Boosting Entity Recognition by leveraging Cross-task Domain Models for Weak Supervision

Sanjay Agrawal
sanjagr@amazon.com
Amazon.com Inc.
Bengaluru, India

Srujana Merugu
smerugu@amazon.com
Amazon.com Inc.
Bengaluru, India

Vivek Sembium
viveksem@amazon.com
Amazon.com Inc.
Bengaluru, India

ABSTRACT

Entity Recognition (ER) is a common natural language processing task encountered in a number of real-world applications. For common domains and named entities such as places and organisations, there exists sufficient high quality annotated data and foundational models such as T5 and GPT-3.5 also provide highly accurate predictions. However, for niche domains such as e-commerce and medicine with specialized entity types, there is a paucity of labeled data since manual labeling of tokens is often time-consuming and expensive, which makes entity recognition challenging for such domains. Recent works such as NEEDLE [48] propose hybrid solutions to efficiently combine a small amount of strongly labeled (human-annotated) with a large amount of weakly labeled (distant supervision) data to yield superior performance relative to supervised training. The extensive noise in the weakly labeled data, however, remains a challenge. In this paper, we propose WeSDoM (**We**ak **S**upervision with **Do**main **M**odels), which leverages pretrained encoder models from the same domain but different tasks to create domain ontologies that can enable the creation of less noisy weakly labeled data. Experiments on internal e-commerce and public biomedical NER datasets demonstrate that WeSDoM outperforms existing SOTA baselines by a significant margin. We achieve new SOTA F1 scores on two popular Biomedical NER datasets, BC5CDR-chem 94.27, BC5CDR-disease 91.23.

CCS CONCEPTS

• **Information systems** → **Ontologies**; • **Computing methodologies** → **Information extraction**.

KEYWORDS

Entity recognition, Cross-Task Domain Encoder, Weak Supervision, Ontologies

ACM Reference Format:

Sanjay Agrawal, Srujana Merugu, and Vivek Sembium. 2024. Boosting Entity Recognition by leveraging Cross-task Domain Models for Weak Supervision. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24), October 21–25, 2024, Boise, ID, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3627673.3680009

CIKM '24, October 21–25, 2024, Boise, ID, USA

ACM ISBN 979-8-4007-0436-9/24/10.
https://doi.org/10.1145/3627673.3680009

1 INTRODUCTION

Entity recognition (ER), i.e., segmenting a text sequence and labeling the segments along predefined entity types, is one of the most fundamental and well-studied natural language processing (NLP) tasks with multiple downstream applications related to structured information extraction and relevance matching [35] [33] [40]. For general domains entity types, such as PERSON and LOCATION, there is ample labeled data [36] [43] and most large foundational language models such as T5[34], GPT-3 [4] yield highly accurate predictions. However, this does not hold true for niche domains such as e-commerce with specialized entity types such as product types, attribute names, attribute values, attribute units that cannot be readily discriminated based on generic language skills such as parts-of-speech. For these domains, there is highly limited availability of human annotated data (i.e., strongly labeled data) since manual labeling of text tokens requires significant time and effort. Therefore, despite the ready availability of transformer models [41] [10], building highly accurate ER models for such domains continues to remain a challenge. To address the dependence on "strong labels" from human annotations, many researchers have experimented with "weak labels", which are automated annotations of large unlabeled datasets using techniques based on knowledge bases (dictionaries), heuristic rule-based approaches, or the output of a legacy NER model trained on a different dataset [30] [28] [20]. These weak labels are often noisy due to **incompleteness of annotation** because of limited coverage of the weak annotators, **labeling errors and bias** since the weak annotators might generate inaccurate labels. Furthermore, there is a substantial **difference in the scale of the weak and strong labels** since the unlabeled data used for weak supervision could be a few orders of magnitude larger than that of human-annotated data as a result of which the combined data could significantly deviate from the true data distribution. It has been shown in [48] that training powerful deep transformer models with a naive combination of the strong and weak labeled data invariably leads to over-fitting on the noisy weak labels and deterioration in the model performance relative to just using the limited strong labels. Approaches to suppress the noise in the weakly labeled data include Semi-supervised Self-Training[1] [11], regularization for the weakly labelled data loss [48] etc. Recently, there is also a heavy interest in improving performance on NLP tasks via synthetic labeled data using large foundational general purpose generative models such as T5, GPT-3.5. However, there is limited work that tries to exploit existing domain-specific encoder-based LLMs, potentially trained on a different task.

