Entity Disambiguation Using a Markov-Logic Network

Hong-Jie Dai[1,2] **Richard Tzong-Han Tsai**[3*] **Wen-Lian Hsu**[1,2*]

[1]Department of Computer Science, National Tsing Hua University,
300 No. 101, Section 2, Kuang-Fu Road, Hsinchu, Taiwan, R.O.C.
[2]Intelligent Agent Systems Lab., Institute of Information Science, Academia Sinica,
128 Academia Road, Sec.2, Nankang, Taipei, Taiwan, R.O.C.
[3]Department of Computer Science & Engineering, Yuan Ze University
135 Yuan-Tung Road, Chungli, Taoyuan, Taiwan, R.O.C.

hongjie@iis.sinica.edu.tw
thtsai@saturn.yzu.edu.tw
hsu@iis.sinica.edu.tw

Abstract

Entity linking (EL) is the task of linking a textual named entity mention to a knowledge base entry. It is a difficult task involving many challenges, but the most crucial problem is entity ambiguity. Traditional EL approaches usually employ different constraints and filtering techniques to improve performance. However, these constraints are executed in several different stages and cannot be used interactively. In this paper, we propose several disambiguation formulae/features and employ a Markov logic network to model interweaved constraints found in one type of EL, gene mention linking. To assess our systems effectiveness in different applications, we adopt two evaluation schemes: article-wide and instance-based precision/recall/F-measure. Experimental results show that our system outperforms the baseline systems and state-of-the-art systems under both evaluation schemes.

1 Introduction

Entity linking (EL) is the task of linking a textual named entity mention to a knowledge base (KB) entry, such as linking a person/organization mention to its Wikipedia entry (Kulkarni, Singh et al. 2009; McNamee and Dang 2009). This task has broad applications and is important in information extraction (IE) and text mining.

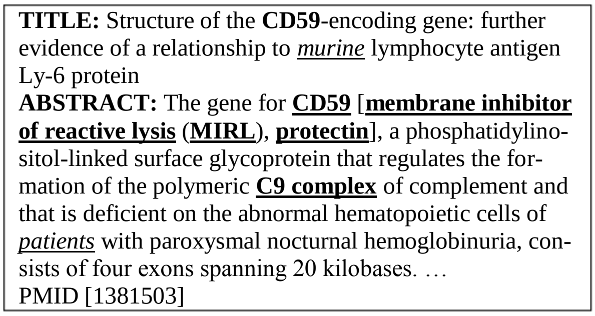
EL involves many challenges, but the most crucial problem in EL is *entity ambiguity*. Take the name John A. Smith for example. It might appear in KB as John Alexander Smith, John Blair Smith, John D. Smith, etc. Entity disambiguation determines which of all the possible John Smith KB entries a given “John Smith” refers to. Several disambiguation approaches have been proposed to address the entity ambiguity problem. For example, Dredze et al. (2010) formulated the disambiguation task as a ranking problem and developed features to link entities to Wikipedia entries. Zhang et al. (2010) used an automatically generated corpus to train a binary classifier to reduce ambiguities. Dai et al. (2010) collected external knowledge for each entity and calculated likelihoods stating the similarity of the current text with the knowledge to improve the disambiguation performance.

In addition to the entity ambiguity problem, the EL task in Text Analysis Conference (TAC) 2009 introduce the absence issue (McNamee and Dang 2009): for entities that have no corresponding entry in the KB a NIL should be returned. To deal with the absence issue, Bunescu and Pasca (2006) filtered out linked mentions whose scores are less than a fixed threshold. Li et al. (2009) trained a separate binary classifier to validate linked mentions. Dredze et al. (2010) treated the NIL as another KB entry candidate to train their EL ranking model.

Most previous works employed separate stages to execute NIL-filtering and disambiguation. However, a separate-stage approach ignores possible dependencies between NIL-filtering and disambiguation can result in error propagation. Figure 1 illustrates the problem. It shows a biomedical abstract, which discusses the relationship of the gene “CD59” to

[*]Corresponding author

Proceedings of the 5th International Joint Conference on Natural Language Processing, pages 846–855,
Chiang Mai, Thailand, November 8 – 13, 2011. ©2011 AFNLP



TITLE: Structure of the **CD59**-encoding gene: further evidence of a relationship to *murine* lymphocyte antigen Ly-6 protein
ABSTRACT: The gene for **CD59** [**membrane inhibitor of reactive lysis** (**MIRL**), **protectin**], a phosphatidylinositol-linked surface glycoprotein that regulates the formation of the polymeric **C9 complex** of complement and that is deficient on the abnormal hematopoietic cells of *patients* with paroxysmal nocturnal hemoglobinuria, consists of four exons spanning 20 kilobases. …
PMID [1381503]

Figure 1: An Example of Entity Linking.

other lymphocyte antigens. An EL system must link the human gene entity “CD59” and all its instances in the body of the abstract (including “membrane inhibitor of reactive lysis”, “protectin”, and “MIRL”, in the first sentence) to the EntrezGene database entry ID:966. However, the same name “CD59” in the title refers to a murine gene and, therefore, must be linked to ID:12509 instead. A separate-stage approach is likely to run into trouble with this example. In the EL stage, “MIRL” can be unambiguously linked to ID:996 with high confidence, because a search for the name in EntrezGene returns only one match. Linking the other mentions (e.g. “CD59” and “protectin”) to ID:996 is not as easy. For example, “CD59” alone has 18 candidate entries. However, because we know that these names are synonyms of MIRL, we can then link them more easily. Unfortunately, a separate NIL-filtering stage may filter out the entity mention “MIRL” because it is listed as an abbreviation of organization names, such as Mineral Industry Research Laboratory. With a joint inference process we can carry out both tasks simultaneously to avoid this type of error propagation (Poon and Domingos 2007).

