Pattern-enhanced Named Entity Recognition with Distant Supervision

Xuan Wang[1], Yingjun Guan[1], Yu Zhang[1], Qi Li[2], Jiawei Han[1]
[1]Department of Computer Science, University of Illinois at Urbana-Champaign, IL, USA
[2]Department of Computer Science, Iowa State University, IA, USA
[1]{xwang174, yingjun2, yuz9, hanj}@illinois.edu, [2]qli@iastate.edu,

2020 IEEE International Conference on Big Data (Big Data) | 978-1-7281-6251-5/20/$31.00 ©2020 IEEE | DOI: 10.1109/BigData50022.2020.9378052

***Abstract*—Supervised deep learning methods have achieved state-of-the-art performance on the task of named entity recognition (NER). However, such methods suffer from high cost and low efficiency in training data annotation, leading to highly specialized NER models that cannot be easily adapted to new domains. Recently, distant supervision has been applied to replace human annotation, thanks to the fast development of domain-specific knowledge bases. However, the generated noisy labels pose significant challenges in learning effective neural models with distant supervision. We propose PATNER, a distantly supervised NER model that effectively deals with noisy distant supervision from domain-specific dictionaries. PATNER does not require human-annotated training data but only relies on unlabeled data and incomplete domain-specific dictionaries for distant supervision. It incorporates the distant labeling uncertainty into the neural model training to enhance distant supervision. We go beyond the traditional sequence labeling framework and propose a more effective fuzzy neural model using the tie-or-break tagging scheme for the NER task. Extensive experiments on three benchmark datasets in two domains demonstrate the power of PATNER. Case studies on two additional real-world datasets demonstrate that PATNER improves the distant NER performance in both entity boundary detection and entity type recognition. The results show a great promise in supporting high quality named entity recognition with domain-specific dictionaries on a wide variety of entity types.**

***Index Terms*—named entity recognition, distant supervision, pattern mining, neural network**

I. INTRODUCTION

Named Entity Recognition (NER) is a fundamental task in text mining that aims to recognize pre-defined types of entities from text. NER is important for various applications, such as information retrieval [35], knowledge base construction [27] and relation extraction [24]. Recent deep learning methods lead to state-of-the-art NER systems [2], [3], [8], [9], [11], [13], [31]. However, such methods suffer from the high cost and low efficiency in manual annotation of training data, especially for the domains like biology and medicine where expert annotation is expensive. Human annotation also leads to highly specialized NER models that cannot be easily adapted to new entity types.

Recently, distant supervision has been applied to replace human annotation, thanks to the fast development of domain-specific knowledge bases. For example, there are Medical Subject Heading (MeSH) and Unified Medical Language System (UMLS) databases for the biomedical domain, and YAGO and WiKiData for the general domain. These knowledge bases provide comprehensive dictionaries that makes it possible to annotate named entities in different domains in a large scale automatically. A straightforward approach is to conduct a dictionary matching: if a token is found in the dictionary, it will be labeled as an entity of the corresponding type.

Dictionary-based distant supervision, however, encounters several issues comparing with human supervision. A major problem lies in the quality of the existing dictionaries. Most dictionaries contain accurate entity names, but have limited coverage, such as missing abbreviations, aliases names, and colloquial names. Thus a simple dictionary matching may lead to a low recall for the NER task. To overcome this issue, some methods use the dictionary matching results as distant supervision and then train neural models to improve the recall [5], [6], [18], [26], [30]. However, this approach suffers from high false negative rate of the training labels due to the limited coverage of the dictionaries. A recent work, AUTONER [26], addresses the false negative labeling problem by introducing "unknown" type when using distant supervision from entity dictionaries. This method improves the performance comparing with existing distantly supervised NER methods. However, it still utilize limited information from the incomplete dictionaries and have a significant performance gap compared with supervised NER methods.

We propose PATNER, a distantly supervised NER model that effectively deals with noisy distant supervision from domain-specific dictionaries. PATNER does not require human-annotated training data but only relies on unlabeled data and incomplete domain-specific dictionaries for distant supervision. PATNER automatically mines the entity naming principles (e.g., disease entities often contain the words "syndrome" or "disorder") from the dictionaries to enhance distant supervision. The entity naming principles are used to automatically expand the input dictionaries by treating the unlabeled word sub-sequences as a mixture of dictionary entities and some random phrases. Based on the expanded dictionary, the distant labels are generated as probability distributions over all the entity types with a Gaussian mixture model (GMM) [28]. This distant labeling uncertainty is then incorporated into the neural model training. We go beyond the traditional sequence labeling framework and propose a more effective fuzzy neural model using the tie-or-break tagging scheme for the NER task.

We demonstrate the power of PATNER in intensive exper-

Authorized licensed use limited to: University of Illinois. Downloaded on May 05,2021 at 20:55:00 UTC from IEEE Xplore. Restrictions apply.

iments on real-world datasets in both biomedical and general domains. PATNER outperforms existing distantly supervised NER methods by a large margin. PATNER even outperforms state-of-the-art supervised NER methods in the biomedical domain. Case studies on two additional real-world datasets demonstrate that PATNER improves the distant NER performance in both entity boundary detection and entity type recognition. These results show a great promise in supporting high quality named entity recognition with user-specified entity dictionaries on a wide variety of entity types.

We summarize our major contributions as follows.

- A distantly supervised NER model, PATNER, is proposed that only relies on unlabeled data and incomplete domain-specific dictionaries for distant supervision
- A distant label generation method is proposed that extracts the entity naming principles from dictionaries to enhance the distant supervision.
- A fuzzy NER neural model is proposed to incorporate the distant labeling uncertainty into the neural model training to improve the NER performance.
- Experiments on three benchmark datasets demonstrate the power of PATNER in the biomedical and technical review domains. Case studies on two additional real-world datasets demonstrate that PATNER improves the distant NER performance in both entity boundary detection and entity type recognition.

II. THE PATNER FRAMEWORK

In this section, we introduce the overall framework of PATNER. It mainly includes two parts: distant label generation and PATNER neural model.

A. Dictionary Preparation

PATNER takes a raw corpus and a domain-specific dictionary as the input. The dictionary can be collected from public databases. The corpus-aware dictionary is firstly tailored in a similar way as proposed by Shang *et al.* [26]. We remove the entities in the dictionary whose canonical name has never appeared in the raw corpus to reduce the false-positive labels by alias matching. Because some alias names are short abbreviations of the canonical names that can cause false positive labeling during dictionary matching. Then we extract a list of high-quality phrases as the **candidate entities** for dictionary expansion. We utilize the state-of-the-art distantly supervised phrase mining method, AUTOPHRASE [25], with the corpus and dictionary given as the input. AUTOPHRASE generates the candidate entities according to four criteria: popularity, concordance, informativeness and completeness. Some low-frequency entities may not be included as candidate entities by AUTOPHRASE at this step, but may be recognized by the trained neural model during inference.

B. Distant Label Generation

In order to improve the recall of the incomplete dictionary, we first expand the dictionary with the candidate phrases generated from the previous step. The expanded dictionary is used to generate distant labels for the neural model training.

TABLE I
EXAMPLES OF DICTIONARY PREPROCESSING.