[1] generate pseudo labels for unlabeled data based on the model learned through supervised learning

Our work is motivated by the assumption that regardless of the prediction task, domain-specific language models encode rich representations of textual content, which can be combined with an unlabeled domain corpus to enhance the quality of automated weak supervision. Specifically, we consider a practical entity recognition scenario, where we have access to (a) a pre-trained cross-task encoder[2] (can even be blackbox API) from the same domain, (b) a large corpus of unlabeled data. We investigate the following research questions: **RQ1:** Can the availability of a cross-task domain model aid in improving the quality of weakly-labeled data and the overall performance ? **RQ2:** What is the relative utility of the unlabeled data and the strongly labeled data? **RQ3:** What are the key steps that contribute to effective utilisation of the unlabeled data and a cross-task model? To address these questions, we propose WeSDoM, a novel NER approach that utilises unlabeled data along with a cross-task legacy model to create relatively noise-free domain ontologies that can function as weak annotators and explore how it performs in various settings. Our **key contributions** can be summarised as follows:

1. We propose a three-stage NER solution framework for utilising cross-task encoder models where (a) Stage 1 [Ontology creation] comprises automated extraction and clustering of domain phrases from unlabeled data based on embeddings from the cross-task model to create domain ontologies, (b) Stage 2 [Weak supervision] consists of annotating the unlabeled data for the target ER task using the ontologies, and (c) Stage 3 [ER Model training] comprises building a custom ER model by appropriate pretraining and fine-tuning using both the strongly and weakly labeled data.
2. We present empirical results on both internal and public datasets which point to the benefits of the proposed approach relative to the existing SOTA methods on utilizing weakly labeled data. We also present an in-depth error diagnosis and ablation studies to assess the relative utility of various modeling steps.
3. Lastly, we present an analysis of the relative utility of labeled data and unlabeled data accompanied by a cross-task model pointing to the benefits of a hybrid strategy.

Note that our methodology, i.e., using a cross-task domain encoder to curate intermediate ontologies or KBs (knowledge base) from unlabeled data and generate weak annotations for a different target task has wide applicability beyond entity tagging. Furthermore, this approach is also easy to implement even if we only have API-based access to the cross-task model and not the model parameters, which opens up the possibility of using pretrained open-source models.

2 PROBLEM STATEMENT

Entity detection is typically posed a sequence labeling problem, where every sentence consists of a sequence of tokens, and the objective is to label the tokens into predefined classes corresponding to various entity types [7] [16], such as brands, products, usecases, etc., in e-commerce domain. Unlike in typical supervised learning setting where one only have access to a set of labeled sequences for training, in our problem setting, we also have access to a frozen cross-task encoder model and also a large amount of in-domain unlabeled text sequences.

Formally, let C denote the set of entity types of interest. Using the popular begin-inside-outside (BIO) scheme [27], we obtain an extended label set C^{BIO} that includes an "Other" [O] label in addition to two labels for each entity type corresponding to whether a token is at start or a later position in an entity of that type. Let D^L denote the set of labeled sequences $\{(X_i^L, Y_i^L)\}_{i=1}^{N_L}$ where each text sequence X_i^L consists of a sequence of tokens and label sequence $Y_i^L (\in C^{BIO})$ consists of labels based on entity types. Let D^W denote the set of unlabeled text sequences $\{(X_h^W)\}_{h=1}^{N_W}$ where X_h^W consists of a sequence of tokens. Let g denote a frozen cross-task encoder model from the same domain that can take any input text X and map to a n-dimensional embedding vector, that encode its semantics i.e., $g(X) = Z$ where $Z \in R^n$.

Given a small amount of labeled data D^L, a large corpus of unlabeled data D^W and the cross-task model $g(\cdot)$, our objective is to learn a ER model $f(X, \theta)$, where θ is a parameter, that can accurately predict the probability of entity labels of tokens in an unseen text sequence X. Concretely, if $p(X)$ is likely probability of occurrence of X and Y is the true label sequence corresponding to X, we seek to estimate θ that minimises the prediction loss, i.e.,

$$\hat{\theta} = argmin_\theta \sum_X p(X) l(Y, f(X, \theta)) \tag{1}$$

where $l(.,.)$ is a suitable loss function such as the cross-entropy loss for token-wise classification model or a negative likelihood loss for Conditional Random Field (CRF) model [22].

3 RELATED WORK

Like other fields of NLP, approaches to NER have revolved around traditional methods like dictionary and rule based [13], to techniques based on feature engineering [44], to traditional deep learning schemes [29] [21] (requiring huge amount of labelled training data) to the current crop of methods revolving around transformers [10]. The deep learning era has typically seen BiLSTM and BERT CRFs being extensively used for NER task [23] [8]. As mentioned in the introduction, labelled data, especially for new domains is more scarce for ER as compared to other NLP tasks, as this involves multiple (entity) annotations for each given sentence, hence increasing the effort and ambiguity of labelling. Thus a fully supervised learning with no pre-training is not suitable, especially for NER, as for other NLP tasks.

In domains with little or very little data, few or zero shot learning techniques [46] [15] are commonly employed, which include language models (with probably large set of parameters) [10]. The general approach is to pre-train large models with data from multiple contrasting domains with masked language modelling and/or early domain adaptation few shot learning. Few shot learning techniques have generally revolved around prototypical networks or meta-learning, and prompting [46] [26] [9]. In prototypical networks [39], each class has a prototype embedding for each entity class in the word token embedding space. New tokens are tagged based on its embedding proximity to one of the tag prototypes. Meta-learning [25] is a similar technique where the tag class of a token is predicted based on the similarity of its embedding with that of a support set of tokens in an embedding space. Prompt based

[2]A cross-task frozen/legacy model would be trained on data from the same domain but on a different task. Note that this is not a multi-task learning set up since the model is already trained for a different task and the model parameters are not being fine-tuned.