Joint inference has become popular recently, because they make it possible for features and constraints to be shared among tasks. For example, Che and Liu (2010) created a joint model for word sense disambiguation (WSD) and semantic role labeling, and Finkel and Manning (2009) integrated parsing and named entity recognition in a joint model. In this paper, we use the Markov Logic Network (MLN) (Richardson and Domingos 2006), a joint model which combines first order logic (FOL) and Markov networks, to unify the NIL-filtering and entity disambiguation stages. The model captures the contextual information of the recognized entities for entity disambiguation as well as the constraints when linking an entity mention to a KB entry—for example, an entity mention can only be linked to a database entry when the mention has not been recognized as an NIL.

Another advantage of employing MLN in our EL disambiguation model is that it is easy to model arbitrary longer range dependencies among entities. For example, the saliency of a given entity in a given discourse is one property that can be modeled with MLN. This property was defined by Gale et al. (1992) in WSD, but has not been applied to EL disambiguation. It states that in a given discourse, there is precisely one entity that is the center of attention. This entity is mentioned over and over again, makes it more salient than others. We can utilize this phenomenon to improve the disambiguation confidence. Continuing with the example shown in Figure 1, ID:996 is a candidate KB entry for the entity “CD59” and all its instances, including “membrane inhibitor of reactive lysis”, “protectin”, and “MIRL” in the first sentence, we can then assume that ID:996 is more salient than other candidate KB entries. As described in the previous paragraph, we can link the entity “MIRL” to ID:996 with high confidence. Therefore, we are more likely to be able to link all the other entities to ID:996 as well.

Finally, existing EL approaches in biomedical domain assess system performance in terms of the effectiveness of database curation (Morgan, Wellner et al. 2007). In addition, we evaluate our system at a fine-grained resolution, entity by entity. Such an evaluation is more relevant to IE tasks such as the bio-molecular event extraction.

2 Markov Logic

In FOL, formulae are constructed using four types of symbols: constants, variables, functions, and predicates. *Constants* represent objects in a domain of discourse (e.g. people: **Ann**, **John**, etc., or database entries). *Variables* (e.g., x, y) range over the objects. *Predicates* represent the relations among objects (e.g. $correlates$), or attributes of objects (e.g. $hasTitle$). Variables and constants may be typed. An *atom* is a predicate symbol applied to a list of arguments, which may be variables or constants (e.g. $hasTitle(\mathbf{John}, x)$). A *ground atom* is an atom all of whose arguments are constants (e.g. $hasTitle(\mathbf{John}, \mathbf{Mr.})$). A *world* is an assignment of truth values to all possible ground atoms. A KB is a partial specification of a world; each atom in it is true, false or (implicitly) unknown.

A Markov network represents the joint distribution of a set of variables $X = (X_1, \dots, X_n) \in \mathcal{X}$ as a product of factors: $\mathrm{P}(\mathbf{X} = \mathbf{x}) = \frac{1}{Z}\prod_k f_k(\mathbf{x}_k)$, where each factor f_k is a non-negative function of a subset of the variables $\mathbf{x}_k$, and Z is a normalization constant. As long as $\mathrm{P}(\mathbf{X} = \mathbf{x}) > 0$ for all $\mathbf{x}$, the distribution can be equivalently represented as a log-linear model: $\mathrm{P}(\mathbf{X} = \mathbf{x}) = \frac{1}{Z}\exp(\sum_i w_i g_i(\mathbf{x}))$, where the features $g_i(\mathbf{x})$ are arbitrary functions of (a subset of) the variables' state.

An MLN L is a set of pairs (F_i, w_i), where F_i is a formula in FOL and w_i is a real number represented a weight. Together with a finite set of constants C, it defines a Markov network $M_{L,C}$, where $M_{L,C}$ contains one node for each possible grounding of each predicate appearing in L. The value of the node is 1 if the ground predicate is true, and 0 otherwise. The probability distribution over possible worlds x is given by $\mathrm{P}(\boldsymbol{X} = x) = \frac{1}{Z}\exp\left(\sum_{i \in F}\sum_{j \in G_i} w_i g_j(x)\right)$ where Z is the partition function, F is the set of all first-order formulae in the MLN, g_j is the set of groundings of the ith first-order formula, and $g_j(x) = 1$ if the jth ground formula is true and $g_j(x) = 0$ otherwise. General algorithms for inference and learning in Markov logic are discussed in Richardson and Domingos (2006). We employ thebeast toolkit (downloaded from http://code.google.com/p/thebeast/) to implement our MLN model. It uses 1-best MIRA online learning method for learning weights and employs cutting plane inference (Riedel 2008) with integer linear programming as its base solver for inference at test time and the MIRA online learning process.