Before pre-processing	After Preprocessing
visual field defects	_visual_field_defects
(R)-alpha-methylhistamine	_LETTER_methylhistamine_
1,25-dihydroxyvitamin	_DIGIT_dihydroxyvitamin

Pattern Extraction. Dictionary expansion is based on the frequent patterns automatically extracted from the input dictionary. Some dictionary preprocessing are first conducted to increase the frequent pattern coverage. All the entities in the dictionary are normalized to lowercase. The digits like '1', '2', '3', 'I', 'II', and the digit combinations like "1,25-", "1-2-" are normalized to a special token "DIGIT". Similarly, the letter characters like 'a', 'b', 'c', 'x', 'y', 'z', "alpha", "gamma" and the corresponding letter combinations are normalized to a special token "LETTER". All the non-alphanumeric characters are changed to underscores. Some examples of the dictionary preprocessing results are shown in Table I.

Then we conduct frequent continuous sequential pattern miming [17] to extract two types of patterns from the dictionary: character patterns and word patterns. The character patterns are sub-word patterns mined from all the word tokens in the dictionary. The word patterns are the sub-phrase patterns mined from all the n-gram phrases in the dictionary. We further conduct a closed pattern filtering to remove the less-informative short patterns. All the parameters of pattern mining (e.g., the minimum support, the maximum length and the closed pattern threshold) are automatically chosen so that the extracted patterns cover more than 90% of the entities in the dictionary. In our experiments, the minimum supports are set to 6 and 2 for the character and word patterns, respectively. The maximum lengths are set to 10 and 6 for the character and word patterns, respectively. The ratio between each pattern and its sub-patterns is set to 0.9 for closed pattern filtering. The sub-word and sub-phrase patterns can then be used to discover new entity names in the candidate list of high-quality phrases. The character and word patterns are used to discover new entities in the list of candidate phrases. The basic principle is that the candidates following the same patterns extracted from the entities in the dictionary should have a high probability to be an entity of the corresponding type (e.g., in Figure 1, a candidate phrase contains "syndrome" is likely to be a disease entity). Based on this principle, we propose a Gaussian mixture model (GMM)-based expansion method for dictionary expansion.

Gaussian Mixture Model (GMM)-based Expansion. The intuition of GMM-based expansion is to treat the candidate phrases as a mixture of dictionary entities and some random phrases. We first give a score to each entity and candidate phrase with the patterns extracted from the previous step. The score distribution of the dictionary entities can be directly calculated from the scores of dictionary entities. We use the

Authorized licensed use limited to: University of Illinois. Downloaded on May 05,2021 at 20:55:00 UTC from IEEE Xplore. Restrictions apply.

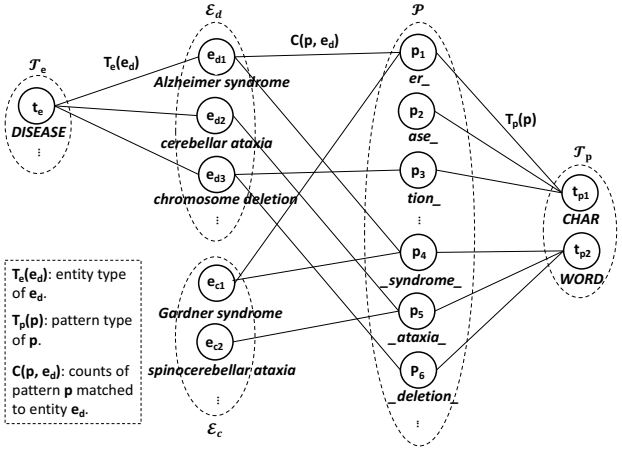

Fig. 1. Bipartite graph of the entities and patterns.

Expectation-Maximization (EM) algorithm to infer the score distribution of the random phrases from the scores of the candidate phrases. Then we have the probability of each candidate phrase being in the dictionary of a corresponding entity type. An advantage of using the GMM model is that we can label the expanded phrases as a probability distribution over all the entity types and incorporate this label distribution in the neural model training to improve the NER performance.

We first calculate a score of each entity and candidate phrase with the patterns extracted from the previous step. Let $\mathcal{P}$ denote the set of patterns extracted by the previous step; $\mathcal{E}_d$ denote the set of entities in the dictionary; $\mathcal{E}_c$ denote the set of phrases in the candidate list; $\mathcal{T}_e$ denote the set of entity types (e.g., Disease); and $\mathcal{T}_p$ denote the set of pattern types (i.e., character and word patterns). The relations between the entities, patterns and types are shown in Figure 1. The scoring consists of two steps: pattern scoring and phrase scoring.

During pattern scoring, each pattern and pattern type is given a score ρ and τ, respectively, based on its frequency in the dictionary. For any $p \in \mathcal{P}$, $t_p \in \mathcal{T}_p$ and $t_e \in \mathcal{T}_e$,

$$\rho(p, t_e) = \frac{\sum_{e_d \in \mathcal{E}_d, T_e(e_d)=t_e} C(p, e_d)}{\sum_{e_d \in \mathcal{E}_d, T_e(e_d)=t_e} \sum_{p' \in \mathcal{P}, T_p(p')=T_p(p)} C(p', e_d)}, \tag{1}$$

$$\tau(t_p, t_e) = \frac{\sum_{e_d \in \mathcal{E}_d, T_e(e_d)=t_e} \sum_{p \in \mathcal{P}, T_p(p)=t_p} C(p, e_d)}{\sum_{e_d \in \mathcal{E}_d, T_e(e_d)=t_e} \sum_{p' \in \mathcal{P}} C(p', e_d)}, \tag{2}$$

where $C(p, e)$ denotes the counts of pattern p matched to entity e; $T_e(e) \in \mathcal{T}_e$ denotes the entity type of entity e; and $T_p(p) \in \mathcal{T}_p$ denotes the pattern type of pattern p (Figure 1).

During phrase scoring, each phrase in the candidate list is given a score ω based on the patterns it matches. For any $e_c \in \mathcal{E}_c$ and $t_e \in \mathcal{T}_e$,

$$\omega(e_c, t_e) = \frac{\sum_{t_p \in \mathcal{T}_p} \sum_{p \in \mathcal{P}} \tau(t_p, t_e) * \rho(p, t_e) * C(p, e_c)}{n(e_c) + k}, \tag{3}$$

where $n(e_c)$ denotes the number of tokens of entity e_c; and k denotes an entity length normalization constant. We choose $k = 2$ in our experiments. Similarly, each entity in the dictionary is also given a score based on the patterns it matches.

Based on the pattern and entity scoring results, we can calculate the dictionary and phrase distributions. The empirical dictionary distribution $\mathcal{N}(\mu_1, \sigma_1^2)$ is directly calculated as the score distribution of the dictionary entities. For any $t_e \in \mathcal{T}_e$,

$$\mu_1 = \frac{\sum_{e \in \mathcal{E}_d, T_e(e)=t_e} \omega(e, t_e)}{|\{e \mid e \in \mathcal{E}_d, T_e(e) = t_e\}|}, \tag{4}$$

$$\sigma_1^2 = \frac{\sum_{e \in \mathcal{E}_d, T_e(e)=t_e} (\omega(e, t_e) - \mu_1)^2}{|\{e \mid e \in \mathcal{E}_d, T_e(e) = t_e\}|}. \tag{5}$$

The random phrase distribution $\mathcal{N}(\mu_2, \sigma_2^2)$ is initialized as a standard normal distribution. Then we use the EM algorithm to infer the random phrase distribution.

