named-entity disambiguation dataset, also know as the AIDA or CoNLL-Yago dataset (Hoffart et al., 2011). It is composed of 1393 news documents from Reuters Corpora where named entity mentions have been manually identified. It is divided in three main parts: *train*, *testa* and *testb*. We used *testa* for development experiments, and *testb* for the final results and comparison with the state-of-the-art. We ignored the training part.

In addition, we also report results in the Text Analysis Conference 2014 Diagnostic Entity Linking task dataset (TAC DEL 2014).[8] The gold standard for this task is very similar to the CoNLL dataset, where target named entity mentions have been detected by hand. Through the beginning of the task (2009 to 2013) the TAC datasets were query-driven, that is, the input included a document and a challenging and sometimes partial target-mention to disambiguate. As this task also involved mention detection and our techniques are sensitive to mention detection errors, we preferred to factor out that variation and focus on the 2014.

The evaluation measure used in this paper is micro-accuracy, that is, the percentage of linkable mentions that the system disambiguates correctly, as widely used in the CoNLL dataset. Note that TAC2014 EDL included several evaluation measures, including the aforementioned micro-accuracy of linkable mentions, but the official evaluation measure was Bcubed+ F1 score, involving also detection and clustering of mentions which refer to entities not in the target knowledge base. We decided to use the same evaluation measure for both datasets, for easier comparison. Table 1 summarizes the statistics of the datasets used in this paper where document and mention counts are presented.

Dataset	Documents	Mentions
CoNLL testa	216	4791
CoNLL testb	231	4485
TAC2014 DEL test	138	2817

Table 1: Document and linkable mention counts for CoNLL and TAC2014 DEL datasets.

5 Development experiments

We started to check the contribution of the acquired background information in the *testa* section of the CoNLL dataset. In fact, we decided to focus first on a subset of *testa* about sports,[9] and also acquired background information from the sports sub-collection of the Reuters corpus.[10] The rationale was that we wanted to start in a controlled setting, and having assumed that the domain of the test documents and the source of the background information could play a role, we decided to start focusing on the sports domain first. Another motivation is that we noticed that the ambiguity between locations and sport clubs (e.g. football, cricket, rugby, etc.) is challenging, as shown in Figure 1.

5.1 Entity similarity with no context

In our first controlled experiment, we wanted to test whether the entity similarity resource provided any added value for the cases where the target mentions had to be disambiguated out of context. Our hypothesis was that the background information from the unannotated Reuters collection, entity similarity in this case, should provide improved performance. We thus simulated a corpus where mentions have no context, extracting the named entity mentions in the sports subset that

[7] The best combination was $\alpha = 0.05$, $\beta = 0.1$, $\gamma = 0.55$ $\delta = 0.15$, $\omega = 0.15$

[8] http://www.nist.gov/tac/2014/KBP/

[9] Including 102 out of the 216 documents in *testa*, totaling 3319 mentions.

[10] Including approx. 35K documents out of the 250K documents in Reuters

Method	m-acc
$P(e)P(s\|e)$	63.83
$P(e)P(s\|e)P(c_{\text{sim}}\|e,s)$	70.98

Table 2: Results on mentions with no context on the sports subset of testa, limited to 85% of the mentions (cf. Section 5.1).

Method	m-acc
$P(e)P(s\|e)$	63.66
$P(e)P(s\|e)P(c\|e)$	66.18
$P(e)P(s\|e)P(c_{\text{sp}}\|e,s)$	67.33
$P(e)P(s\|e)P(c\|e)P(c_{\text{sp}}\|e,s)$	68.78

Table 3: Results on mentions with access to limited context on the sports subset of testa, limited to the 45% of mentions (cf. Section 5.2).

had an entry in the entity similarity resource (cf. Section 2.1), totaling 85% of the 3319 mentions.

Table 2 shows that the entity similarity resource improves the results of the model combining the entity prior and mention probability, similar to the so-called most frequent sense baseline (MFS). Note that the combination of both entity prior and mention probability is a hard-to-beat baseline, as we will see in Section 6. This experiment confirms that entity similarity information is useful when no context is present.