3 Methods

In this paper, we tackle one type of EL, gene mention linking, which links each gene entity mention to one EntrezGene ID. EntrezGene (Maglott, Ostell et al. 2006) is the most popular large scale gene database. Generally speaking, the EL task can be separated into four steps: (1) identifying entities in a given article, (2) filtering NIL entity mentions, (3) finding candidate KB entries (or database IDs) for the remaining entity mentions, and (4) picking one KB entry for each entity mention. To concentrate on EL's major challenge, our MLN-based system only focuses on steps 2 and 4. In the following section, we define the main formulae used in our MLN-based EL system.

3.1 Linking Constraints Formulae

First, we illustrate a basic formula for the assumption that if an entity is mapped to only one KB entry, it should be linked to the entry:

Formula L.1

$$\exists!\, id.\, hasCandidate(i, id) \Rightarrow isLinkedTo(i, id)$$

where $hasCandidate(i, id)$ represents that the ith entity mention has a candidate entry id; and $isLinkedTo(i, id)$ represents that i is linked to id.

Note that we refer to an entity by its order in the article (e.g., the ith entity) for several reasons. One, not all names can be found in the training data. Secondly, even if two entities have the same surface string, they may link to different KB entries. Lastly, this allows us to model the order information and dependency among all entities.

Because the objective of the EL task is to discover a unique KB entry for each entity, we must define a formula to ensure that the constraint is satisfied. We use the following formula to prevent an entity associating with more than two entries.

Formula L.2 $isLinkedTo(i, id_1) \wedge id_1 \neq id_2 \Rightarrow \neg isLinkedTo(i, id_2)$

Formula L.2 is a hard constraint that must always hold whereas the others are soft and can be violated.

3.2 Saliency Formula

The saliency property can be written as follows:

Formula S.1: Saliency:

$$i < j \wedge isLinkedTo(i, id) \wedge hasCandidate(j, id) \Rightarrow isLinkedTo(j, id)$$

In other words, if the KB entry id is linked to an entity i that precedes the current mention j, and id is a candidate KB entry of j, then the current entity j should also be linked to id.

3.3 Disambiguation Formulae

As shown in Table 1, there are numerous observed predicates defined for the disambiguation process. Before diving into the details of all the formulae, we summarize the basic idea and describe how one could apply it to other EL tasks.

In our disambiguation approach, we rely on background knowledge k, such as an entity's inhabited location, or an entity's skill or functionality. k describes various aspects of the entity i's ambiguous KB entry, id. Whenever the entity is discussed, some of these aspects will be mentioned as well. Using k, we can write formulae like the following for disambiguation:

	hasCandidate(*i*, *id*)
	hasChromosomeInfo(*i*, *id*, *sd*) *hasWord*(*w*): the abstract contain a word *w*. *PPIKeyword*(*w*), *isPPIPartner*(*id*$_1$, *id*$_2$) *hasPPIPartnerRank* (*i*, *id*, *r*), *hasGOTermRank*(*i*, *id*, *r*), *hasTissueTermRank*(*i*, *id* ,*r*) *hasPrecedingWord*(*i*, *w*, *l*), *hasFollowingWord*(*i*, *w*, *l*) *hasUnigramBetween*(*i*, *j*, *w*)
Variable Type	*i*: an integer, which refers to the *i*th gene mention in the given article (similarly *j* refers to the *j*th mention) *id*: an EntrezGene ID, which refers to a linked KB entry. *sd*: an integer, which refers to the sentence distance. *w*: a word. *r*: an integer, which refers to the rank of the matching. *l*: an integer, which refers to a context window length.

Table 1: Main Predicates for Disambiguation.

$$Content\_Knowledge(i, id, k) \Rightarrow isLinkedTo(i, id)$$

The formula shows that if the context of the entity i, which has a linking candidate KB entry id, contains the background knowledge information k, the entity i should be linked to id.

In this paper, we define four predicates to capture the recognized genes' background information, including chromosome location, protein-protein interaction (PPI), tissue type and gene ontology. For example, the predicate $hasChromosomeInfo(i, id, sd)$ indicates that the chromosome location information of the i th entity, which has the KB entry id as its linked candidate entry, can be found in the surrounding text in the range sd. Applying this predicate to the sentence: "The human **<u>UBQLN3</u>** gene was mapped to the *11p15* region of chromosome 11," the entity mention UBQLN3 must be linked to the KB entry ID:50613 because 50613's chromosome location, *11p15*, is found in the same sentence. The formula describing the relation of $hasChromosomeInfo$ and $isLinkedTo$ is defined as follows:

$$\begin{aligned} &hasChromosomeInfo(i, id, +sd) \\ &\qquad \Rightarrow isLinkedTo(i, id) \end{aligned}$$

Here, we can see that there is an additional parameter ($+sd$) in $hasChromosomeInfo$. *sd* indicates where the chromosome information corresponding to *id* locates. The "+" notation in the above formula indicates that we must learn a separate weight for each grounded variable (*sd*). For example, $hasChromosomeInfo(i, id, 0)$ and $hasChromosomeInfo(i, id\ ,1)$ are given two different weights in our MLN model after training.