During the E-step, we calculate an assignment matrix $\mathbf{A} \in [0, 1]^2$ to get the probability of each phrase being in the dictionary distribution and the random phrase distribution. Let f_1 denote the probability density function of $\mathcal{N}(\mu_1, \sigma_1^2)$ and f_2 the probability density function of $\mathcal{N}(\mu_2, \sigma_2^2)$. For any $e_c \in \mathcal{E}_c$ and $t_e \in \mathcal{T}_e$,

$$A(e_c, f_1) = \frac{f_1(\omega(e_c, t_e))}{f_1(\omega(e_c, t_e)) + f_2(\omega(e_c, t_e))}, \tag{6}$$

$$A(e_c, f_2) = \frac{f_2(\omega(e_c, t_e))}{f_1(\omega(e_c, t_e)) + f_2(\omega(e_c, t_e))}. \tag{7}$$

During the M-step, we re-calculate the parameters of the random phrase distribution. For any $t_e \in \mathcal{T}_e$,

$$\mu_2 = \frac{\sum_{e_c \in \mathcal{E}_c} \omega(e_c, t_e) * A(e_c, f_2)}{|\mathcal{E}_c|}, \tag{8}$$

$$\sigma_2^2 = \frac{\sum_{e_c \in \mathcal{E}_c} (\omega(e_c, t_e) - \mu_2)^2}{|\mathcal{E}_c|}. \tag{9}$$

After each E-M iteration, the KL divergence is calculated between the inferred dictionary distribution $\mathcal{N}'(\mu_1', \sigma_1'^2)$ and the empirical dictionary distribution $\mathcal{N}(\mu_1, \sigma_1^2)$. For any $e_c \in \mathcal{E}_c$ and $t_e \in \mathcal{T}_e$,

$$\mu_1' = \frac{\sum_{e_c \in \mathcal{E}_c} \omega(e_c, t_e) * A(e_c, f_1)}{|\mathcal{E}_c|}, \tag{10}$$

$$\sigma_1'^2 = \frac{\sum_{e_c \in \mathcal{E}_c} (\omega(e_c, t_e) - \mu_1')^2}{|\mathcal{E}_c|}, \tag{11}$$

$$KL(\mathcal{N}, \mathcal{N}') = log(\frac{\sigma_1}{\sigma_1'}) + \frac{\sigma_1^2 + (\mu_1 - \mu_1')^2}{2\sigma_1'^2} - \frac{1}{2}. \tag{12}$$

The E-M algorithm converges when the change of the KL divergence is smaller than a threshold of $1e-5$.

Label Distributions. There are two ways to utilize the probabilities from the E-M algorithm. One way is to simply select all the candidate phrases with $f_1 > 0.5$ and add them into the dictionary. The dictionary so expanded is used by our methods PATNER *(w/o neural model)* and PATNER *(w/o fuzzy label)*. Since the GMM-based expansion is performed for each entity type independently, one phrase could be expanded into

Authorized licensed use limited to: University of Illinois. Downloaded on May 05,2021 at 20:55:00 UTC from IEEE Xplore. Restrictions apply.

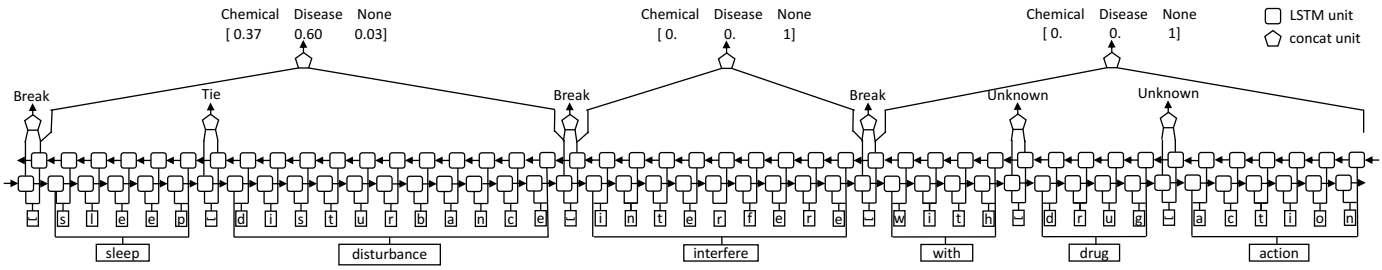

Fig. 2. Illustration of PATNER neural model.

Algorithm 1 Distant Label Generation

Input: $\mathcal{P}$, $\mathcal{E}_d$, $\mathcal{E}_c$, $\mathcal{T}_e$, $\mathcal{T}_p$ and $\epsilon = 1e-5$.
Output: Expanded dictionary $D = \{(e, L_e(e))\}$.

Initialization: $D \leftarrow \{(\mathcal{E}_d, L_e(\mathcal{E}_d))\}$, $\mathcal{N}(\mu_2, \sigma_2^2) \leftarrow \mathcal{N}(0,1)$, $\delta \leftarrow 1$, $kl \leftarrow \infty$.
for $p \in \mathcal{P}$, $t_p \in \mathcal{T}_p$ and $t_e \in \mathcal{T}_e$ **do**
    Calculate the pattern score S_p using Eq. 1.
    Calculate the pattern type score S_{t_p} using Eq. 2.
for $e_d \in \mathcal{E}_d$, $e_c \in \mathcal{E}_c$ and $t_e \in \mathcal{T}_e$ **do**
    Calculate the entity score S_e using Eq. 3.
Calculate $\mathcal{N}(\mu_1, \sigma_1^2)$ using Eq. 4, 5.
while $\delta \geq \epsilon$ **do**
    for $e_c \in \mathcal{E}_c$ and $t_e \in \mathcal{T}_e$ **do**
        Calculate the assignment matrix $\mathbf{A}$ using Eq. 6, 7.
    for $e_c \in \mathcal{E}_c$ and $t_e \in \mathcal{T}_e$ **do**
        Calculate $\mathcal{N}(\mu_2, \sigma_2^2)$ using Eq. 8, 9.
    Calculate $\mathcal{N}(\mu_1', \sigma_1'^2)$ using Eq. 10, 11.
    Calculate the KL-divergence between $\mathcal{N}(\mu_1, \sigma_1^2)$ and $\mathcal{N}(\mu_1', \sigma_1'^2)$ as kl_{new} using Eq. 12.
    $\delta = kl - kl_{new}$, $kl \leftarrow kl_{new}$.
for $e_c \in \mathcal{E}_c$ and $t_e \in \mathcal{T}_e$ **do**
    Calculate $L_e(e_c)$ using Eq. 13, 14, 15.
    $D = D \cup \{(e_c, L_e(e_c))\}$.

the dictionary with multiple entity types. Another way is to label each phrase as a probability distribution over all the entity types. Let L_e denote the label distribution of an entity, $|L_e| = |\mathcal{T}_e| + 1$ that includes all the possible entity types and a "None" type. For any $e_c \in \mathcal{E}_c$ and $t_c \in \mathcal{T}_e$,

$$L_e(e_c, t_e) = \frac{f_1(\omega(e_c, t_e))}{\sum_{t_e' \in \mathcal{T}_e} f_1(\omega(e_c, t_e')) + f_1(None)}, \quad (13)$$

$$L_e(e_c, None) = \frac{f_1(None)}{\sum_{t_e' \in \mathcal{T}_e} f_1(\omega(e_c, t_e')) + f_1(None)}, \quad (14)$$

$$f_1(None) = \prod_{t_e' \in \mathcal{T}_e} (1 - f_1(\omega(e_c, t_e')). \quad (15)$$

This label distribution is used by our final method PATNER. The algorithm flow is shown in Algorithm 1.