5.2 Selectional preferences with short context

In our second controlled experiment, we wanted to test whether the selectional preferences provided any added value for the cases where the target mentions had limited context, that of the dependency template. Our hypothesis was that the background information from the unannotated Reuters collection, selectional preferences in this case, should provide improved performance with respect to the baseline generative model of context. We thus simulated a corpus where mentions have only short context, exactly the same as the dependency templates which apply to the example, constructed extracting the named entity mentions in the sports subset that contained matching templates in the selectional preference resource (cf. Section 2.2), totaling 45% of the 3319 mentions.

Table 3 shows that the selectional preference resource (third row) allows to improve the results with respect to the no-context baseline (first row) and, more importantly, with respect to the baseline generative model (second row). The last row shows that the context model and the selectional preference model are complementary, as they produce the best result in the table. This experiment confirms that selectional preference information is effective when limited context is present.

Method	m-acc
$P(e)P(s\|e)P(c\|e)$	69.54
$P(e)P(s\|e)P(c\|e)P(c_{\text{sp}}\|e,s)$	71.25
$P(e)P(s\|e)P(c\|e)P(c_{\text{sim}}\|e,s)$	72.64
Full	73.94

Table 4: Results on mentions with limited context on the sports subset of testa, limited to the 41% of the mentions (cf. Section 5.3)

Models	Spor.	Reut.
$P(e)P(s\|e)$	65.52	65.52
$P(e)P(s\|e)P(c\|e)$	72.81	72.81
$P(e)P(s\|e)P(c\|e)P(c_{\text{sp}}\|e,s)$	73.56	73.06
$P(e)P(s\|e)P(c\|e)P(c_{\text{sim}}\|e,s)$	75.73	76.62
Full	76.30	76.87

Table 5: Results on the entire sports subset of testa: middle column uses the sports subset of Reuters to acquire background information, right column uses the full Reuters (cf. Section 5.4).

5.3 Combinations

In our third controlled experiments, we combine all three context and background models and evaluate them in the subset of the sports mentions that have entries in the similarity resource, and also contain matching templates in the selectional preference resource (41% of the sports subset). Note that, in this case, the context model has access to the entire context. Table 4 shows that, effectively, the background information adds up, with best results for the full combined model (cf. Section 3), confirming that both sources of background information are complementary to the baseline context model and between themselves.

5.4 Sports subsection of CoNLL testa

The previous experiments have been run on a controlled setting, limited to the subset where our constructed resources could be applied. In this section we report results for the entire sports subset of CoNLL testa. The middle column in Table 5 shows the results for the two baselines, and the improvements when adding the two background models, separately, and in combination. The results show that the improvements reported in the controlled experiments carry over when evaluating to all mentions in the Sport subsection, with an accumulated improvement of 3.5 absolute points over the standard NED system (second row).

The experiments so far have tried to factor out domain variation, and thus the results have been produced using the background information acquired from the sports subset of the Reuters collection. In order to check whether this control of the target domain is necessary, reproduced the same experiment using the full Reuters collection to build the background information, as reported in the rightmost column in Table 5. The results are very similar,[11] with a small decrease for selectional preferences, a small increase for the similarity resource, and a small increase for the full system. In view of these results, we decided to use the full Reuters collection to acquire the background knowledge for the rest of the experiments, and did not perform further domain-related experiments.

5.5 Results on CoNLL testa

Finally, Table 6 reports the results on the full development dataset. The results show that the good results in the sports subsection carry over to the full dataset. The table reports results for the baseline systems (two top rows) and the addition of the background models, including the Full model, which yields the best results.

In addition, the two rows in the bottom report the results of the ensemble methods (cf. Section 3.1) which learn the weights on the same development dataset. These results are reported for completeness, as they are an over-estimation, and are over-fit. Note that all hyper-parameters have been tuned on this development dataset, including the ensemble weights, smoothing parameters λ and θ (cf. Section 3), as well as the number of similar entities and the number of fillers in the selectional preferences. The next section will show that the good results are confirmed in unseen test datasets.