Correlation information, such as "KB entry id_1 usually interacts with an entry id_2", can be used in disambiguating an entity i as follows. The PPI information recorded in the database can provide the correlation information. Based on the correlation information as well as the candidate KB entry distribution in the current context, the id that correlated with the most unambiguous entries is the most likely id to be linked to i. We define the predicate $hasPPIPartnerRank(i, id, r)$ to represent this concept. The formula defining the relationship between $hasPPIPartnerRank$ and $isLinkedTo$ is:

$$\begin{aligned} \exists!\, id.\, &hasPPIPartnerRank(i, id, 1) \\ &\land \exists w.\, hasWord(w) \\ &\land PPIKeyword(w) \\ &\Rightarrow isLinkedTo(i, id) \end{aligned}$$

One can see that there are two predicates, $hasWord$ and $PPIKeyword$, in this formula that check if the article contains PPI keywords. Two similar predicates, $hasGOTermRank$ and $hasTissueTermRank$, represent the concept that *i* should be linked to the *id* with the largest number of corresponding gene ontology terms (entity's function) or tissue terms (entity's location) found in the context.

For the correlation information, we further define the following formula to capture the dependency that an entity j should be linked to id_2 if another entity i has been linked to id_1, and id_1 forms a correlation with id_2:

Formula D.1

$$\begin{aligned} \exists w.\, hasWord(w) \land\, &PPIKeyword(w) \\ &\land isLinkedTo(i, id_1) \\ &\land hasCandidate(j, id_2) \\ &\land isPPIPartner(id_1, id_2) \\ &\Rightarrow isLinkedTo(j, id_2) \end{aligned}$$

This formula shows another long range dependency among entities used in our MLN model (The first long dependency formula is S.1).

Finally, an entity mention j may sometimes be followed by its variant i (abbreviation or full name). Usually, the variant i is put in parentheses. If i can be uniquely linked to id, it is very likely that j is also linked to id. An example formula is shown as follows:

$$\begin{gathered} hasPrecedingWord(i, "(", 1) \land \\ hasFollowingWord(i, ")", 1) \land \\ \exists!\, u.\, hasUnigramBetween(i, j, u) \land \\ \exists!\, id.\, hasCandidate(i, id) \land u = "(" \Rightarrow \\ isLinkedTo(j, id) \end{gathered}$$

3.4 NIL-filtering Formulae

We approach the absence issue by filtering the following mention type: those belonging to classes that are not in the database curation target; called NILs. In linking gene mentions to KB entries, NILs appear when the gene mentions are protein families or complexes, or in a specific organism that is not considered. For example, in Figure 1, an EL system must return NIL for "C9

	hasName(*i*, *n*) *hasFirstWord*(*i*, *w*), *hasLastWord*(*i*, *w*), *isBlacklisted*(*w*): the word sequence *w* is blacklisted, *containsMoreSpecificMentions*(*i*): the *i*-th gene mention collocates with more specific gene mentions in the current context. *SpeciesTerm*(*w*), *AllUpperCase*(*i*), *hasPartOfSpeech*(*i*, *k*, *p*)
Variable Type	*n*: a word or a sequence of words that refer to the surface name of a gene mention. *ch*: characters. *d*: an integer. *k*: the *k*th index of the gene mention. *p*: a part-of-speech

Table 2: Main Predicates for NIL-filtering.

complex" in the first sentence, because the mention is a protein complex that is not EntrezGene's curation target. The predicate $isSuitableForLinking$ describes this concept.

We then employ the following formula to ensure that, whenever the ith entity is linked to a KB entry id, it must be an entity suitable for linking.

$$isLinkedTo(i, id) \Rightarrow isSuitableForLinking(i)$$

The formula models the stage decisions determined by traditional EL systems. The KB entry id does not have to be linked to the entity i proposed by the entity recognition/classification stage; however, the id cannot be assigned to the ith gene mention that has not been proposed as a potential entity. The formula is a hard constraint.

The initial formula containing $isSuitableForLinking$ treats any entity i with surface name n as a potential entity:

$$hasName(i, +n) \Rightarrow isSuitableForLinking(i)$$

Again, the "+" notation in the above formula indicates that the MLN must learn a separate weight for each entity name n.

In person name EL, for example, one could define $hasTitle$ to indicate that a title, such as Mr. or Mrs., appears in the ith entity's context and apply the formula for the $isSuitableForLinking$ predicate:

$$hasTitle(i, +t) \Rightarrow isSuitableForLinking(i).$$

In our work, we construct our formulae by using the observed predicates defined in Tables 1 and 2 to determine whether or not i is a NIL by checking i's context. For example:

$$hasFirstWord(i, +w) \wedge SpeciesTerm(+w) \Rightarrow isSuitableForLinking(i)$$

implies that a certain gene mention i's suitability for linking depends on whether or not i's first word is a certain species keyword w.

4 Results

4.1 Evaluation Metrics

We use two metrics to evaluate our approach and compare it with other EL disambiguation methods. Both use the standard precision, recall, and F-measure metrics (PRF) at two resolutions (article and instance).

Article-wide evaluation is based on the standard used in the BioCreative challenge (Morgan, Lu et al. 2008), which was designed to determine an EL system's performance as an aid for the curation of biological databases. The EL system outputs a list of EntrezGene IDs for a given article. The list is then compared to the gold standard IDs list for the article. The PRF scores are calculated based on the sums of true/false positives/negatives (TP, TN, FP, FN).