C. PATNER *Neural Model*

The expanded dictionary is used to label the raw corpus as distant supervision to train a neural model. Given the input corpus and the expanded dictionary, we first conduct exact string matching to generate distant labels. Conflicted matches are resolved by maximizing the total number of matched tokens on each sentence. Based on the dictionary-matching results, we go beyond the traditional sequence labeling framework and propose a more effective fuzzy neural model using the tie-or-break tagging scheme. We first introduce the basic network model with the "tie-or-break" tagging schema. Then we introduce the fuzzy neural model of PATNER.

"Tie-or-break" Tagging Schema. For example, in Figure 2, "sleep disturbance" is an entity that is expanded into the dictionary; "drug action" is a candidate phrase that is not expanded into the dictionary; and "interfere" and "with" are two tokens that are not matched to any entities or candidate phrases. Thus the connection between "sleep" and "disturbance" is labeled as *Tie*; the connection between "drug" and "action" is labeled as *Unknown*; and the connection between "disturbance" and "interfere" is labeled as *Break*. Tokens between two consecutive *Break*s form a token span. Each token span is associated with a label distribution we get from the previous GMM model. For example, in Figure 2, "sleep disturbance" is expanded into the dictionary with both the "Chemical" and "Disease" types. The label of "sleep disturbance" is a probability distribution $[0.37, 0.60, 0.03]$ across the types "Chemical", "Disease" and "None". An advantage of using the "tie-or-break" tagging schema is that token spans with "unknown" type connections can be skipped during the learning of entity type recognition, which greatly reduces the false negative labeling problem for distant supervision.

Tokens between two consecutive *Break*s form a token span. Each token span is associated with a label distribution we get from the previous GMM model. For example, in Figure 2, "sleep disturbance" is expanded into the dictionary with both the "Chemical" and "Disease" types. The label of "sleep disturbance" is a probability distribution $[0.37, 0.60, 0.03]$ across the types "Chemical", "Disease" and "None". An advantage of using the "tie-or-break" tagging schema is that token spans with "unknown" type connections can be skipped during the learning of entity type recognition, which greatly reduces the false negative labeling problem for distant supervision.

Authorized licensed use limited to: University of Illinois. Downloaded on May 05,2021 at 20:55:00 UTC from IEEE Xplore. Restrictions apply.

TABLE II
BASIC STATISTICS OF THE DATASETS.

Dataset	BC5CDR	NCBI-Disease	LaptopReview	PubMed_3M	Yelp_3M
Raw Sent. #	20,217	7,284	4,521	19,204	39,942
Dictionary	MeSH + CTD	MeSH + CTD	ComputerHope.com	Gene Ontology	Open Food Facts
Entity Types	Chemical, Disease	Disease	Aspect Term	Biological Process, Molecular Function, Cellular Component	Food

Fuzzy Neural Model. The tie-or-break tagging schema encodes the entity spans and entity types into two folds. Therefore, the PATNER neural model learning is divided into two steps: entity span detection and entity typing. For entity span detection, a binary classifier is built to determine whether a connection has a label of *Break* or *Tie*. A BiLSTM layer is utilized to encode the character and word embeddings to predict whether the connection y_i between tokens w_{i-1} and w_i is *Break*. Then the output of the BiLSTM layer will be concatenated as one vector u_i and fed into a Sigmoid layer,

$$p(y_i = Break|u_i) = \sigma(w^T u_i).$$

The loss function of entity span detection is

$$\mathcal{L}_1 = \sum_{y_i=Break} l(y_i, p(y_i = Break|u_i)),$$

where $l(\cdot,\cdot)$ is the logistic loss.

After the entity boundary is determined, each token span is represented with a new vector v_j and fed into a Softmax layer to determine its entity type. Let $\mathcal{T} = \mathcal{T}_e \cup \{\text{None}\}$,

$$p(t_j = t|v_j) = \frac{\exp(e_t^T v_j)}{\sum_{t' \in \mathcal{T}} \exp(e_{t'}^T v_j)},$$

where t_j denotes the label of entity span j and e_t is the embedding vector of the entity type $t \in \mathcal{T}$. The loss function of entity type prediction is

$$\mathcal{L}_2 = \sum_j H(\hat{p}(\cdot|v_i, \mathcal{T}), p(\cdot|v_i)),$$

where $H(\cdot,\cdot)$ is the cross entropy function and $\hat{p}(\cdot|v_i, \mathcal{T})$ is the soft supervision distribution. Since L_e is the label distribution of entity e_{t_j} that we get from the previous GMM model,

$$\hat{p}(t_j|v_j, \mathcal{T}) = \frac{L_e(e_{t_j}, t_j)\exp(e_{t_j}^T v_j)}{\sum_{t' \in \mathcal{T}} L_e(e_{t'}, t')\exp(e_{t'}^T v_j)}.$$

III. EXPERIMENTAL EVALUATION

We evaluate the performance of PATNER on three benchmark datasets in two domains. We compare the performance of PATNER with the supervised benchmark and the state-of-the-art distantly supervised NER models on three benchmark datasets. We also compare the performance of PATNER with existing distantly supervised NER models on two additional real-world datasets with new entity types (e.g., biological process) where human annotation is not available. We further investigate the effects of corpus size and dictionary size in PATNER. The results demonstrate the power of PATNER on a wide variety of entity types.

A. Experiment Setup

Datasets. The datasets and dictionaries used in our experiments are summarized in Table II. Three benchmark datasets (BC5CDR, NCBI-disease and LaptopReview) are used for quantitative study. Two realworld datasets (PubMed and Yelp) are used for case studies on new entity types without human annotation.

- **BC5CDR** [10] is a benchmark dataset released in the BioCreative V Chemical Disease Relation task. It contains 1,500 articles with 15,935 Chemical and 12,852 Disease entity mentions. The whole corpus is divided into three parts, each having 500 articles, for training, development and testing.
- **NCBI-Disease** [4] is a benchmark dataset for disease entity recognition. The corpus contains 793 abstracts with 6,881 Disease entities, and it is separated into three subsets: training (593), development (100) and testing (100).
- **LaptopReview** [6] is from the SemEval 2014 Challenge, Task 4 Subtask 1, focusing on laptop Aspect Term (e.g., "disk drive") Recognition. It consists of 3,845 review sentences and 3,012 AspectTerm mentions.
- **PubMed_3M** is a subset of the PubMed[1] abstracts of biomedical literature. It consists of 19,204 unlabeled sentences.
- **Yelp_3M** is a subset of the Yelp[2] review dataset. It consists of 39,924 unlabeled sentences.

The three benchmark datasets (BC5CDR, NCBI-disease and LaptopReview) are publicly available. In addition to the benchmark datasets, we collect two other datasets in the biomedical domain (PubMed) and the general domain (Yelp), respectively, as a case study for new entity type recognition. We use the same data splitting as in [26] for the benchmark datasets. For supervised method, the training and development sets are used for training. For distantly supervised methods, the entire corpus without human annotation is used as the raw input corpus.

Domain-Specific Dictionaries. We use the same dictionaries as in [26] for the benchmark datasets. For the biomedical datasets, we use a combination of the MeSH[3] and CTD[4]

[1]https://www.ncbi.nlm.nih.gov/pubmed/
[2]https://www.yelp.com/dataset
[3]https://www.nlm.nih.gov/mesh/
[4]http://ctdbase.org

Authorized licensed use limited to: University of Illinois. Downloaded on May 05,2021 at 20:55:00 UTC from IEEE Xplore. Restrictions apply.

databases for the chemical and disease vocabularies. For the laptop review dataset, we use 13,457 computer terms crawled from a public website[5]. For the new entity types, we collect three entity types (biological process, molecular function and cellular component) from Gene Ontology[6] for PubMed_3M and one entity type (food) from Open Food Facts[7] for Yelp_3M. All the dictionaries are publicly available.