techniques [6] have also given promising directions for few shot domain adaptation. The other set of techniques and that which we are interested in is weak labelling schemes [30] [38]. Most of the weak labelling techniques will have a small set of strongly labelled data [14] [48] which will be used as a high impact learning set with weak labels aiding to learn depending on the extend of noise in the weak labels. The generation of weak labels are either done from domain knowledge bases [37] or using weakly trained models [5]. [5] uses a classification module and a sequence labelling module. They pre-train the classification module to capture the textual context semantics from massive noisy data, and then the sequence labeling module further fine-tunes a neural network using a Partial-CRFs layer. NEEDLE[48] again is another weak labelling technique, however their method suppresses the extensive noise in weakly labeled data employing a regularization for the weakly labelled data loss, thus learning from strongly and weakly labeled data more effectively.

However, the effectiveness of suppressing extensive noise in weakly labeled data, and learning from both strong and weakly labeled data has not yet been well investigated in ER. Different from previous approaches, our method leverages pre-trained encoder model from the same domain but different tasks to create ontologies that produce less noisy weakly labeled data and then learn from both strong and weakly labeled data using a regularization loss.

4 PROPOSED METHODOLOGY

4.1 Stage I (Ontology Creation):

The first stage of WeSDoM involves developing automated domain ontologies from unlabeled data by leveraging cross-task legacy encoder from the same domain. We use a phrase extractor to generate n-gram key phrases (entities) from large unlabeled data followed by legacy encoders to create n-dimensional semantic representations, enabling clustering of domain phrases using hierarchical clustering methods.

• **Legacy Cross-task Encoder** There are many domains where pre-trained in-domain BERT-based encoders are openly available, such as BioBERT [24] for biomedical corpora, FinBERT [2] for financial services corpora, SciBERT [3] for science literature etc. Our proposed method can directly use these models as legacy encoder models in their specific domains. For our internal usecase (e-commerce), we fine-tune pre-trained BERT model on a human-audited query-product relevance dataset in siamese fashion so that query model $g(\cdot)$ can capture phrases semantically in a n-dimensional embedding space. The Figure 1 shows the architecture of our legacy encoder BERT. Inspired from [1], we construct the ranking loss (described in eq 2) based on the cosine score and the human-audited relevance label, where human-annotated label belongs to one of the three classes (i) strict relevance (ii) standard relevance (iii) irrelevant. We construct our ranking loss to take advantage of hard labels' ordinal nature. The relevance gradation ensures strictly relevant products are prioritized over standard relevant products when available. Particularly, human-audited datasets are noise-free and are capable of building robust models. Considering query model can be used to convert any query/phrase into n-dimensional semantic

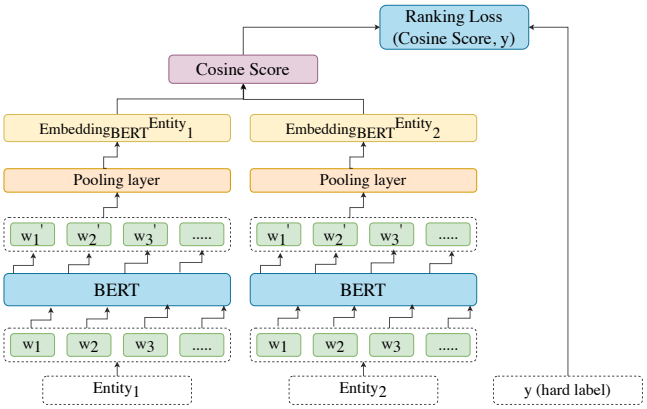

Figure 1: Legacy Encoder BERT

representations [1], we built a siamese model in our use-case.

$$L_{\text{RL}} = \sum_{(e_i^1, e_i^2, y_i) \in D_{label}} (1_{y_i = strict}(\hat{y_i} - 1)^2 + 1_{y_i = standard}((min(0, \hat{y_i} - \theta_{smin}))^2 + (max(0, \hat{y_i} - \theta_{smax}))^2) + 1_{y_i = irrelevant}(min(\hat{y_i}, 0))^2) \quad (2)$$

where θ_{smin} and θ_{smax} are hyper-parameters. $\hat{y_i}$ is the cosine score computed between two entities (query and product title).

• **Phrase Extraction** The use of embedding-based phrase (or entities) extraction methods [42] [31] has recently gained traction, with KeyBERT proposed by [18] being the most recent state-of-the-art method, which leverages pretrained BERT based embeddings for phrase extraction. In order to extract key phrases (i.e., NER entities) from unlabeled data, we adopt the KeyBERT method with n-grams ranging from 1 to 5. Following this, we use our trained legacy query encoder and create an embedding space using a pool of distinct in-domain NER entities.

• **Hierarchical clustering** In order to form domain ontologies, we adopt DBSCAN [12] clustering algorithm. In DBSCAN, two parameters are required: ϵ (eps) and the minimum number of points required to form a dense region (minPts). We tune these two parameters (ϵ=0.25, minPts=3) to ensure we form small clusters with low semantic variance between entities within clusters, and do not consider singleton clusters that formed during this process to avoid labeling bias errors. Furthermore, to remove phrases that might not be proper entities and are generated from the phrase extraction step, we analyze the frequencies (<10) in unlabeled data and eliminate them from our ontologies.