Instance-based evaluation measures the EL performance at a fine-grained IE resolution, which can support the development of advanced IE tasks. In contrast to the first metric, the PRF scores are calculated based on the sums of TP, TN, FP and FN for all instances in the test dataset; we further consider whether the boundary matches that of the linked KB entry's mention. Therefore, under this criterion, an FP could link a true entity to the wrong KB entry or link a false entity to any KB entry; while an FN could link a true entity to the wrong KB entry or fail to recognize a true entity. In cases where a true entity is linked to the wrong KB entry, both the FN and FP are increased by 1.

For TP/FP/FN, we need to determine when the predicted boundary matches that of the gold standard. Most entity recognition tasks use "exact-matching" as the primary criterion. Under this criterion, a candidate entity can only be counted as a TP if both its left and right boundaries fully coincide with the gold answer. However, in a real scenario, a gene mention can be tagged in several ways (e.g., "… between serum <entity>LH</entity>" and "… between <entity>serum LH</entity>" are both correct). Furthermore, for the EL task, the correctness of the linked KB entry is more important than its boundaries. Therefore, we use approximate-matching (Subramaniam, Mukherjea et al. 2003) to determine the boundary criterion. For example, a TP is counted when a machine-linked gene mention is a substring of the gold standard-linked gene mention or vice versa, and the linked KB entry is equal to the gold entry.

4.2 Dataset

In the experiments we used the training and test sets released by the BioCreative II gene normalization (GN) task (Morgan, Lu et al. 2008). The dataset provides a list of human gene EntrezGene database entries (IDs) that appear in each abstract. Although the gold standard answers contain each EntrezGene database entry's surface name shown in the abstract, they do not give the exact location of the corresponding entity mention in the abstract. So the original BioCreative II dataset can only be used to evaluate article-wide EL performance.

To obtain instance-based evaluation results, we need to expand the original annotation of the dataset. Our in-lab biologists annotated the exact locations and boundaries of the KB entries' gene mentions in a semi-automated manner. The automated annotation process used the entry's surface name provided by the gold standard to tag the entire corpus. Human annotators then corrected the recognized entities and linked results based on the context.

To compile the EL training corpus for our EL models, we employed a state-of-the-art gene mention linking system released by Lai et al. (2009) to recognize all gene mentions and generate candidate KB entries for each entity. For each mention *m* in a sentence *s* recognized by Lai's system and the set of KB entry candidates for *m* output by Lai's system, we searched *s* for the first human annotated mention *n* overlapping with *m* and set *n*'s KB entry as *m*'s true KB entry. Other candidates were set as *m*'s incorrect KB entries.

For the NIL-filtering corpus (NIL corpus), again, for each mention *m* in a sentence *s* recognized by Lai's system, we checked whether or not the boundary of the mention *m* match with the human annotated boundary. All matched mentions are regarded as true positives while the others are negative instances.

4.3 Model Configurations

For our experiments, we constructed four MLN-based configurations. In addition, separate-stage models for NIL-filtering and disambiguation were also constructed.

MLN$_{\textbf{LINK}}$: To assess baseline performance without disambiguation and NIL-filtering, we constructed an MLN-based configuration, MLN$_{\text{LINK}}$, only employing formulae related to linking constraints (refer to section 3.1). We compared its performance with that of a modified version of Lai's system, for which all mentions with only one KB entry were directly treated as answers, and entities with more than one candidate KB entry were discarded. This system is referred to as **Lai$_{\textbf{NO\_DIS}}$**.

MLN$_{\textbf{SAL}}$: To assess the performance gain from the saliency property, we constructed a second MLN-based configuration, MLN$_{\text{SAL}}$, by adding the formula corresponding to the saliency property (Formula S.1) to the MLN$_{\text{LINK}}$ configuration.

MLN$_{\textbf{DIS}}$: This configuration uses the constraints defined in Section 3.1, the saliency property in Section 3.2, and the disambiguation formulae defined in Section 3.3. We compared it with two other previous approaches: Lai et al.'s rule-based approach and Crim et al.'s (2005) supervised learning approach, which treated the EL disambiguation task as a classification problem and solved by employing the maximum entropy (ME) model. For Lai et al.'s approach, denoted as **Lai$_{\textbf{DIS}}$**, we directly employed Lai et al.'s original system, which had been well-tuned on the BioCreative II GN dataset. One can refer to their work (Lai, Bow et al. 2009) for more details. For the supervised learning approach, denotes as **ME$_{\textbf{DIS}}$**, we transformed the formulae described in Section 3.3 to binary feature functions in ME.

MLN$_{\textbf{JOINT}}$: The final MLN-based configuration (MLN$_{\text{JOINT}}$) was constructed by adding the NIL-filtering formulae to MLN$_{\text{DIS}}$. That is, all formulae introduced in Section 3 were employed.

ME$_{\textbf{NF}}$ for separate NIL-filtering: To simulate and compare the separate-stage NIL-filtering and EL disambiguation approach with MLN$_{\text{JOINT}}$, we developed a separate NIL-filtering model for Lai$_{\text{DIS}}$ and ME$_{\text{DIS}}$, denoted as **ME$_{\textbf{NF}}$**. The ME$_{\text{NF}}$ model was executed before the disambiguation step. We formulated the NIL-filtering task as a classification problem and used the features equivalent to the formulae described in Section 3.4 to train a ME-based classifier.