Baseline Methods. We evaluate the performance of PATNER by comparing with four groups of methods.

Supervised benchmark:

- On the biomedical datasets, BiLSTM-CRFs [31] and BioBERT [9] are used as supervised baselines. On BC5CDR and NCBI-Disease, *LM-LSTM-CRF* [11] and *LSTM-CRF* [8] are used respectively [31]. BioBERT [9] is a recent work that train the BERT [3] word embedding on the whole PubMed and PubMed Central. BioBERT is fine-tuned on several text mining tasks including biomedical named entity recognition.
- On the LaptopReview dataset, the highest score [19], [29] in the SemEval2014 Challenge Task 4 Subtask 1 with the in-domain training dataset are presented. Note that in the SemEval2014 Challenge, a higher performance can be achieved using additional annotated data for training and feature extraction. BiLSTM-CRFs [1] and BERT [32] are also used as supervised baselines on LaptopReview.

Domain-specific distantly supervised methods:

- *SwellShark* [5] is a benchmark distantly supervised method in the biomedical domain. It needs no human annotated data. However, it requires extra expert effort for entity span detection on building POS tagger, designing effective regular expressions, and hand-tuning for special cases. It is compared on BC5CDR and NCBI-Disease datasets only.
- *Distant-LSTM-CRF* [6] is specifically designed for Aspect Terms Extraction using distant supervision. It uses the sentiment lexicon and human provided syntactic rules in the ATE task. It is compared on LaptopReview dataset only.

Dictionary-based methods:

- *Dictionary-Match* is a simple baseline. We first generate quality phrases using AutoPhrase [25] and then type the entities if they can be matched from the dictionary.
- PATNER *(w/o neural model)* conducts a dictionary enrichment by *GMM-based Expansion* (Section II-B).

Distantly supervised methods:

- *Fuzzy-LSTM-CRF* [26] is adapted from LSTM-CRF [8]. The char- and word-level BiLSTM architecture is retained but the CRF layer becomes Fuzzy-CRF so that it can support a "modified BIOES" scheme.
- AUTONER [26] uses *Dictionary-Match* result as training, and then applies the BiLSTM-Softmax neural model with "tie-or-break" scheme.
- PATNER *(w/o fuzzy label)* uses PATNER *(w/o neural model)* result as training and then applies the BiLSTM-Softmax neural model with "tie-or-break" scheme.
- PATNER is the proposed full model. It uses PATNER *(w/o neural model)* result as training and then applies the fuzzy BiLSTM-Softmax neural model with "tie-or-break" scheme.

Parameters. The optimization method is gradient descent with momentum. The batch size and the momentum are set to be 10 and 0.9. The learning rate is set to 0.05. The dropout ratio is set to 0.5. Gradient clipping of 5.0 is used.

Pre-trained Word Embeddings. For the biomedical datasets, we use the 200-dimension word embeddings[8] from [16]. For the laptop review dataset, we use the GloVe 100-dimension word embeddings[9].

Evaluation Metric. Evaluation is conducted on testing sets for all methods. We compare the performance in terms of precision, recall, and the micro-averaged F1 score.

B. Experiments on Benchmark Datasets

The performance (entity-level F1, precision and recall scores) of each method on the benchmark datasets are shown in Tables III, IV.

We first compare all the methods without human effort for training data annotation. Among the dictionary-based methods, Dictionary-Match has high precision on all benchmark datasets (over 90%), but the recalls are rather low (lower than 60% on biomedical domain and lower than 45% on technical review). This result validates our observation that dictionaries usually have precise entity names but low coverage. PATNER *(w/o neural model)* achieves significantly better performance than dictionary matching with a comparable precision and a dramatically boosted recall. Noticeably and surprisingly, on NCBI-Disease, the F1 score of PATNER *(w/o neural model)* even outperforms a Supervised Benchmark BiLSTM-CRFs. The key lies in the high precision and recall of the GMM-based dictionary expansion. Among the distantly supervised methods, since both AUTONER and PATNER *(w/o fuzzy label)* use the same neural model, the key factor is the quality of distant training. PATNER *(w/o fuzzy label)* outperforms AUTONER by a substantial margin (4.76% for BC5CDR, 16.05% for NCBI-Disease and 5.27% for LaptopReview in F1). Because the training set of PATNER *(w/o fuzzy label)* obtained from PATNER *(w/o neural model)* achieves significantly better performance than Dictionary-Match, the training set of AUTONER. PATNER uses the same training as PATNER *(w/o fuzzy label)* but a different neural model. On all benchmark datasets, PATNER constantly achieves better performance compared with the ablation model PATNER *(w/o*

[5]https://www.computerhope.com/jargon.htm
[6]http://geneontology.org/
[7]https://world.openfoodfacts.org/
[8]http://bio.nlplab.org/
[9]https://nlp.stanford.edu/projects/glove/

Authorized licensed use limited to: University of Illinois. Downloaded on May 05,2021 at 20:55:00 UTC from IEEE Xplore. Restrictions apply.

TABLE III
[BIOMEDICAL DOMAIN] NER PERFORMANCE COMPARISON. OUR METHODS ARE MARKED IN BOLD.

Method	Human Effort	BC5CDR			NCBI-Disease		
		Prec	Rec	F1	Prec	Rec	F1
BiLSTM-CRFs [31]	Gold Annotations	88.84	85.16	86.96	86.11	85.49	85.80
BioBERT [9]		90.45	**90.86**	**90.65**	89.04	89.69	89.36
SwellShark [5]	Regex Design + Special Case	86.11	82.39	84.21	81.6	80.1	80.8
	Regex Design	84.98	83.49	84.23	64.7	69.7	67.1
Dictionary-Match [26]	None	**93.93**	58.35	71.98	90.59	56.15	69.32
PATNER (w/o neural model)		91.95	76.37	83.44	**92.86**	84.06	88.24
Fuzzy-LSTM-CRF [26]	None	88.27	76.75	82.11	79.85	67.71	73.28
AUTONER [26]		88.96	81.00	84.79	79.42	71.98	75.52
PATNER (w/o fuzzy label)		90.61	88.51	89.55	91.48	91.67	**91.57**
PATNER		90.92	89.69	**90.31**	92.54	**91.77**	**92.15**

TABLE IV
[TECHNICAL REVIEW] NER PERFORMANCE COMPARISON. OUR METHODS ARE MARKED IN BOLD.

Method	LaptopReview		
	Prec	Rec	F1
SemEval-2014 [29]	79.31	63.30	70.41
BiLSTM-CRFs [1]	-	-	**81.08**
BERT [32]	-	-	**84.26**
Distant-LSTM-CRF [6]	74.03	31.59	53.93
Dictionary-Match [26]	90.68	44.65	59.84
PATNER (w/o neural model)	**91.00**	54.13	67.88
Fuzzy-LSTM-CRF [26]	85.08	47.09	60.63
AUTONER [26]	72.27	59.79	65.44
PATNER (w/o fuzzy label)	78.65	64.22	70.71
PATNER	78.45	**65.14**	**71.18**

fuzzy label). These results demonstrate that the incorporation of uncertainty in the entity expansion can improve the BiLSTM-Softmax neural model.