4.2 Stage II (Weak Supervision):

Stage II involves annotation of the large unlabeled data for the target NER task based on the ontologies (i.e., clusters of domain phrases) created in stage I. We assign each cluster with the most common entity type (pre-defined based on NER task) based on training data entities. This assignment is carried out by mapping each entity within the cluster to its three nearest neighbors present in strongly labeled data and the mapping is achieved using embeddings generated by the domain encoder model. The most common

Table 1: Examples of semantically similar clustered domain ontologies. The entity types are generic and not indicative of a production setting.

Clusters	Entity Type
i3 10th, 10th gen, i5 10th, i7 10th	value-&-units
television, tv, smart tv	Product-Category
17 inch, 15 inch, 16 inch, 14 inch	value-&-units
windows10, windows7, windows8	Feature-name
desktop, computer, desktop computer	Product-Category
python, java, python programming, coding	Use-case

entity type for that cluster is then determined by considering all the closest neighboring entities. We use these assigned clusters to label each sentence in the unlabeled in-domain data by direct matching (following BIO schema) to generate weakly labeled data. We consider only those samples in our weakly labeled dataset that have at least one entity label. Table 1 shows some examples of domain ontologies and tagged entity types derived from this method.

4.3 Stage III (Model Training):

Building a NER model involves multiple training phases -

• **In-domain MLM** In transformer-based models, a popular method for leveraging large unlabeled data is unsupervised pre-training using masked language modelling (MLM) [41]. Inspired by [24], we use an open-domain pretrained BERT and perform in-domain continual masked language model pre-training on large unlabeled in-domain dataset. As a result of MLM pre-training the BERT contains encoder parameters ($\theta_{encoder}$) and classification head parameters (θ_{cls}), where we use encoder parameters $\theta_{encoder}$ in the next stage.

• **Supervised Pre-training with CRF Layer** In the next phase of training, we replace the MLM head θ_{cls} with CRF classification head θ_{crf}, and then perform supervised continual pre-training with strongly labeled data $\{(X_i^L, Y_i^L)\}_{i=1}^{N_L}$. In this phase, $\theta_{encoder}$ are initialized from the previous phase, while θ_{crf} parameters are generated at random before training begins.

• **Pre-training with Noise-aware Loss Function** Stage II involves generating weak labels for a large unlabeled dataset. Furthermore, inspired from [48], we generate a complete weakly labeled dataset $\{(X_h^W, Y_h^W)\}_{h=1}^{N_W}$ by labeling[3] only non-entities in stage II using BERT-CRF model. In a model where the loss is directly applied to weakly labeled data, the model tends to overfit the noise of weak labels. Inspired from [48], we adopt a loss function that is noise-aware based on the estimated confidence of the assigned weak labels (Y^W), where estimated confidence is defined as the estimated probability of Y^W being the true label Y: $\hat{P}(Y^W = Y|X^W)$. Section 4.3.1 provides more details on confidence estimation.

When the confidence is lower/higher, the noise-aware loss function makes the fitting to weak labels more conservative/aggressive. When $Y^W = Y$, we consider loss function to be the, $L(.|Y^W = Y) = l(.;.)$[4]; when $Y^W \neq Y$, we consider loss function to be the $L(.|Y^W \neq Y) = l^-(.;.)$[5]. Accordingly, we define the noise aware loss function as follows:

$$\begin{aligned} l_{NA}(Y^W, f(X^W;\theta)) = \hat{P}(Y^W = Y|X^W) l(Y^W, f(X^W;\theta)) \\ +\hat{P}(Y^W \neq Y|X^W) l^-(Y^W, f(X^W;\theta)) \end{aligned} \tag{3}$$

For both strongly and weakly labeled data, the training objective is defined as follows:

$$\begin{aligned} \hat{\theta} = argmin_\theta \sum_{X^L} p(X^L) l(Y^L, f(X^L;\theta)) + \\ \gamma \sum_{X^W} p(X^W) l_{NA}(Y^W f(X^W;\theta)) \end{aligned} \tag{4}$$

• **Final Fine-Tuning (FT)** Training phases prior to this focus mainly on preventing the model from overfitting to the noise of weak labels. Meanwhile, they suppress the fitting of the model to the strongly labeled data. We propose fine-tuning the model again on strongly labeled data to address this issue. The results of our experiments indicate that such fine-tuning is necessary.

4.3.1 *Confidence Estimation of Weak Labels.*

In this section, we explain how we estimate weak labels' confidence $\hat{P}(Y^W = Y|X^W)$. Please note that, Y^W in WeSDoM is composed of weak labels from ontology creation Y^O as well as model predictions Y^P. Inspired from [48], we can estimate the confidence in weak labels using the confidence in these two parts as follows:

$$\begin{aligned} \hat{P}(Y^W = Y|X^W) = \frac{\#MatchedTokens}{\#TotalTokens} \hat{P}(Y^O = Y|X^W) \\ +(1 - \frac{\#MatchedTokens}{\#TotalTokens}) \hat{P}(Y^P = Y|X^W) \end{aligned} \tag{5}$$

Our linear combination uses this weighting because we use all matched tokens in weakly-supervised data derived from ontologies, while in other tokens we use model predictions.