We employed the greedy backward sequential selection algorithm (Aha and Bankert 1995) to select the optimized feature sets for ME$_{\text{NF}}$ with ten-fold cross validation on the training dataset. The algorithm starts from all features transformed from NIL-filtering formulae and repeatedly removes a feature whose removal yields the maximal performance improvement in the overall EL task. Note that the feature selection procedure is designed for optimizing the performance of EL not NIL-filtering. We will discuss this in Section 5.3.

In the next sub-section, we discuss the instance-based IE results. Then, we derive BioCreative's evaluation results by merging the linked KB entries in all indexes and removing duplicated entries.

4.4 Experiment Results

Table 3 shows the instance-based and article-wide results evaluated on the test set. The first three columns show each system's PRF. The last column shows the relative advantage of F-score over the rule-based baseline (Lai$_{\text{NO\_DIS}}$).

In the second row, we can see that MLN$_{\text{LINK}}$ achieves exactly the same performance as Lai$_{\text{NO\_DIS}}$, indicating that the MLN-based system can simulate Lai$_{\text{NO\_DIS}}$ by only employing the linking constraints. In the third row, we can observe that, by adding S.1, the recall rate is improved by 2.7% which results in an improved F-score. This shows that the saliency property is effective in instance-based evaluation. However, MLN$_{\text{SAL}}$ performs slightly worse than MLN$_{\text{LINK}}$ in the article-wide evaluation, the reason for which is explained in Section 5.1.

From the fourth to the sixth row, we can see that MLN outperforms the other two models. Adding disambiguation formulae also further boost the EL performance in both instance-based and article-wide evaluations by an apparent large margin (3.6% and 13.7%).

Finally, in the last configuration set (7th, 8th, and 9th row), we can see that MLN$_{\text{JOINT}}$ does better than the compared separate-stage methods under both evaluation metrics. MLN$_{\text{JOINT}}$ also achieves the best performance among all MLN-based models under both instance-based and article-wide evaluations.

5 Discussion

5.1 Model Long-range Dependencies

One advantage of employing MLN in our EL modeling is that it is easy to model arbitrary longer range dependencies, as expressed by the formula S.1 and D.1. It is difficult to model such dependencies using ME. As shown in Table 3 (MLN$_{\text{SAL}}$), adding the S.1 dependency improves instance-based EL performance without any domain knowledge. A similar result is achieved by adding D.1 with PPI—instance-based performance can be improved by 1.1%/0.6% on the training/test set, respectively. We also observe that adding Linking formulae with S.1 reduces the recall rate in the article-wide evaluation. According to our analysis, S.1 improves the recall in the instance-based evaluation. In contrast, for article-wide, S.1 slightly reduces the recall. This is because after adding S.1, mentions tend to be linked to "salient" KB entries. In instance-based evaluation, the salient KB entries have higher frequency; therefore, the improvement of linking salient entries can cover the losses caused by disregarding the un-salient entries. However, in the article-wide evaluation, all entries in an article are counted equally; therefore, the improvement of instance-based evaluation does not transfer to article-wide evaluation.

Metrics	Instance-based (%)				Article-wide (%)			
Config.	P	R	F	Adv	P	R	F	Adv
Lai$_{\text{NO\_DIS}}$	80.7	56.3	66.3	-	77.3	71.5	74.3	-
MLN$_{\text{LINK}}$	80.7	56.3	66.3	-	77.3	71.5	74.3	-
MLN$_{\text{SAL}}$	79.5	59.0	67.7	+2.1	77.2	71.3	74.1	-0.2
Lai$_{\text{DIS}}$	72.9	63.9	68.1	+2.7	82.6	83.4	83.0	+11.7
ME$_{\text{DIS}}$	79.2	58.2	67.1	+1.2	88.8	79.0	83.6	+12.5
MLN$_{\text{DIS}}$	73.8	64.3	68.7	+3.6	86.1	83.0	84.5	+13.7
ME$_{\text{NF}}$+Lai$_{\text{DIS}}$	73.7	64.2	68.7	+3.6	84.1	83.7	83.9	+12.9
ME$_{\text{NF}}$+ME$_{\text{DIS}}$	80.2	58.4	67.6	+2.0	90.2	79.0	84.3	+13.4
MLN$_{\text{JOINT}}$	77.5	63.7	70.0	+5.6	87.7	83.8	85.7	+15.3

Table 3: Instance-/Article-wide Results on the BioCreative Test Set.

5.2 Linking to Multiple KB Entries

Another advantage of our model is its flexibility. The EL task is usually defined as linking an entity to a unique KB entry. However, in some cases, there are entities that cannot be uniquely linked. For example, the "ABCB9 protein" in the sentence "**ABCB9 protein** appears to be most highly expressed in the Sertoli cells of the seminiferous tubules in *mouse and rat testes*."

The EL system cannot link each of the gene mentions in the above sentences to just one EntrezGene KB entry. Our model can deal with the issue easily by modifying the constraint in L.2 with a larger cardinality, or introducing additional formulae to determine the cardinal constraint dynamically.