6 Overall Results

In the previous sections we have seen that the background information is effective improving the results on development. In this section we report the result of our model in the popular CoNLL testb and TAC2014 DEL datasets, which allow to compare to the state-of-the-art in NED.

System	**testa**
$P(e)P(s\|e)$	73.76
$P(e)P(s\|e)P(c\|e)$	78.98
$P(e)P(s\|e)P(c\|e)P(c_{\text{sp}}\|e,s)$	79.32
$P(e)P(s\|e)P(c\|e)P(c_{\text{sim}}\|e,s)$	81.76
Full	81.90
$P(e)^{\alpha}P(s\|e)^{\beta}P(c\|e)^{\gamma}$	85.20
Full weighted	**86.62**

Table 6: Results on the full testa dataset (cf. Section 5.5).

System	**CoNLL**	**TAC14**
$P(e)P(s\|e)$	73.07	78.31
$P(e)P(s\|e)P(c\|e)$	79.98	82.11
$P(e)P(s\|e)P(c\|e)P(c_{\text{sp}}\|e,s)$	81.31	82.61
$P(e)P(s\|e)P(c\|e)P(c_{\text{sim}}\|e,s)$	82.72	83.24
Full	82.85	83.21
$P(e)^{\alpha}P(s\|e)^{\beta}P(c\|e)^{\gamma}$	86.44	81.61
Full weighted	**88.32**	**83.46**

Table 7: Overall micro accuracy results on the CoNLL testb and TAC 2014 DEL datasets.

Table 7 reports our results, confirming that both background information resources improve the results over the standard NED generative system, separately, and in combination, for both datasets (Full row). All differences with respect to the standard generative system are statistically significant according to the Wilcoxon test (p-value $<$ 0.05).

In addition, we checked the contribution of learning the ensemble weights on the development dataset (testa). Both the generative system with and without background information improve considerably.

The error reduction between the weighted model using background information (Full weighted row) and the generative system without background information (previous row) exceeds 10% in both datasets, providing very strong results, and confirming that the improvement due to background information is consistent across both datasets, even when applied on a very strong system. The difference is statistically significant in both datasets.

[11] The two first rows do not use background information, and are thus the same.

System	CoNLL	TAC14
Full weighted	88.32	83.46
(Barrena et al., 2015)	83.61	80.69
(Lazic et al., 2015)	86.40	—
(Alhelbawy & Gaizauskas,14)	*87.60	—
(Chisholm and Hachey, 2015)	88.70	—
(Pershina et al., 2015)	*91.77	—
TAC14 best (Ji et al., 2014)	—	82.70

Table 8: Overall micro accuracy results on the CoNLL testb and TAC 2014 DEL datasets, including the current state-of-the-art. Starred results are not comparable, see text.

7 Related Work

Our generative model is based on (Han and Sun, 2011b), which is basically the core method used in later work (Barrena et al., 2015; Lazic et al., 2015) with good results. Although the first do not report results on our datasets the other two do. (Barrena et al., 2015) combines the generative model with a graph-based system yielding strong results in both datasets. (Lazic et al., 2015) adds a parameter estimation method which improved the results using unannotated data. Our work is complementary to those, as we could also introduce additional disambiguation probabilities (Barrena et al., 2015), or apply more sophisticated parameter estimation methods (Lazic et al., 2015).

Table 8 includes other high performing or well-known systems, which usually use complex methods to combine features coming from different sources, where our results are only second to those of (Chisholm and Hachey, 2015) in the CoNLL dataset and best in TAC 2014 DEL. The goal of this paper is not to provide the best performing system, but yet, the results show that our use of background information allows to obtain very good results.

Alhelbawy and Gaizauskas (2014) combines local and coherence features by means of a graph ranking scheme, obtaining very good results on the CONLL 2003 dataset. They evaluate on the full dataset, i.e. they test on train, testa and testb (20K, 4.8K and 4.4K mentions respectively). Our results on the same dataset are 84.25 (Full) and 88.07 (Full weighted), but note that we do tune the parameters on testa, so this might be slighly over-estimated. Our system does not use global coherence, and therefore their method is complementary to our NED system. In principle, our proposal for enriching context should improve the results of their system.