Following [48], we assume the ontologies creation and matching process are less ambiguous, and consider the confidence in weak labels $\hat{P}(Y^O = Y|X^W) = 1$. The confidence of $\hat{P}(Y^P = Y|X^W)$ can be estimated using the CRF score and histogram binning [47]. More specifically, we categorize samples into multiple bins based on their BERT-CRF score. Then, we take into account a validation dataset separate from the final evaluation dataset and estimate the confidence of each bin based on the predicted sample. Finally, for a new sample, we first calculate the BERT-CRF score, and estimate the prediction confidence by the confidence of the corresponding bin in the histogram.

5 EMPIRICAL EVALUATION

We present our findings on the benefits of utilizing cross-task domain models via weak supervision for entity recognition focusing on the three research questions listed in introduction. We begin with a discussion of the datasets and the experimental setup.

[3] predictions from the BERT-CRF model

[4] $l(Y, f(X;\theta)) = -logP_{f(X;\theta)}(Y)$

[5] $l^-(Y, f(X;\theta)) = -log(1 - P_{f(X;\theta)}(Y))$

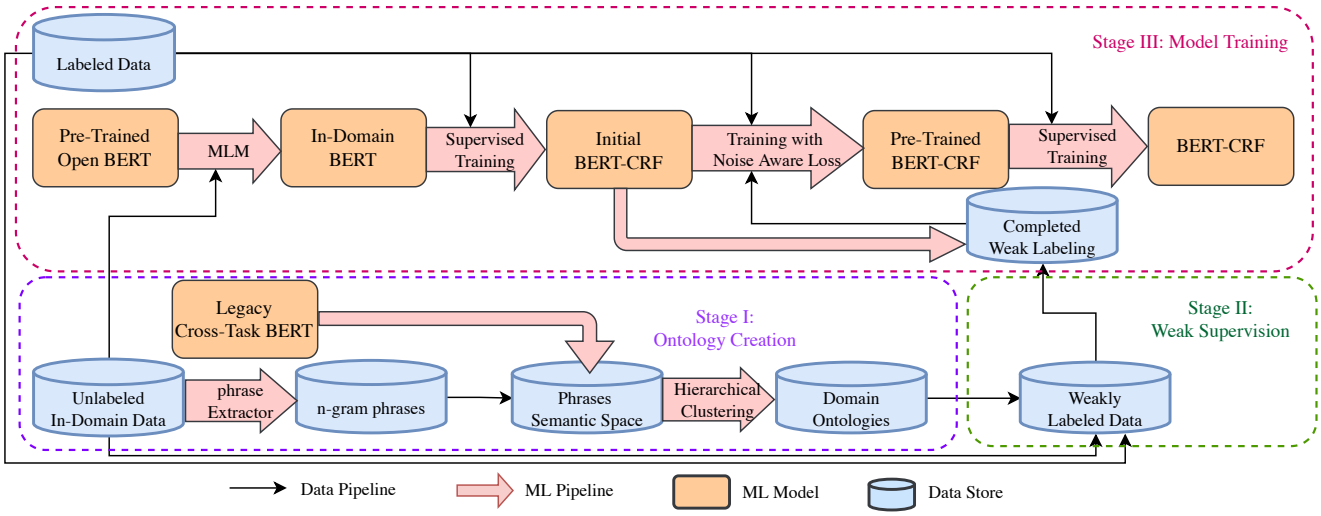

Figure 2: Three Stage WeSDoM Framework.

5.1 Experimental Setup

Dataset Generation For ***e-commerce*** query NER task, all data used in our analysis is anonymized, aggregated, and does not reflect production distribution. Our training dataset consists of 10K strongly labeled samples and 250K weakly labeled samples derived from an in-domain unlabeled dataset. Unlabeled in-domain data is obtained from Amazon IN marketplace by aggregating anonymous user behavior data. Additionally, we created validation and testing datasets that each contained 5K samples, which were not part of the training dataset. Our NER task includes eight types of entities predefined based on search queries. In case of ***biomedical***, we use two popular datasets, BC5CDR-Chem and BC5CDR-Disease [45], which consists of single entity types, Chemical and Disease, respectively. Similar to [48], we collected unlabeled data from the PubMed 2019 baseline[6] and used it to create domain ontologies. As a legacy model, we use the publicly available PubMedBERT model [19], which has been pre-trained on medical in-domain data. To compare the results, the test dataset is used as a benchmark.

Experimental Details We use bert-base-uncased EN model as a base model to build a NER model. All the experiments are conducted using the BIO tagging scheme. As part of stage III, we do the MLM training for 5 epochs, masking 15% of the words randomly. Then, we train the BERT-CRF NER model for 5 epochs followed by 5 epochs with noise aware loss functions. Finally, we train the model again on strongly labeled data for 5 more epochs. We use a batch size of 512 and ADAM optimizer with a learning rate of $1^{*}10^{-5}$ for e-commerce NER model. The experiments were conducted on an AWS p2.8xlarge EC2 instance with only one GPU. All of the hyper parameters that were used in this study were chosen empirically based on the results of the experiments conducted. We use PubMedBERT as a base model for entity recognition on Biomedical dataset since it's already pre-trained in the same domain, so MLM training is not needed here. Rest of the setup is similar to e-commerce NER. We repeated the experiments five times using different random seeds to test the statistical significance of the results.

