Collective Bio-Entity Recognition in Scientific Documents using Hinge-Loss Markov Random Fields

Alexander Miller
UC Santa Cruz
California
almumill@ucsc.edu

Naum Markenzon
UC Santa Cruz
California
nmarkenz@ucsc.edu

Varun Embar
UC Santa Cruz
California
vembar@ucsc.edu

Lise Getoor
getoor@soe.ucsc.edu
UC Santa Cruz
California

ABSTRACT

Identifying biological entities such as genes and proteins from scientific documents is crucial for further downstream tasks such as question answering and information retrieval. This task is challenging as the same surface text can refer to a gene or a protein based on the context. Traditional approaches such as Huang et al. [12] consider the words present in the surrounding text to infer the context. However, they fail to consider the semantics of these words which are better represented by contextual word embeddings such as BERT [6]. Deep learning based approaches, on the other hand, fail to make use of the relational structure of scientific documents. In this work-in-progress paper, we propose our graph-based approach that represents the various sources of relational and semantic information as a graph. We introduce a novel probabilistic approach that jointly classifies all entity references using a class of undirected graphical models called hinge-loss Markov random fields [1]. We show that our proposed approach can combine relational information with embedding-based word semantics. Further, our approach can be easily extended to incorporate new sources of information. Our initial evaluation on the JNLPBA shared task corpus [4] shows that our joint classification approach outperforms both tradition machine learning approaches and semantic models based on word embeddings by up to 7.5% on F1 score.

ACM Reference Format:
Alexander Miller, Naum Markenzon, Varun Embar, and Lise Getoor. 2018. Collective Bio-Entity Recognition in Scientific Documents using Hinge-Loss Markov Random Fields. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Bio-entity recognition systems that identify biological entities in unstructured scientific literature are crucial for tasks such as information extraction, question answering and summarization [4]. Genes and proteins are two important classes of entities recognised by these systems. Identifying genes and proteins in scientific literature is challenging as both entities can have the same surface text. For example, consider the following two sentences:

> *By UV cross-linking and immunoprecipitation, we show that SBP2 specifically binds selenoprotein mRNAs both in vitro and in vivo.*
>
> *The SBP2 clone used in this study generates a 3173 nt transcript.*

The surface text *SBP2* in the first sentence corresponds to a protein whereas the second sentence corresponds to a gene.

Several approaches have been proposed for the task of classifying entity references as gene and protein [3, 7, 8, 11, 12, 23]. Traditional approaches such as Chen and Al-Mubaid [3] and Huang et al. [12] construct feature vectors using the words present in the context window around the surface text and use models such as support vector machines to classify these references. These approaches typically use a bag-of-words-based model to represent the context window and can capture the similarity and semantics of the words in the context. More recently, several BERT-based contextual embeddings trained on a large corpus of biological text have been proposed to address this challenge [9, 14]. However, embedding-based approaches such as Giorgi and Bader [8] identify entities across sentences independently and do not make use of relational structure present in the data. For example, two references that have the same surface text and are present in the same abstract are both likely to be genes or proteins. Another drawback of the current approaches is that they are not easily extensible to incorporate new sources of information. There is a need for a robust, extensible framework that can jointly reason over all the references and can incorporate semantic information present in the word embeddings.

In this work, we propose a novel approach that leverages the flexibility of probabilistic programming to combine relational information with word semantics present in the contextual embeddings. We represent each reference as a node in the graph. Along with various sourcse of relational information such as references being present in the same abstract, we also include semantic information in form of embedding similarities as edges. We then make use of

Woodstock '18, June 03–05, 2018, Woodstock, NY
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

hinge-loss Markov random field (HL-MRF) [1], a class of undirected graphical models, to jointly reason over all references in the graph.

The contributions of our approach include:

- We propose a novel probabilistic approach that combines relational data with word semantics for the task of gene/protein classification.
- Our approach uses a probabilistic programming language and can be extended to include other sources of information.
- Initial experimental evaluation on the 2004 JNLPBA Shared Task corpus [4] shows that our approach is able to outperform traditional word-based models and semantic embedding-based models by up to 7.5% on F1 score.