5.3 Joint Model vs. Separate-stage Models

Compared with the two separate-stage approaches, MLN$_{\text{JOINT}}$ has the following two advantages: (1) it performs several functions using one model, and (2) it finds the optimal solution for the integrated stages. The first advantage has been illustrated by Meza-Ruiz et al. (2009). This is to be contrasted with separate-stage systems where several components need to be trained and integrated by different strategies.

The second advantage is based on our observation on the training set, employing all features transformed from NIL-filtering formulae in the ME_{NF} model might be able to achieve the best NIL-filtering performance, but it does not guarantee that the final integrated EL performance can also be the best. This is the reason why we need to employ the backward feature selection algorithm to optimize EL performance for the separate-stage systems as described in Section 4.3.

5.4 Boundary Issue in EL

Our experiment results raise an interesting question: What causes absolute score differences between the instance-based and article-wide evaluations. Several works have studied the boundary issue in entity recognition (Finkel, Dingare et al. 2005; Tsai, Wu et al. 2006). We observe that the issue also has a significant effect on the performance of EL. For example, consider the following sentence in the training set (PMID: 9346890): "<entity id=3083>Hepatocyte growth factor (HGF) activator</entity> is a serine protease responsible for proteolytic activation of <entity id=3082>HGF</entity> in response to tissue injury"

We found that Lai et al.'s system and the three publically available gene mention recognition systems[1] all separate the first gene mention (ID:3083) into at least one mention, ("hepatocyte growth factor" or "HGF"). The incorrect boundary leads to errors in EL, and could result in the extraction of an incorrect self-activation event: <entity id=3082>HGF</entity> activates <entity id=3082>HGF</entity>. An experiment conducted on the test set shows that our MLN model can achieve an F-score 79.4% in instance-based evaluation if we replaced the predicted mentions' boundaries with their corresponding overlapping gold boundaries.

5.5 Related Works

The EL problem comes up in many fields of research. Among database researchers, this problem is described as data de-duplication (Sarawagi and Bhamidipaty 2002). In the AI community, the same EL problem is described as entity resolution (Singla and Domingos 2006; Bhattacharya and Getoor 2007). In the biomedical field, term identification (Krauthammer and Nenadic 2004) or normalization (Hirschman, Colosimo et al. 2005) are used to refer to the same concepts.

Several approaches had been proposed to deal with EL tasks. The following four points explain the distinctiveness of our work. The first is that, as the names of different database entries might be identical, techniques based purely on character/token-based similarity metrics do not work well. The second is that, in our task, all KB entries in the database are unique while it is not true in the tasks tackled by previous works. Third, in our EL task, measuring the similarity between a KB entry and an entity mention requires comparing their related information (fields). The former's can be acquired from the KB while the latter's should be extracted from its context. The latter's is hard to extract because the context is written in natural language, which is unstructured and some information may not appear in the context. This phenomena is against the assumption most previous EL approaches (Fellegi and Sunter 1969; Elmagarmid, Ipeirotis et al. 2007) based on: entities should have the same set of fields. Finally, as described in Section 1, our work needs to consider the absence issue to filter out NIL entities, which are not considered in the most previous EL approaches.

6 Conclusion

In this paper, we present a novel approach that employs MLN to model the constraints and decisions in the EL task. The contribution of this paper is threefold. First, we propose several features for EL disambiguation and NIL-filtering and demonstrate a feasible approach for modeling them by using an MLN. Second, unlike existing EL disambiguation approaches, which do not model the dependencies among entities, the proposed approach learns to model the dependencies, including the saliency and interaction among KB entries, and the performance improvement is promising. Third, we describe a new evaluation scheme that use the BioCreative corpus to analyze EL tasks from an additional perspective, instance-based evaluation, which have not yet been applied in the EL field thus far.

Acknowledgement

This research was supported in part by the National Science Council under grant NSC99-3112-B-001-005 and NSC98-2221-E-155-060-MY3, the Academia Sinica Investigator Award 95-02 and the research center for Humanities and Social Sciences under grant IIS-50-23.

[1] ABNER, GENIA tagger and BANNER.

Reference

Bhattacharya, I. and L. Getoor. 2007. Collective entity resolution in relational data. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1): 5.

Bunescu, R and M Pasca. 2006. Using encyclopedic knowledge for named entity disambiguation. In: *European Chapter of the Assocation for Computational Linguistics*.

Che, Wanxiang and Ting Liu. 2010. Jointly Modeling WSD and SRL with Markov Logic. In: *Proceedings of the 23rd International Conference on Computational Linguistics*, Beijing, China.

Crim, Jeremiah, Ryan McDonald and Fernando Pereira. 2005. Automatically Annotating Documents with Normalized Gene Lists. *BMC Bioinformatics*, 6(Suppl 1): S13.

Dai, Hong-Jie, Po-Ting Lai and Richard Tzong-Han Tsai. 2010. Multistage Gene Normalization and SVM-Based Ranking for Protein Interactor Extraction in Full-Text Articles. *IEEE TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS*, 7(3): 412-420.

Dredze, Mark, Paul McNamee, Delip Rao, Adam Gerber and Tim Finin. 2010. Entity Disambiguation for Knowledge Base Population. In: *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, Beijing.

Elmagarmid, Ahmed K., Panagiotis G. Ipeirotis and Vassilios S. Verykios. 2007. Duplicate Record Detection: A Survey. *IEEE Trans. on Knowl. and Data Eng.*, 19(1): 1-16.