Algorithm Baselines In this paper, we compare our proposed method with the following baselines. For fair comparison, all models used in the baseline methods have been continually pre-trained on in-domain unlabeled data.

1. NEEDLE [48] : Author proposes a novel computational framework which effectively suppresses the extensive noise in weak labeled data, and learns from both strongly and weakly labeled data.

2. Supervised Learning Baseline: We directly finetune BERT-CRF model on strongly labeled data.

3. Semi-supervised Self-Training (SST)[11]: SST generates pseudo labels for unlabeled data based on the model learned through supervised learning and then conduct semi-supervised learning.

4. Weakly Supervised Learning (WSL) [32]: Using WSL, the author simply combines strongly labeled data with weakly labeled data and then trains the model.

5. Weighted WSL: WSL with weighted loss, in which weakly and strongly labeled samples are combined based on the weighted loss function. We tune the weight parameter γ for best results. (see equation 6)

6. Robust WSL [17]: WSL with mean squared error loss, which is robust against label noise.

7. Partial WSL[37]: In WSL, the training loss is excluded for non-entity weak labels.

Metrics. As evaluation metrics, we use span-level precision/recall/F1. Span-level metrics refers to the precision calculated at the level of entire entity spans rather than individual tokens.

$$\hat{\theta} = argmin_{\theta} \frac{1}{M+\tilde{M}} [\sum_{m}^{M} l(Y_m, f(X_m;\theta)) + \gamma \sum_{\tilde{m}}^{\tilde{M}} l(Y_{\tilde{m}}, f(\tilde{X}_m;\theta))] \quad (6)$$

[6]Titles and abstracts of Biomedical articles: https://ftp.ncbi.nlm.nih.gov/pubmed/baseline/

Table 2: Results on E-commerce Dataset: Mean & std. error for F1 are reported based on 5 trials runs.

Method	Precision	Recall	F1
WeSDoM	**65.21**	**74.23**	**69.42**±0.07
NEEDLE	64.12	69.24	66.58±0.09
Supervised Baseline			
BERT-CRF	63.10	68.24	65.56±0.05
Semi-Supervised Baseline			
SST	63.25	68.89	65.95±0.09
Weakly-Supervised Baselines			
WSL	59.35	61.64	60.47±0.08
Weighted WSL	62.28	64.37	63.30±0.11
Partial WSL	58.45	66.26	62.11±0.05
Weighted Partial WSL	61.78	67.45	64.49±0.09
Robust WSL	55.34	49.32	52.16±0.06

5.2 Main Results

In Table 2, we present our proposed method results for the e-commerce query NER, which are compared to SOTA baselines.

• **WeSDoM** In our experiments, WeSDoM achieves the highest performance among all baseline methods, outperforming NEEDLE and the fully supervised baseline, by a significant margin.

• **Weakly Supervised Baselines** WSL, Weighted WSL, Partial WSL, and Robust WSL are all weakly supervised baseline methods that produce worse performance than the supervised baseline method. NEEDLE paper also shows that weakly labeled data can hamper the model's performance if not handled properly. Our results are consistent with their claim.

• **Semi-supervised self-training (SST)** It was found that SST outperforms the supervised baseline, indicating that pseudo labels generated from model prediction on unlabeled data have some inherited meaning. However, there are weakly supervised baselines that are lower than the supervised baseline, implying that weak labels are not handled effectively. WeSDoM, on the other hand, outperforms NEEDLE, SST, and supervised baselines, which suggests that weak labels can provide additional knowledge and enhance model performance when their noise is suppressed.

Biomedical Results: In Table 3, we summarize our results on two biomedical datasets, BC5CDR-Chem and BC5CDR-Disease. We don't include weekly supervised baselines here because they are in any case lower than supervised baseline. Among all comparison methods, WeSDoM performs the best. Our results on both BC5CDR datasets outperform previous SOTA PubMedBERT [19] and NEEDLE, resulting in new SOTA F1 scores of 94.27 & 91.23 on BC5CDR-Chem and BC5CDR-Disease, respectively.

5.3 Research Questions

5.3.1 ***RQ1: Effectiveness of Legacy Cross-task Model.*** Our legacy model is trained on a human audited query-product relevance dataset with over ~2.4 million samples which achieves an AUC of 0.954. As a way to examine legacy model efficacy, we weaken legacy model performance by reducing training samples, resulting in AUC drops. Our experiments demonstrate that if we drop training samples by 10%, 20%, and 50% prior to training our legacy model, the ROC-AUC of our legacy model drops to 0.939, 0.930, and 0.902 respectively, resulting in a NER model F1 score of 69.05, 68.60, and 68.02. As a result, we conclude that NER model performance improves when a high-performing legacy model is used.