2 RELATED WORK

We first give a overview of the various approaches proposed in literature for the task of gene/protein classification. We then describe work on word sense disambiguation, a closely related task.

Collective entity disambiguation methods which use graphical models have been proposed [5, 24], but feature engineering for traditional classifiers is the primary focus of research on gene and protein name disambiguation [3, 12]. Chen and Al-Mubaid [3] utilize the information gain of words in their training corpus to construct features. In addition to information gain, Huang et al. [12] looked at the co-occurence relationships of words as part of their feature engineering. While both of these approaches ultimately utilize SVMs, more recent approaches like Yoon et al. [23] utilize deep learning methods and embedding representations of words for more general bio-entity recognition.

Word-sense disambiguation (WSD) is the task of identifying the right meaning of words based on the context. Knowledge, often in a human-curated form such as a thesaurus or ontology, is a key part of many WSD approaches. A survey on various approaches for the task of WSD can be found in Navigli [17]. More recent, approaches that use embeddings [13] and deep learning methods [20] have been proposed. In our work, we propose a novel approach for the task of gene/protein WSD that leverages both relational information and embeddings to jointly reason over all references.

3 APPROACH

Our proposed approach combines relational information with semantics by representing them as a graph. Fig. 1 shows an example graph constructed for three references. The references are represented as nodes. Gene references are represented in blue and protein references are represented in orange. Unlabeled references are shown in white. There are two labeled references and one unlabeled reference. Relational information such as the fact that two references occur in the same abstract or happen to share a bigram in their *context windows* are represented as edges. Semantic information such as cosine similarity computed on the contextual embeddings of the references are also represented as edges.

Having constructed the graph, we combine these information sources and reason jointly over all the references using a hinge-loss Markov random field (HL-MRF). HL-MRFs are a class of conditional probabilistic models over continuous random variables that support modelling of richly structured relational data. HL-MRFs use hinge-loss feature functions and admit tractable and efficient inference. The continuous random variables in HL-MRFs allows us to incorporate similarity measures and confidences of other sources.

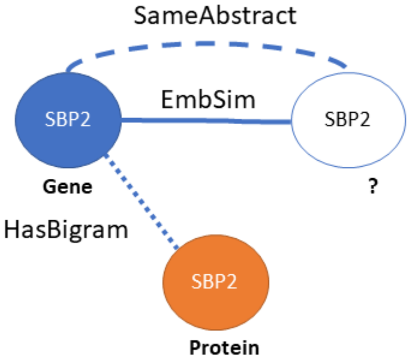

Figure 1: Example graph containing three references represented as nodes. Relational and semantic information are represented as edges. Gene references are represented in blue and protein references are represented in orange. Unlabeled references are shown in white.

We use a probabilistic programming language called probabilistic soft logic (PSL) to generate a HL-MRF from the given graph. We first provide a brief overview of PSL and then describe our proposed approach. For more details on PSL and HL-MRFs, we refer the the reader to Bach et al. [1].

3.1 Probabilistic Soft Logic

PSL is a probabilistic programming language used to define a HL-MRF. PSL supports modeling of rich relational data using weighted logical clauses that encode statistical dependencies and structural constraints. For example, consider the rule:

$$w : \textsc{ExplicitGene}(\mathtt{A}) \rightarrow \textsc{IsGene}(\mathtt{A})$$

Here the predicate ExplicitGene is set to 1 if the surface text contains the term"gene". This PSL rule states that A is likely to be a gene if the surface text contain the word *gene* (e.g *"IL-2 gene"*). The logical atoms in PSL are represented using continuous random variables in the interval $[0, 1]$, and the rule satisfaction is computed using the Lukasiewicz relaxation of Boolean logic. Each relation type in the constructed graph has a corresponding predicate that is used to specify rules.