Fellegi, I.P. and A.B. Sunter. 1969. A theory for record linkage. *Journal of the American Statistical Association*, 64(328): 1183-1210.

Finkel, Jenny, Shipra Dingare, Christopher Manning, Malvina Nissim, Beatrice Alex and Claire Grover. 2005. Exploring the boundaries: gene and protein identification in biomedical text. *BMC Bioinformatics*, 6(Suppl 1): S5.

Finkel, Jenny Rose and Christopher D. Manning. 2009. Joint parsing and named entity recognition. In: *Proceedings of NAACL 2009*.

Gale, WA, KW Church and D Yarowsky. 1992. A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26(5): 415-439.

Hirschman, Lynette, Marc Colosimo, Alexander Morgan and Alexander Yeh. 2005. Overview of BioCreAtIvE task 1B: normalized gene lists. *BMC Bioinformatics*, 6(Suppl 1): S11.

Krauthammer, Michael and Goran Nenadic. 2004. Term identification in the biomedical literature. *Journal of Biomedical Informatics*, 37(6): 512-526.

Kulkarni, Sayali, Amit Singh, Ganesh Ramakrishnan and Soumen Chakrabarti. 2009. Collective annotation of wikipedia entities in web text. In: *Proceeding of the 17th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)* Paris, France, ACM.

Lai, Po-Ting, Yue-Yang Bow, Chi-Hsin Huang, Hong-Jie Dai, Richard Tzong-Han Tsai and Wen-Lian Hsu. 2009. Using Contextual Information to Clarify Gene Normalization Ambiguity. In: *The IEEE International Conference on Information Reuse and Integration (IEEE IRI 2009)*, Las Vegas, USA, IEEE Press.

Li, Fangtao, Zhicheng Zheng, Fan Bu, Yang Tang, Xiaoyan Zhu and Minlie Huang. 2009. THU QUANTA at TAC 2009 KBP and RTE Track. In: *Proceedings of Test Analysis Conference 2009 (TAC 09)*, Gaithersburg, Maryland USA.

Maglott, Donna, Jim Ostell, Kim D. Pruitt and Tatiana Tatusova. 2006. Entrez Gene: gene-centered information at NCBI. *Nucleic Acids Research*, 35(suppl 1): D26-D31.

McNamee, Paul and Hoa Trang Dang. 2009. Overview of the TAC 2009 Knowledge Base Population Track. In: *Proceedings of the Second Text Analysis Conference (TAC 2009)*, Gaithersburg, Maryland.

Meza-Ruiz, Ivan and Sebastian Riedel. 2009. Jointly identifying predicates, arguments and senses using Markov logic. In: *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Boulder, Colorado, Association for Computational Linguistics.

Morgan, AA, B Wellner, JB Colombe, R Arens, ME Colosimo and L Hirschman. 2007. Evaluating the automatic mapping of human gene and protein mentions to unique identifiers. In: *Pac Symp Biocomput.*

Morgan, Alexander, Zhiyong Lu, Xinglong Wang, Aaron Cohen, Juliane Fluck, Patrick Ruch, Anna Divoli, Katrin Fundel, Robert Leaman, Jorg Hakenberg, Chengjie Sun, Heng-hui Liu, Rafael Torres, Michael Krauthammer, William Lau, Hongfang Liu, Chun-Nan Hsu, Martijn Schuemie, K Bretonnel Cohen and Lynette Hirschman. 2008. Overview of BioCreative II gene normalization. *Genome Biology*, 9(Suppl 2): S3.

Poon, H and P Domingos. 2007. Joint inference in information extraction. In, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

Richardson, Matthew and Pedro Domingos. 2006. Markov logic networks. *Machine Learning*, 62(Special Issue: Multi-Relational Data Mining and Statistical Relational Learning): 107-136.

Riedel, Sebastian. 2008. Improving the accuracy and efficiency of map inference for markov logic. In: *Proceedings of UAI 2008*, AUAI Press.

Sarawagi, Sunita and Anuradha Bhamidipaty. 2002. Interactive deduplication using active learning. In: *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, Edmonton, Alberta, Canada, ACM.

Singla, Parag and Pedro Domingos. 2006. Entity Resolution with Markov Logic. In: *Proceedings of the Sixth International Conference on Data Mining*, IEEE Computer Society.

Subramaniam, L. Venkata, Sougata Mukherjea, Pankaj Kankar, Biplav Srivastava, Vishal S. Batra, Pasumarti V. Kamesam and Ravi Kothari. 2003. Information extraction from biomedical literature: methodology, evaluation and an application. In: *Proceedings of the twelfth international conference on Information and knowledge management*, New Orleans, LA, USA, ACM.

Tsai, Richard Tzong-Han, Shih-Hung Wu, Wen-Chi Chou, Chun Lin, Ding He, Jieh Hsiang, Ting-Yi Sung and Wen-Lian Hsu. 2006. Various criteria in the evaluation of biomedical named entity recognition. *BMC Bioinformatics*, 7(92): 14.

Zhang, Wei, Jian Su, Chew Lim Tan and Wen Ting Wang. 2010. Entity Linking Leveraging Automatically Generated Annotation. In: *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, Beijing.