We also compare PATNER with the methods that require human effort for special case tuning or training data annotation. Among all the distantly supervised methods, even though SwellShark and Distant-LSTM-CRF utilize more domain-specific features and expert effort, PATNER outperforms them by a large margin (6.08% for BC5CDR, 11.35% for NCBI-Disease and 17.25% for LaptopReview in F1). Comparing with the supervised NER methods, PATNER outperforms the BiLSTM-CRFs by a substantial margin (3.35% for BC5CDR, 6.35% for NCBI-Disease in F1) on the biomedical datasets. PATNER has comparable performance with BioBERT on BC5CDR (90.31% v.s 90.65% in F1) and even outperforms BioBERT on the NCBI-Disease dataset by a substantial margin (2.79% in F1). One reason that PATNER has a better performance in the biomedical domain could be that the biomedical entities usually follow more standard naming principals that enhance the PATNER performance. In summary, PATNER is the most competitive distantly supervised NER method on three benchmark datasets in both biomedical and technical domains.

C. Top Patterns and Entities for Expansion

The above experiments show that all the components of PATNER can improve the performance of the NER task under distant supervision. To further illustrate the advantages of PATNER, we list some top quality patterns and top quality entities extracted by PATNER in Tables V and VI, respectively.

The top patterns extracted are of high quality, and thus can accurately label more candidate entities. For example, words that contain "ine" or "amin" are likely to be chemical-related names. Phrases that contain the words such as "_acid_", "_antagonists_" or "_inhibitors_" are strong indicators that certain phrase is a chemical name. Similarly, words that contain "tion" or "som" are likely to be disease-related names. Phrases that contain the words such as "_syndrome_", "_disorder_" or "_diseases_" are strong indicators that certain phrase is a chemical name. In the technical review domain, the word patterns are more abundant and informative for dictionary expansion. For example, phrases that contain the words such as "_DIGIT_", "_screen_" and "_card_" are strong indicators that certain phrase is a aspect term in the laptop reviews.

The top entities extracted are also of high quality. For example, the top entities extracted as chemicals are "R-(alpha)-methylhistamine" and "metamphetamine". The top entities extracted for BC5CDR-Disease "hepatic complication" and "hepatic dysfunction". Similar results are also observed in the technical review domain. The top entities extracted as aspect terms are "16GB RAM" and "DDR5" that are of high quality.

D. Experiments with Different Factors

We further investigate the effect of corpus size and dictionary size on the NER performance to better apply PATNER in real-world applications. The results are shown in Figure 3.

To test the performance (F1 scores) with respect to the size of raw corpus, we sample sentences uniformly random from the given raw corpus and then evaluate the performance of PATNER model on the selected sentences. When the raw corpus is small, PATNER does not perform as good as the Supervised Benchmark. Once the distantly labeled corpus is moderate in size (5,000 sentences), PATNER already achieves a similar performance compared with BiLSTM-CRF trained

Authorized licensed use limited to: University of Illinois. Downloaded on May 05,2021 at 20:55:00 UTC from IEEE Xplore. Restrictions apply.

TABLE V
TOP PATTERNS FROM PATNER IN BENCHMARK DATASETS.

BC5CDR-Chemical		BC5CDR-Disease		NCBI-Disease		LaptopReview-AspectTerm	
Character	Word	Character	Word	Character	Word	Character	Word
ne_	_acid_	al_	_syndrome_	ia_	_DIGIT_	ard_	_DIGIT_
ine	_factor_	ion	_disorder_	er_	_syndrome_	-	_screen_
ine_	_agents_	ic_	_pain_	rom	_DIGIT_LETTER_	-	_card_
ate	_antagonists_	tion	_coronary_	eas	_ataxia_	-	_mouse_
ami	_factor_LETTER_	ion_	_cancer_	ome_	_TYPE_DIGIT_	-	_drive_

TABLE VI
TOP EXPANDED ENTITIES FROM PATNER IN BENCHMARK DATASETS.

BC5CDR-Chemical	BC5CDR-Disease	NCBI-Disease	LaptopReview-AspectTerm
R-(alpha)-methylhistamine	hepatic complication	spinocerebellar ataxia	16GB RAM
metamphetamine	hepatic dysfunction	chromosome 15 anomaly	DDR5
dexamphetamine	atrial fibrillation	cerebellar degeneration	DDR3
amphetamine	neuroleptic medications	spinocerebellar ataxia type 3	Win7
5-hydroxytryptamine	Staphylococcal infections	Gardner syndrome	Win8

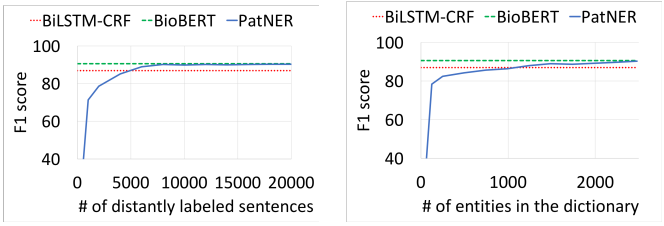

Fig. 3. Left: test F1 scores vs. the number of distantly labeled training sentences for BC5CDR. Right: test F1 scores vs. the number of entities in the dictionary for BC5CDR.

on 20,000 human-annotated sentences. Further increasing the corpus size beyond 8,000 sentences does not further improve the PATNER performance.

To test the performance (F1 scores) with respect to the size of the input dictionary, we sample entities uniformly random from the given dictionary and then evaluate the performance of PATNER on the raw corpus. When the input dictionary is small, PATNER does not perform as good as the Supervised Benchmark. Once the dictionary is moderate in size (1,200 entities), PATNER already achieves a similar performance compared with BiLSTM-CRF trained on 20,000 human-annotated sentences. Further increasing the dictionary size tends to constantly increase the PATNER performance. These results further demonstrate the importance of developing a high quality dictionary expansion method in improving the NER performance under distant supervision.

E. Running Time Comparison

We show that PATNER significantly outperforms the most competitive supervised methods, BiLSTM-CRFs and BioBERT, in Section III-B. In this section, we further show the simplicity of our model compared with those methods.

BioBERT is trained on 8 NVIDIA V100 (32GB) GPUs for the pre-training. It takes more than 10 days to pre-train BioBERT on the whole PubMed and PubMed Central [9]. Both Supervised Benchmark and PATNER use pre-trained word embeddings[10] with word2vec on the same corpus (whole PubMed and PubMed Central) [15], [16]. BioBERT is fine-tuned on a single NVIDIA Titan Xp (12GB) GPU for each BioNER task. Fine-tuning usually takes less than an hour in average for each dataset [9]. Both BiLSTM-CRFs and PATNER are trained on one GeForce GTX 1080 GPU. The average training time for each dataset is an hour and 0.5 hour for Supervised Benchmark and PATNER, respectively. The neural architecture of PATNER does not include a CRF layer (Figure 2). Therefore, the distant training step can be accelerated substantially in contrast to the popular LSTM-CRF models.

F. New Entity Type Recognition

To further demonstrate the power of PATNER on recognizing new entity types, we conduct some qualitative case studies on two real-world datasets with no human annotation available, in the biomedical domain (PubMed_3M) and the general domain (Yelp_3M), respectively.