Given a graph containing references and and a set of rules, PSL generates a HL-MRF by instantiating each rule in the model with the references in the graph. This process is known as *grounding*. The logical atoms in the *ground* rules correspond to the random variables in the HL-MRF, and the ground rules correspond to hinge-loss feature functions. Given a set of random variables, some of which are observed **X**, a PSL model defines a probability distribution over the unobserved variables Y given by:

$$p(\mathrm{Y}|\mathbf{X}) = \frac{1}{Z} e^{-\sum_{j=1}^{m} w_j \phi_j(\mathrm{Y},\mathbf{X})}$$

where Z is the normalization constant, ϕ_j is a *hinge-loss potential* and w_j is a positive weight associated with ϕ_j. The hinge-loss potentials are defined as follows:

$$\phi_j(\mathbf{Y}, \mathbf{X}) = max\{l_j(\mathbf{Y}, \mathbf{X}), 0\}^p$$

where $p \in \{1, 2\}$, l_j is the Lukasiewicz relaxation of the Boolean clause. Getting a maximum a posteriori (MAP) estimate for $\mathbf{Y}$ is a convex optimization task which PSL can perform efficiently.

3.2 PSL Model

We now describe the various rules that incorporate various information sources for the task of gene/protein classification.

Contextual Semantic Similarity: BERT-based contextual word embeddings capture the semantics of the words and are a crucial component of many state-of-the-art NLP approaches [15, 23]. Unlike traditional word embeddings such as Glove[18] and fasttext[16] that generate the same embedding for a given word, contextual embeddings such as BERT[6] and ELMO[19] generate different embeddings for the same word based on the context. This allows us to handle references such as *SBP2*, where based on the context they could either refer to a gene or a protein. We use BioMed-RoBERTa [10], a model based on the RoBERTa architecture [15], which has been pretrained on 2.68 million scientific papers. Since these embeddings contain contextual information, a high level of similarity between the embeddings of two references indicates they are likely to be both genes or proteins. We introduce the predicate EmbeddingSimilarity(A, B) that captures the cosine similarity of reference A and reference B's embeddings. Because all references share significant semantic features, such as the fact that they are all nouns, they have a high degree of baseline similarity. To address this, we set EmbeddingSimilarity(A, B) to be equal to 0 if it is below a fixed threshold θ. In our experiments we set θ to 0.98.

We then propagate the node labels based on the contextual semantic similarity using the the following rules:

$$\textsc{EmbeddingSimilarity}(\mathsf{A},\mathsf{B}) \land \textsc{IsGene}(\mathsf{A}) \rightarrow \textsc{IsGene}(\mathsf{B})$$
$$\textsc{EmbeddingSimilarity}(\mathsf{A},\mathsf{B}) \land \neg\textsc{IsGene}(\mathsf{A}) \rightarrow \neg\textsc{IsGene}(\mathsf{B})$$

Computing similarity for all possible pairs of references is intractable. We use a strategy called *blocking* to identify potential pairs that are likely to have high similarity. In order to reduce the number of pairs for which we do a similarity computation, we only consider pairs that share at least two words in their *context*. This cheap blocking metric dramatically reduces computation time while still yielding useful similarity values. We define the *context* of a reference to be the 10 words which come before and after it.

Contextual Word Similarity: Two references that have identical surface texts and also have several *discriminative* words or bigrams in common are likely to belong to the same class. We define a predicate HasWord(A, X) that is set to 1 if the word X is present in the context of A. To make sure we only look at discriminative words, we look at words that have high information gain (IG). IG is defined as follows:

$$\begin{aligned} IG(w) = &-\sum_{i=1}^{m} P(c_i) log P(c_i) \\ &+P(w)\sum_{i=1}^{m} P(c_i \mid w) log P(c_i \mid w) \\ &+P(\overline{w})\sum_{i=1}^{m} P(c_i \mid \overline{w}) log P(c_i \mid \overline{w}) \end{aligned}$$

where w refers to a given word, m refers to the total number of classes (2 in our case, for gene/protein), and c_i refers to the specific class. We calculate the IG values for all words that occur frequently and chose the top-k words that have the highest IG. We define the predicate HighIG(X) to be equal to 1 if a word X is frequent and occurs in this list of top-k words. We use a frequency threshold of 300 and set $k = 200$ in our experiments.