Pershina et al. (2015) propose a system closely resembling (Alhelbawy and Gaizauskas, 2014). They report the best known results on CONNL 2003 so far, but unfortunately, their results are not directly comparable to the rest of the state-of-the-art, as they artificially insert the gold standard entity in the candidate list.[12]

In (Chisholm and Hachey, 2015) the authors explore the use of links gathered from the web as an additional source of information for NED. They present a complex two-staged supervised system that incorporates global coherence features, with large amount of noisy training. Again, using additional training data seems an interesting future direction complementary to ours.

We are not aware of other works which try to use additional sources of context or background information as we do. (Cheng and Roth, 2013) use relational information from Wikipedia to add constraints to the coherence model, and is somehow reminiscent of our use dependency templates, although they focus on recognizing a fixed set of relations between entities (as in information extraction) and do not model selectional preferences. (Barrena et al., 2014) explored the use of syntactic collocations to ensure coherence, but did not model any selectional preferences.

Previous work on word sense disambiguation using selectional preference includes (McCarthy and Carroll, 2003) among others, but they report low results. (Brown et al., 2011) applied wordNet hypernyms for disambiguating verbs, but they did not test the improvement of this feature. (Taghipour and Ng, 2015) use embeddings as features which are fed into a supervised classifier, but our method is different, as we use embeddings to find similar words to be fed as additional context. None of the state-of-the-art systems, e.g. (Zhong and Ng, 2010), uses any model of selectional preferences.

8 Discussion

We performed an analysis of the cases where our background models worsened the disambiguation performance. Both distributional similarity and selectional preferences rely on correct mention detection in the background corpus. We detected

[12] https://github.com/masha-p/PPRforNED/readme.txt

that mentions where missed, which caused some coverage issues. In addition, the small size of the background corpus sometimes produces arbitrary contexts. For instance, subject position fillers of "score" include mostly basketball players like Michael Jordan or Karl Malone. A similar issue was detected in the distributional similarity resource. A larger corpus would produce a broader range of entities, and thus use of larger background corpora (e.g. Gigaword) should alleviate those issues.

Another issue was that some dependencies do not provide any focused context, as for instance arguments of *say* or *tell*. We think that a more sophisticated combination model should be able to detect which selectional preferences and similarity lists provide a focused set of instances.

9 Conclusions and Future Work

In this article we introduced two novel kinds of background information induced from corpora to the usual context of occurrence in NED: (1) given a mention we used distributionally similar entities as additional context; (2) given a mention and the syntactic dependencies in the context sentence, we used the selectional preferences of those syntactic dependencies as additional context. We showed that similar entities are specially useful when no textual context is present, and that selectional preferences are useful when limited context is present.

We integrated them in a Bayesian generative NED model which provides very strong results. In fact, when integrating all knowledge resources we yield the state-of-the-art in the TAC KBP DEL 2014 dataset and get the third best results in the CoNLL 2003 dataset. Both resources are freely available for reproducibility.[13]

The analysis of the acquired information and the error analysis show several avenues for future work. First larger corpora should allow to increase the applicability of the similarity resource, and specially, that of the dependency templates, and also provide better quality resources.

Acknowledgments

We thank the reviewers for their suggestions. This work was partially funded by MINECO (TUNER project, TIN2015-65308-C5-1-R). The IXA group is funded by the Basque Government (A type Research Group). Ander Barrena enjoys an PhD scholarship from the Basque Government.

References

Eneko Agirre, Ander Barrena, and Aitor Soroa. 2015. Studying the wikipedia hyperlink graph for relatedness and disambiguation. *CoRR*, abs/1503.01655.

Ayman Alhelbawy and Robert Gaizauskas. 2014. Graph ranking for collective named entity disambiguation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 75–80, Baltimore, Maryland, June. Association for Computational Linguistics.