Table 3: Biomedical Datasets Results: Span Level F1 Scores

Method	BC5CDR-Chem	BC5CDR-Disease
WeSDoM	**94.27**±0.05	**91.23**±0.09
NEEDLE	93.58±0.06	90.54±0.05
Supervised BERT-CRF	92.70±0.11	85.09±0.06
SST	93.04±0.08	85.38±0.04
Reported F1 scores in [19]		
BioBERT	92.85	84.70
PubMedBERT	93.33	85.62

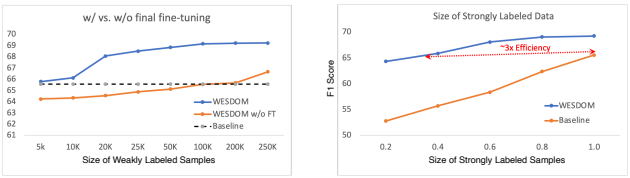

Figure 3: A comparison of the size of weakly labeled data and performance. We present WeSDoM performance (left fig) with and without final fine-tuning using E-commerce NER data. In the (right) figure, we see WeSDoM's efficiency on strongly labeled data. Supervised BERT-CRF trained on strongly labeled data is the baseline here.

Table 4: Assessment of WeSDoM's components' effectiveness

Method	Precision	Recall	F1
WeSDoM w/o MLM/OC/NALF	62.38	66.98	64.59
WeSDoM w/o MLM/NALF	61.78	66.24	63.93
WeSDoM w/o MLM/FT	62.14	64.53	63.31
WeSDoM w/o OC	63.10	68.24	65.56
WeSDoM w/o MLM	65.08	70.56	67.71
WeSDoM w/o NALF	64.56	72.56	68.32
WeSDoM w/o FT	60.23	67.42	63.62
WeSDoM	**65.21**	**74.23**	**69.42**

5.3.2 ***RQ2: Performance vs. size of weakly (or strongly) labeled data.*** In order to demonstrate that WeSDoM exploits weakly labeled data more effectively, we test the model with a randomly subsampled dataset of weakly labeled data. In Figure 3 (left fig), we plot the F1 curve for e-commerce NER with increasing size of weakly labeled data with (w/) and without (w/o) final fine-tuning

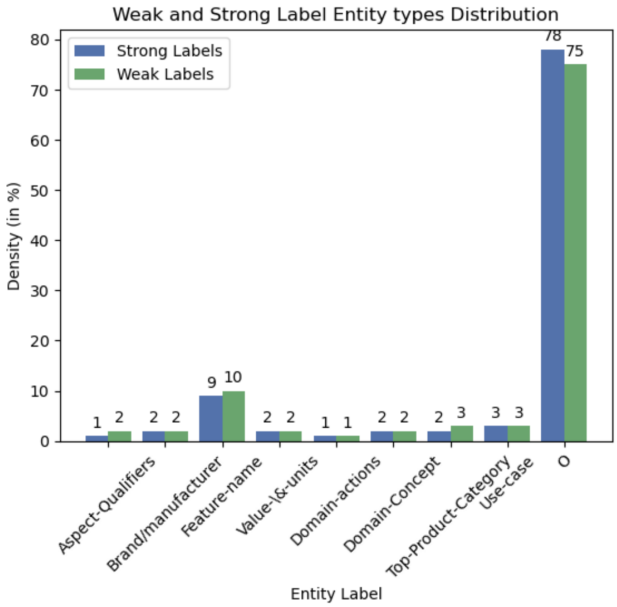

Figure 4: Analyzing Entity Distributions in Weak and Strong Labels. In this context, 'O' represents the non-entity type, indicating labels that do not belong to any entity type.

(FT). In both cases, the size of weakly labeled data is found to benefit WeSDoM more effectively. This demonstrates that WeSDoM is significantly more effective for suppressing noise as adding weakly labeled data tailored to improve performance irrespective of FT. Furthermore, we tested it on randomly subsampled strongly labeled data. To achieve the same performance as the (fully) supervised baseline, WeSDoM only requires 30% strongly labeled data as shown in (right) Figure 3.

5.3.3 ***RQ3: Effect of Each Component in WeSDoM.*** To determine how effective WeSDoM components are, we examine the evaluation metrics after one or more components are removed. In particular, we use the following abbreviation to identify each component of WeSDoM (i) MLM: Masked Language Modelling (ii) OC: Ontology Creation (Stage I) (iii) NALF: Pre-training with Noise-aware Loss Function (iv) FT: Final fine-tuning. It should be noted that NALF cannot be used without OC, and when NALF is absent, the model is trained on both strongly and weakly labeled datasets by minimizing loss in equation 1. Furthermore, when referring to WeSDoM w/o OC, it denotes the Supervised BERT-CRF baseline, as the absence of OC implies the lack of weakly labeled data. Table 4 shows that all components are beneficial, with OC and FT being the most useful.

5.4 Ablation Study

5.4.1 ***Quantify the Impact of Weak Labels.*** We examine the impact of weak labels on improving NER model performance. Similar to NEEDLE, we check the errors made by the WeSDoM model on the validation set. There are 1094 entities that have been incorrectly classified by the BERT-CRF model. The results of WeSDoM reveal that 390 of 1094 entities have been classified correctly. However, the model makes 241 more incorrect predictions. Note that not all entities in validation data are directly affected by the weakly labeled data, i.e., they are not observed in the weakly labeled data. By excluding the entities (from validation) that are not observed in the weakly annotated entities, we find 146 new correctly classified entities and 74 new incorrectly classified entities. With a ratio of 146/74 = 1.97 »1, whereas NEEDLE has a ratio of 1.84, WeSDoM has the advantage over NEEDLE.