We introduce a predicate SameName(A, B) which is set to 1 when A and B have the same surface text. We include the following rules in the model:

$$\textsc{SameName}(\mathsf{A},\mathsf{B}) \land \textsc{HasWord}(\mathsf{A},\mathsf{X}) \land \textsc{HasWord}(\mathsf{B},\mathsf{X})$$
$$\land\textsc{HighIG}(\mathsf{X}) \land \textsc{IsGene}(\mathsf{A}) \rightarrow \textsc{IsGene}(\mathsf{B})$$
$$\textsc{SameName}(\mathsf{A},\mathsf{B}) \land \textsc{HasWord}(\mathsf{A},\mathsf{X}) \land \textsc{HasWord}(\mathsf{B},\mathsf{X})$$
$$\land\textsc{HighIG}(\mathsf{X}) \land \neg\textsc{IsGene}(\mathsf{A}) \rightarrow \neg\textsc{IsGene}(\mathsf{B})$$

We also include rules that propagate node labels between references that have the same surface text and have one or more common bigrams in the context. We introduce the predicate HasBigram and include the following rules in the model:

$$\textsc{SameName}(\mathsf{A},\mathsf{B}) \land \textsc{HasBigram}(\mathsf{A},\mathsf{X}) \land \textsc{HasBigram}(\mathsf{B},\mathsf{X})$$
$$\land\textsc{IsGene}(\mathsf{A}) \rightarrow \textsc{IsGene}(\mathsf{B})$$
$$\textsc{SameName}(\mathsf{A},\mathsf{B}) \land \textsc{HasBigram}(\mathsf{A},\mathsf{X}) \land \textsc{HasBigram}(\mathsf{B},\mathsf{X})$$
$$\neg\textsc{IsGene}(\mathsf{A}) \rightarrow \neg\textsc{IsGene}(\mathsf{B})$$

We experimented with feature selection approaches for bigrams, but found that bigrams are unique enough textual features for this to be unnecessary.

Continuity Within Abstracts: References in the same abstract that have the same surface text are likely to belong to the same class. To capture this, we included two rules of the form:

$$\textsc{SameAbstract}(\mathsf{A},\mathsf{B}) \land \textsc{SameName}(\mathsf{A},\mathsf{B})$$
$$\land\textsc{IsGene}(\mathsf{A}) \rightarrow \textsc{IsGene}(\mathsf{B})$$
$$\textsc{SameAbstract}(\mathsf{A},\mathsf{B}) \land \textsc{SameName}(\mathsf{A},\mathsf{B})$$
$$\land\neg\textsc{IsGene}(\mathsf{A}) \rightarrow \neg\textsc{IsGene}(\mathsf{B})$$

Frequent Co-occurrence: We define two references to be *adjacent* if they occur in the same abstract and no other reference occurs between them. We introduce the predicate Adjacent(A, B) to be equal to 1 if references A and B are adjacent. If two pairs of adjacent references are correspondingly identical in their surface text, and three out of four of these are gene references, then the fourth is likely also a gene reference. Rules which incorporate this idea follow the form of:

$$\textsc{Adjacent}(\mathtt{A},\mathtt{B}) \land \textsc{Adjacent}(\mathtt{C},\mathtt{D})$$
$$\land \textsc{SameName}(\mathtt{A},\mathtt{C}) \land \textsc{SameName}(\mathtt{B},\mathtt{D})$$
$$\land \textsc{IsGene}(\mathtt{A}) \land \textsc{IsGene}(\mathtt{B})$$
$$\land \textsc{IsGene}(\mathtt{C}) \rightarrow \textsc{IsGene}(\mathtt{D})$$