Ander Barrena, Eneko Agirre, Bernardo Cabaleiro, Anselmo Peñas, and Aitor Soroa. 2014. "one entity per discourse" and "one entity per collocation" improve named-entity disambiguation. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2260–2269, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Ander Barrena, Aitor Soroa, and Eneko Agirre. 2015. Combining mention context and hyperlinks from wikipedia for named entity disambiguation. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 101–105, Denver, Colorado, June. Association for Computational Linguistics.

Bernd Bohnet. 2010. Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 89–97, Stroudsburg, PA, USA. Association for Computational Linguistics.

Susan Windisch Brown, Dmitriy Dligach, and Martha Palmer. 2011. Verbnet class assignment as a wsd task. In *Proceedings of the Ninth International Conference on Computational Semantics*, IWCS '11, pages 85–94, Stroudsburg, PA, USA. Association for Computational Linguistics.

R. C. Bunescu and M. Pasca. 2006. Using encyclopedic knowledge for named entity disambiguation. In *Proceesings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 9–16, Trento, Italy. The Association for Computer Linguistics.

X. Cheng and D. Roth. 2013. Relational inference for wikification. In *EMNLP*.

Andrew Chisholm and Ben Hachey. 2015. Entity disambiguation with web links. *Transactions of the Association for Computational Linguistics*, 3:145–156.

[13] `http://ixa2.si.ehu.es/anderbarrena/2016ACL_files.zip`

Katrin Erk. 2007. A simple, similarity-based model for selectional preferences. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 216–223, Prague, Czech Republic, June. Association for Computational Linguistics.

B. Hachey, W. Radford, J. Nothman, M. Honnibal, and J.R. Curran. 2012. Evaluating Entity Linking with Wikipedia. *Artificial Intelligence*, 194:130–150, January.

X. Han and L. Sun. 2011a. A generative entity-mention model for linking entities with knowledge base. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 945–954, Stroudsburg, PA, USA. Association for Computational Linguistics.

Xianpei Han and Le Sun. 2011b. A generative entity-mention model for linking entities with knowledge base. In *ACL HLT*.

J. Hoffart, M.A. Yosef, I. Bordino, H. Fürstenau, M. Pinkal, M. Spaniol, B. Taneva, S. Thater, and G. Weikum. 2011. Robust Disambiguation of Named Entities in Text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 782–792, Stroudsburg, PA, USA. Association for Computational Linguistics.

Heng Ji, Joel Nothman, and Ben Hachey. 2014. Overview of tac-kbp2014 entity discovery and linking tasks. In *Proc. Text Analysis Conference (TAC2014)*.

Nevena Lazic, Amarnag Subramanya, Michael Ringgaard, and Fernando Pereira. 2015. Plato: A selective context model for entity resolution. *Transactions of the Association for Computational Linguistics*, 3:503–515.

David D Lewis, Yiming Yang, Tony G Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. *The Journal of Machine Learning Research*, 5:361–397.

Diana McCarthy and John Carroll. 2003. Disambiguating nouns, verbs, and adjectives using automatically acquired selectional preferences. *Comput. Linguist.*, 29(4):639–654, December.

Paul McNamee and Hoa Dang. 2009. Overview of the TAC 2009 Knowledge Base Population track. In *TAC*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Maria Pershina, Yifan He, and Ralph Grishman. 2015. Personalized page rank for named entity disambiguation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 238–243, Denver, Colorado, May–June. Association for Computational Linguistics.

Philip Resnik. 1996. Selectional constraints: An information-theoretic model and its computational realization. *Cognition*, 61:127–159, November.

Kaveh Taghipour and Hwee Tou Ng. 2015. Semi-supervised word sense disambiguation using word embeddings in general and specific domains. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 314–323, Denver, Colorado, May–June. Association for Computational Linguistics.

Zhi Zhong and Hwee Tou Ng. 2010. It makes sense: A wide-coverage word sense disambiguation system for free text. In *ACL: System Demonstrations*.