We show some NER tagging results on PubMed and Yelp in Table VII. The new entities types are Biologcal Process (BP), Molecular Function (MF) and Cellular Component (CC) for PubMed and Food for Yelp. We compare three distant supervision methods: Dictionary-Match, AUTONER and PATNER. PATNER always shows the best performance both in entity boundary and entity type recognition. For example, in Sentence 1, the input dictionary cannot match any entities. AUTONER recognizes a new entity "apoptotic cell death" as biological process and PATNER recognizes both "apoptotic cell death" and "oxidative damage" as biological process. In Sentence 2, AUTONER recognizes "signaling" as biological process, while PATNER recognizes the whole entity "signaling pathway" as a biological process correctly. In Sentence 3, PATNER recognizes a new entity "enzyme activity" as molecular

[10]http://bio.nlplab.org/

Authorized licensed use limited to: University of Illinois. Downloaded on May 05,2021 at 20:55:00 UTC from IEEE Xplore. Restrictions apply.

TABLE VII
CASE STUDY OF THE DISTANT NER METHODS ON PUBMED AND YELP.

		PubMed
1	Dictionary	This results ... but also in ROS overproduction, oxidative damage, and apoptotic cell death ...
	AUTONER	This results ... but also in ROS overproduction, oxidative damage, and [**apoptotic cell death**]$_{BP}$ ...
	PATNER	This results ... but also in ROS overproduction, [**oxidative damage**]$_{BP}$, and [**apoptotic cell death**]$_{BP}$ ...
2	Dictionary	We focused on striatal alterations in intracellular signaling pathways, oxidative stress and [**cell death**]$_{BP}$.
	AUTONER	We focused on striatal alterations in [**intracellular**]$_{CC}$ [**signaling**]$_{BP}$ pathways, oxidative stress and [**cell death**]$_{BP}$.
	PATNER	We focused on striatal alterations in [**intracellular**]$_{CC}$ [**signaling pathways**]$_{BP}$, [**oxidative stress**]$_{BP}$ and [**cell death**]$_{BP}$.
3	Dictionary	Resveratrol increased endogenous and substrate-supported cortisol production like nonhydroxylated flavones tested, but it had no effect on CYP11B1 gene expression and enzyme activity.
	AUTONER	Resveratrol increased endogenous and substrate-supported cortisol production like nonhydroxylated flavones tested, but it had no effect on CYP11B1 [**gene expression**]$_{BP}$ and enzyme activity.
	PATNER	Resveratrol increased [**endogenous**]$_{CC}$ and substrate-supported [**cortisol production**]$_{BP}$ like nonhydroxylated flavones tested, but it had no effect on CYP11B1 [**gene expression**]$_{BP}$ and [**enzyme activity**]$_{MF}$.
		Yelp
4	Dictionary	We had their beef tartar and pork belly to start and a [**salmon**]$_{Food}$ dish and lamb meal for mains.
	AUTONER	We had their [**beef tartar**]$_{Food}$ and [**pork**]$_{Food}$ belly to start and a [**salmon**]$_{Food}$ dish and [**lamb**]$_{Food}$ meal for mains.
	PATNER	We had their [**beef tartar**]$_{Food}$ and [**pork belly**]$_{Food}$ to start and a [**salmon**]$_{Food}$ dish and [**lamb meal**]$_{Food}$ for mains.
5	Dictionary	My two favourite dishes are the [**rice flour**]$_{Food}$ [**rolls**]$_{Food}$ and the chicken pho.
	AUTONER	My two favourite dishes are the [**rice flour**]$_{Food}$ [**rolls**]$_{Food}$ and the [**chicken**]$_{Food}$ pho.
	PATNER	My two favourite dishes are the [**rice flour rolls**]$_{Food}$ and the [**chicken pho**]$_{Food}$.
6	Dictionary	We ordered 3 appetizers : spinach and artichoke dip , [**chicken**]$_{Food}$ lettuce wraps , and the [**mussels**]$_{Food}$.
	AUTONER	We ordered 3 appetizers : [**spinach**]$_{Food}$ and [**artichoke dip**]$_{Food}$, [**chicken**]$_{Food}$ lettuce [**wraps**]$_{Food}$, and the [**mussels**]$_{Food}$.
	PATNER	We ordered 3 appetizers : [**spinach and artichoke dip**]$_{Food}$, [**chicken lettuce wraps**]$_{Food}$, and the [**mussels**]$_{Food}$.

function that has never been recognized by the other methods. Similar results are observed in the Yelp dataset. For example, in Sentence 4, AUTONER recognizes "pork" and "lamb" as food, while PATNER recognizes the whole entity "pork belly" and "lamb meal" as food correctly. Similary, in Sentence 5, AUTONER recognizes "rice flour" and "rolls" as two separate food entities, while PATNER recognizes the whole entity "rice flour rolls" as food correctly. In Sentence 6, the input dictionary matches some sort entities such as "chicken" and "mussels". AUTONER recognizes some new short entities, such as "spinach", "artichoke dip" and "wraps". PATNER recognizes all the three appetizers "spinach and artichoke dip", "chicken lettuce wraps" and "mussels" correctly. These case studies on the real-world datasets further demonstrate the power of PATNER and show the potential of dictionary-based distant supervision in the NER task.

IV. RELATED WORK

As an important task in natural language processing, named entity recognition (NER) has been studied for decades. Taken expert-curated training data, early techniques explore hand-crafted features, using methods like hidden Markov models (HMMs) [33], [34] and conditional random fields (CRFs) [12], [14]. In recent years, deep learning models, such as recurrent neural networks (RNNs), have been widely applied to NER, achieving state-of-the-art results [2], [7], [8], [11], [13], [31].

Aiming to reduce expensive human labor, distant supervision attracts recent attention especially in more specific domains. The major research effort in the distantly supervised NER task lies in how to better utilize entity information in dictionaries or knowledge bases. In the general domain, Ren et al. [20]–[23] link mentions to knowledge base and use linked entities to infer the types of unlinked ones using label propagation on heterogeneous graphs. Distant-LSTM-CRF [6] has been proposed for the extraction of aspect terms, a task closely related to NER. In the biomedical domain, there are also NER designs for the distant supervision setting [5], [26]. SwellShark [5], specifically designed for biomedical NER, leverages a generative model to unify and model noise across different supervision sources for named entity typing. However, it leaves the named entity span detection to a heuristic combination of dictionary matching and part-of-speech tag-based regular expressions, which requires extensive expert effort to cover many special cases.

The most related work to the proposed PATNER is AUTONER [26] that uses a "tie-or-break" tagging scheme to leverage distant supervision from entity dictionaries. Comparing with traditional "IOBES" tagging scheme, "tie-or-break" tagging scheme can reduce the effects of false negative issue brought by the low recall of dictionary, and thus achieves better performance. However, comparing with human supervision, there is still a significant gap in performance, due to the low recall of dictionaries. The proposed PATNER addresses this issue by mining patterns (i.e., entity naming principles) to enhance the distant supervision. A fuzzy NER model is developed by revising the neural model to incorporate the uncertainty in dictionary expansion.

Authorized licensed use limited to: University of Illinois. Downloaded on May 05,2021 at 20:55:00 UTC from IEEE Xplore. Restrictions apply.

V. Conclusions

We have proposed PatNER, a distantly supervised NER model that effectively deals with noisy distant supervision from domain-specific dictionaries. PatNER does not require human-annotated training data but relies on unlabeled data and incomplete domain-specific dictionaries for distant supervision. It automatically mines the entity naming principles from dictionaries and enhances distant supervision with a fuzzy NER neural model to incorporate the uncertainty of dictionary expansion. Extensive experiments on three benchmark datasets in two domains demonstrate the power of PatNER.Case studies on two additional real-world datasets demonstrate that PatNER improves the distant NER performance in both entity boundary detection and entity type recognition. The results show a great promise in supporting high quality named entity recognition with domain-specific dictionaries on a wide variety of entity types. Future work may include extending the study to more entity types and fine-grained type levels.