Surface Text Observations (SO): The presence of discriminative terms in a reference's surface text are a strong signal that it belongs to a particular class. For example, if a reference's surface text contains the word "gene", as in "*IL-2 gene*", it is quite likely that it refers to a gene. For the words "gene" and "protein" which are strong indicators of the class label, we define the predicates ExplicitGene(A) and ExplicitProtein(A) that are equal to 1 if the surface text of the reference A contains the terms "gene" and "protein" respectively. We then include the following rules:

$$\textsc{ExplicitGene}(\mathtt{A}) \rightarrow \textsc{IsGene}(\mathtt{A})$$
$$\textsc{ExplicitProtein}(\mathtt{A}) \rightarrow \neg\textsc{IsGene}(\mathtt{A})$$

We incorporate similar rules about words which are less explicit by first computing the mutual information (MI) of the word with respect to both the gene and protein class. MI is defined as:

$$MI_i(w) = \frac{N \times a}{(a+b) \times (a+c)}$$

Where N is the number of documents, a is the number of times w occurs in a document of class i, b is the number of times w occurs in a document which doesn't belong to class i, and c is the number of documents in class i which do not contain w. We then define the predicate StrongGene(X) to be equal to 1 if a word X occurs frequently and has a gene class MI value above a specified threshold. StrongProtein is defined similarly. We define the predicate SurfaceTextHasWord(A, X) to be equal to 1 when the surface text of reference A contains the word X.

Mutual information is strongly influenced by the marginal probability of the class for which it is calculated [22]. As our dataset is imbalanced, we use different MI thresholds for each class. We set the gene-class MI threshold to 1.5 and the protein-class MI threshold to 1. We use a frequency threshold of 200 for both classes.

We include the following rules:

$$\textsc{StrongGene}(\mathtt{X}) \land \textsc{SurfaceTextHasWord}(\mathtt{A},\mathtt{X}) \rightarrow \textsc{IsGene}(\mathtt{A})$$
$$\textsc{StrongProtein}(\mathtt{X}) \land \textsc{SurfaceTextHasWord}(\mathtt{A},\mathtt{X}) \rightarrow \neg\textsc{IsGene}(\mathtt{A})$$

Other Classifiers (OC): One of PSL's unique strengths is its ability to incorporate information from multiple sources. We incorporate predictions from four other classifiers, all of which use either a SVM or logistic regression model, which we use as baselines in our experimental evaluation. The details of these classifiers will be discussed in Section 4. As an example, we define the predicate LR(A) to be equal to 1 when a logistic regression-based classifier predicts reference A to be a gene and 0 otherwise. We include rules of the form:

$$\mathrm{LR}(\mathtt{A}) \rightarrow \textsc{IsGene}(\mathtt{A})$$
$$\neg\mathrm{LR}(\mathtt{A}) \rightarrow \neg\textsc{IsGene}(\mathtt{A})$$

We add four rules of this form that correspond to the outputs of SVMs and logistic regression trained on the bag-of-words representation and the embeddings of the surface text.

4 EXPERIMENTAL EVALUATION

4.1 Experimental Results

We evaluate the effectiveness of our approach on the JNLPBA 2004 shared task [4]. The corpus contains 2,000 abstracts containing bioentity references that are annotated as genes and proteins. There are 30,269 protein and 9,533 gene references.

We evaluate our approach in two different settings which we will refer to as the *completely-unlabeled* (**CU**) setting and the *partially-labeled* (**PL**) setting. In **CU**, we generate equally-sized folds by randomly assigning each abstract to a fold. In the **CU** setting, all references in an abstract are either unlabeled or completely labeled. In **PL**, we randomly assign each reference to one of the folds. In this setting, the abstracts are partially labeled. Some references within an abstract are labeled while others are unlabeled. For both **CU** and **PL** we create 10 folds and evaluate our model by performing 10-fold CV. We performed paired t-tests and all results in bold are statistically significant at the $p = 0.05$ level.

4.2 Baselines

We compare the performance of our approach against two embedding-based semantic models and two context-based models. For the embedding-based semantic models, we generate the embeddings of the references using BioMed-RoBERTa and train a logistic regression and support vector machine using the embedding dimensions as features. We will refer to these models as the *Embeddings SVM/LR* models or *embeddings-based SVM/LR* models. Similarly, for the context-based models we train a support vector machine and logistic regression using the bag-of-words representation consisting of unigrams and bigrams counts present in each reference's context. We will refer to these models as the *BoW SVM/LR* models or the *bag-of-words-based* models. For each reference we define the context as five words preceding it, the reference's surface text, and the five words after it. Both these approaches classify each reference independently and fails to make use of relational data.

For our PSL-based approach, we use all the rules mentioned in Section 3. We use our validation set to learn weights using an approach based on Bayesian optimization [21] and use these in our experiments. Since PSL outputs truth values in the range $[0, 1]$ for each reference, we binarize the labels by considering all references with ISGENE value ≥ 0.5 to be genes.

The precision, recall, F1 score and accuracy for the completely-unlabeled setting is shown in Table 1. Among the baselines we observe that embedding-based approaches perform better than bag-of-words based models. This is due the semantic information contained in the word embeddings. We also see that SVMs perform better than logistic regression.

We observe that PSL outperforms all baselines across all metrics. PSL's improvement can be attributed to the fact that it combines semantic information with relational information and jointly infers the class labels for all references. In **CU** PSL outperforms the strongest baseline, embedding-based SVMs, by 3.8% on F1 score.

Model	F1	Accuracy	Precision	Recall
PSL	**0.895** ± 0.014	**0.951** ± 0.007	**0.922** ± 0.020	**0.870** ± 0.016
Embeddings SVM	0.862 ± 0.017	0.937 ± 0.008	0.894 ± 0.016	0.833 ± 0.025
Embeddings LR	0.856 ± 0.018	0.933 ± 0.008	0.886 ± 0.020	0.829 ± 0.026
BoW SVM	0.811 ± 0.015	0.914 ± 0.009	0.854 ± 0.020	0.773 ± 0.018
BoW LR	0.718 ± 0.021	0.884 ± 0.011	0.857 ± 0.019	0.618 ± 0.031

Table 1: Completely-Unlabeled Abstracts: F1, Accuracy, Precision, Recall, Std. Deviations

Model	F1	Accuracy	Precision	Recall
PSL	**0.937** ± 0.007	**0.970** ± 0.003	**0.941** ± 0.006	**0.934** ± 0.011
Embeddings SVM	0.872 ± 0.008	0.941 ± 0.003	0.902 ± 0.011	0.844 ± 0.012
Embeddings LR	0.863 ± 0.011	0.937 ± 0.005	0.892 ± 0.010	0.836 ± 0.014
BoW SVM	0.820 ± 0.011	0.916 ± 0.005	0.845 ± 0.013	0.797 ± 0.015
BoW LR	0.731 ± 0.012	0.888 ± 0.005	0.863 ± 0.011	0.634 ± 0.015

Table 2: Partially-Labeled Abstracts: F1, Accuracy, Precision, Recall, Std. Deviations

Moreover, PSL tends to have lower standard deviations when compared to other baselines.

The precision, recall, F1 score and accuracy in **PL** is shown in Table 2. Similar to the previous setting, we observe that PSL outperforms all other baselines. PSL has an improvement of 7.5% on F1 score over the embeddings-based SVM. This increase comes from achieving a 4.3% increase in average precision and a 10.6% increase in average recall. The average accuracy also increases by 3% over the embeddings-based SVM.

From the two tables we observe that all models perform better in the partially-labeled setting when compared to completely-unlabeled setting. For PSL, this improvement is due to the propagation of node labels between references in the same abstract. We hypothesize the improvement in the baselines is due to the increased overlap of words and bigrams in the references' context windows as they are part of the same abstract.

5 FUTURE WORK

In this work-in-progress paper, we proposed an probabilistic approach for the task of bio-entity recognition that combines semantic information using contextual embeddings with relational information. Further, our approach uses probabilistic programming and can easily incorporate other sources of information. Our initial experiments shows that the proposed approach is able to outperform purely semantic-based approaches as well as traditional bag-of-words approach.