Acknowledgment

Research was sponsored in part by US DARPA KAIROS Program No. FA8750-19-2-1004 and SocialSim Program No. W911NF-17-C-0099, National Science Foundation IIS-19-56151, IIS-17-41317, IIS 17-04532, and IIS 16-18481, and DTRA HDTRA11810026. Any opinions, findings, and conclusions or recommendations expressed herein are those of the authors and should not be interpreted as necessarily representing the views, either expressed or implied, of DARPA or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for government purposes notwithstanding any copyright annotation hereon.

References

[1] Ł. Augustyniak, T. Kajdanowicz, and P. Kazienko. Comprehensive analysis of aspect term extraction methods using various text embeddings. *arXiv preprint arXiv:1909.04917*, 2019.

[2] J. P. C. Chiu and E. Nichols. Named Entity Recognition with Bidirectional LSTM-CNNs. *Trans. Assoc. Comput. Linguist.*, 4:357–370, 2016.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186, Minneapolis, Minnesota, June 2019. ACL.

[4] R. I. Doğan, R. Leaman, and Z. Lu. Ncbi disease corpus: a resource for disease name recognition and concept normalization. *Journal of biomedical informatics*, 47:1–10, 2014.

[5] J. Fries, S. Wu, A. Ratner, and C. Ré. Swellshark: A generative model for biomedical named entity recognition without labeled data. *arXiv preprint arXiv:1704.06360*, 2017.

[6] A. Giannakopoulos, C. Musat, A. Hossmann, and M. Baeriswyl. Unsupervised aspect term extraction with b-lstm & crf using automatically labelled datasets. *In Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 180–188, 2017.

[7] M. Habibi, L. Weber, M. Neves, D. L. Wiegandt, and U. Leser. Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics*, 33(14):i37–i48, 2017.

[8] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer. Neural architectures for named entity recognition. In *NAACL-HLT*, pages 260–270. ACL, 2016.

[9] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang. Biobert: pre-trained biomedical language representation model for biomedical text mining. *arXiv preprint arXiv:1901.08746*, 2019.

[10] J. Li, Y. Sun, R. Johnson, D. Sciaky, C.-H. Wei, R. Leaman, A. P. Davis, C. J. Mattingly, T. C. Wiegers, and Z. Lu. Annotating chemicals, diseases, and their interactions in biomedical literature. In *Proceedings of the fifth BioCreative challenge evaluation workshop*, pages 173–182, 2015.

[11] L. Liu, J. Shang, F. Xu, X. Ren, H. Gui, J. Peng, and J. Han. Empower Sequence Labeling with Task-Aware Neural Language Model. In *AAAI*, pages 5245–5253, 2018.

[12] Y. Lu, D. Ji, X. Yao, X. Wei, and X. Liang. CHEMDNER system with mixed conditional random fields and multi-scale word clustering. *J. Cheminf.*, 7(S1):S4, 2015.

[13] X. Ma and E. Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *ACL*, pages 1064–1074, 2016.

[14] R. McDonald and F. Pereira. Identifying gene and protein mentions in text using conditional random fields. *BMC bioinformatics*, 6(1), 2005.

[15] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119. MIT Press, 2013.

[16] S. Moen and T. S. S. Ananiadou. Distributional semantics resources for biomedical text processing. *LBM*, pages 39–44, 2013.

[17] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu. Mining sequential patterns by pattern-growth: The prefixspan approach. *TKDE*, 16(11):1424–1440, 2004.

[18] M. Peng, X. Xing, Q. Zhang, J. Fu, and X. Huang. Distantly supervised named entity recognition using positive-unlabeled learning. *ACL*, 2019.

[19] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, A.-S. Mohammad, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq, et al. Semeval-2016 task 5: Aspect based sentiment analysis. In *SemEval*, pages 19–30, 2016.

[20] X. Ren, A. El-Kishky, C. Wang, F. Tao, C. R. Voss, and J. Han. Clustype: Effective entity recognition and typing by relation phrase-based clustering. In *KDD*, pages 995–1004. ACM, 2015.

[21] X. Ren, W. He, M. Qu, L. Huang, H. Ji, and J. Han. Afet: Automatic fine-grained entity typing by hierarchical partial-label embedding. In *EMNLPs*, pages 1369–1378, 2016.

[22] X. Ren, W. He, M. Qu, C. R. Voss, H. Ji, and J. Han. Label noise reduction in entity typing by heterogeneous partial-label embedding. In *KDD*, pages 1825–1834. ACM, 2016.

[23] X. Ren, Z. Wu, W. He, M. Qu, C. R. Voss, H. Ji, T. F. Abdelzaher, and J. Han. Cotype: Joint extraction of typed entities and relations with knowledge bases. In *WWW*, pages 1015–1024. IW3C, 2017.

[24] S. Riedel, L. Yao, A. McCallum, and B. M. Marlin. Relation extraction with matrix factorization and universal schemas. In *ACL*, pages 74–84, 2013.

[25] J. Shang, J. Liu, M. Jiang, X. Ren, C. R. Voss, and J. Han. Automated phrase mining from massive text corpora. *TKDE*, 2018.

[26] J. Shang, L. Liu, X. Ren, X. Gu, T. Ren, and J. Han. Learning named entity tagger using domain-specific dictionary. In *EMNLP*. ACL, 2018.

[27] W. Shen, J. Wang, P. Luo, and M. Wang. Linden: linking named entities with knowledge base via semantic knowledge. In *WWW*, pages 449–458. ACM, 2012.

[28] J. C. Spall and J. L. Maryak. A feasible bayesian estimator of quantiles for projectile accuracy from non-iid data. *Journal of the American Statistical Association*, 87(419):676–681, 1992.

[29] Z. Toh and W. Wang. Dlirec: Aspect term extraction and term polarity classification system. In *SemEval*, pages 235–240, 2014.

[30] X. Wang, Y. Zhang, Q. Li, X. Ren, J. Shang, and J. Han. Distantly supervised biomedical named entity recognition with dictionary expansion. In *IEEE-BIBM, pages=, year=2019*.

[31] X. Wang, Y. Zhang, X. Ren, Y. Zhang, M. Zitnik, J. Shang, C. Langlotz, and J. Han. Cross-type biomedical named entity recognition with deep multi-task learning. *Bioinformatics*, page bty869, 2018.

[32] H. Xu, B. Liu, L. Shu, and P. S. Yu. Bert post-training for review reading comprehension and aspect-based sentiment analysis. In *NAACL*, jun 2019.

[33] G. Zhou and J. Su. Named entity recognition using an hmm-based chunk tagger. In *ACL*, pages 473–480. ACL, 2002.

[34] G. Zhou and J. Su. Exploring deep knowledge resources in biomedical name recognition. In *Proc. Int. Jt. Work. Nat. Lang. Process. Biomed. its Appl.*, pages 96–99, 2004.

[35] J. Zhu, V. Uren, and E. Motta. Espotter: Adaptive named entity recognition for web browsing. In *Biennial Conference on Professional Knowledge Management/Wissensmanagement*, pages 518–529. Springer, 2005.

Authorized licensed use limited to: University of Illinois. Downloaded on May 05,2021 at 20:55:00 UTC from IEEE Xplore. Restrictions apply.