In the current work, we compared our approach to existing state-of-the-art machine learning techniques trained on contextual word embeddings. In our future work, we will compare our approach to full deep learning based models such as Yoon et al. [23], that both extract and label bio-entities. We also intend to incorporate other sources of domain knowledge to help disambiguate references. A promising source of such knowledge is the UMLS Metathesaurus [2], a database that contains the relationships between and hierarchies of terms and words as they appear in a biomedical context. Further, we plan to extend our approach to both extract and classify bio-entities from unstructured text.

REFERENCES

[1] Stephen Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. 2017. Hinge-Loss Markov Random Fields and Probabilistic Soft Logic. *JMLR* (2017).

[2] O. Bodenreider. 2004. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Res* 32 (2004), D267–270.

[3] Ping Chen and Hisham Al-Mubaid. 2006. Context-based Term Disambiguation in Biomedical Literature. *FLAIRS* (2006).

[4] Nigel Collier and Jin-Dong Kim. 2004. Introduction to the Bio-entity Recognition Task at JNLPBA. In *BioNLP*. COLING, Geneva, Switzerland, 73–78.

[5] Hongjie Dai, Richard Tzong, Richard Tzong-Han Tsai, and Wen-Lian Hsu. 2012. Entity Disambiguation Using a Markov-Logic Network. (2012).

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805* (2018).

[7] Filip Ginter, Jorma Boberg, Jouni Järvinen, and Tapio Salakoski. 2004. New techniques for disambiguation in natural language and their application to biological text. *JMLR* 5 (2004), 605–621.

[8] John M Giorgi and Gary D Bader. 2019. Towards reliable named entity recognition in the biomedical domain. *Bioinformatics* 36, 1 (2019), 280–286.

[9] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. In *ACL*.

[10] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. In *ACL*.

[11] Vasileios Hatzivassiloglou, Pablo A Duboue, and Andrey Rzhetsky. 2001. Disambiguating proteins, genes, and RNA in text: a machine learning approach. *Bioinformatics* 17 (2001), S97–S106.

[12] Changqin Huang, Jia Zhu, Xiaodi Huang, Min Yang, Gabriel Fung, and Qintai Hu. 2018. A Novel Approach for Entity Resolution in Scientific Documents Using Context Graphs. *Information Sciences* (2018).

[13] Ignacio Iacobacci, Mohammad Taher Pilehvar, and Roberto Navigli. 2016. Embeddings for Word Sense Disambiguation: An Evaluation Study. In *ACL*.

[14] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* 36, 4 (2020), 1234–1240.

[15] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR* abs/1907.11692 (2019).

[16] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in Pre-Training Distributed Word Representations. In *LREC*.

[17] Roberto Navigli. 2009. Word sense disambiguation. *Comput. Surveys* 41, 2 (2009), 1–69.

[18] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

[19] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL*.

[20] A. K. M. Sabbir, Antonio Jimeno-Yepes, and Ramakanth Kavuluru. 2016. Knowledge-Based Biomedical Word Sense Disambiguation with Neural Concept Embeddings and Distant Supervision. *CoRR* abs/1610.08557 (2016).

[21] Sriram Srinivasan, Golnoosh Farnadi, and Lise Getoor. 2020. BOWL: Bayesian Optimization for Weight Learning in Probabilistic Soft Logic. In *AAAI Conference on Artificial Intelligence (AAAI)*.

[22] Yiming Yang and Jan O. Pedersen. 1997. A Comparative Study on Feature Selection in Text Categorization. In *ICML*.

[23] Wonjin Yoon, Chan Ho So, Jinhyuk Lee, and Jaewoo Kang. 2019. CollaboNet: collaboration of deep neural networks for biomedical named entity recognition. *Bioinformatics* 20, S10 (2019).

[24] Weixin Zeng, Jiuyang Tang, Xiang Zhao, Bin Ge, and Weidong Xiao. 2018. Named Entity Disambiguation via Probabilistic Graphical Model with Embedding Features. In *NeuRIPS*.