5.4.2 ***Comparison of Weak and Strong Label Distribution:*** This section presents the label distribution between weak and strong labels, and demonstrates how WeSDoM can help close this gap. Specifically, Figure 4 compares the entity distributions of the true labels and weak labels. As can be seen, weak labels have marginally more entities than strong labels. However, the distribution of weak labels across entities is nearly the same as that of strong labels which explains why WeSDoM can directly improves the performance.

5.5 Deployment Considerations

Our current line of research on entity detection for a niche domain was motivated by the requirements of a conversational shopping assistant meant to aid customers in emerging markets during their pre-purchase shopping journey. For this pre-purchase shopping scenario, the entity types that need to be detected, e.g., feature units (e.g., MB, GB), sorting criteria (cheapest), are different from typical NER systems trained to detect named entities such as names, places, organisations, etc. Furthermore, by implementing our proposed model to replace human annotation in identifying entities within user queries and products at Amazon, we've conserved thousands of hours of human annotator bandwidth.

6 CONCLUSION

In NER problems, for new domains such as e-commerce, obtaining a large amount of human-annotated strongly labeled data is challenging due to token level labeling. In contrast to fully human-annotated labeled data, the majority of existing research relies on weakly labeled data together with limited human annotations. NEEDLE [48] proposes hybrid solutions to efficiently combine weakly labeled data with strongly labeled data to achieve superior results compared to supervised training. However, the large amount of noise in the weakly labeled data remains a significant obstacle. In this paper, we present a three-stage NER solution framework that utilizes cross-task legacy encoder models to effectively suppress the extensive noise in weakly labeled data and learn from both strongly and weakly labeled data. Our proposed approach for leveraging frozen specialized domain encoder models (e.g., FinBERT, BioBERT) along with unlabelled data to improve performance on predictive NLU tasks such as entity detection can be used for other applications.

REFERENCES

[1] MS Ankith, Sourab Mangrulkar, and Vivek Sembium. 2022. HISS: A novel hybrid inference architecture in embedding based product sourcing using knowledge distillation. (2022).

[2] Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063* (2019).

[3] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: Pretrained Language Model for Scientific Text. In *EMNLP*. arXiv:arXiv:1903.10676

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[5] Yixin Cao, Zikun Hu, Tat-seng Chua, Zhiyuan Liu, and Heng Ji. 2019. Low-resource name tagging learned with weakly labeled data. *arXiv preprint arXiv:1908.09659* (2019).

[6] Yanru Chen, Yanan Zheng, and Zhilin Yang. 2022. Prompt-Based Metric Learning for Few-Shot NER. *arXiv preprint arXiv:2211.04337* (2022).

[7] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of machine learning research* 12, ARTICLE (2011), 2493–2537.

[8] Savelie Cornegruta, Robert Bakewell, Samuel Withey, and Giovanni Montana. 2016. Modelling radiological language with bidirectional long short-term memory networks. *arXiv preprint arXiv:1609.08409* (2016).

[9] L Cui, Y Wu, J Liu, S Yang, and Y Zhang. [n. d.]. Template-based named entity recognition using BART. arXiv 2021. *arXiv preprint arXiv:2106.01760* ([n. d.]).

[10] 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).

[11] Jingfei Du, Edouard Grave, Beliz Gunel, Vishrav Chaudhary, Onur Celebi, Michael Auli, Ves Stoyanov, and Alexis Conneau. 2020. Self-training improves pre-training for natural language understanding. *arXiv preprint arXiv:2010.02194* (2020).

[12] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise.. In *kdd*, Vol. 96. 226–231.

[13] Dimitra Farmakiotou, Vangelis Karkaletsis, John Koutsias, George Sigletos, Constantine D Spyropoulos, and Panagiotis Stamatopoulos. 2000. Rule-based named entity recognition for Greek financial texts. In *Proceedings of the Workshop on Computational lexicography and Multimedia Dictionaries (COMLEX 2000)*. 75–78.

[14] Aleksander Ficek, Fangyu Liu, and Nigel Collier. 2022. How to tackle an emerging topic? Combining strong and weak labels for Covid news NER. *arXiv preprint arXiv:2209.15108* (2022).

[15] Alexander Fritzler, Varvara Logacheva, and Maksim Kretov. 2019. Few-shot classification in named entity recognition task. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*. 993–1000.

[16] Abbas Ghaddar and Philippe Langlais. 2018. Robust lexical features for improved neural network named-entity recognition. *arXiv preprint arXiv:1806.03489* (2018).

[17] Aritra Ghosh, Himanshu Kumar, and P Shanti Sastry. 2017. Robust loss functions under label noise for deep neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 31.

[18] Maarten Grootendorst. 2020. KeyBERT: Minimal keyword extraction with BERT. *Zenodo* (2020).

[19] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)* 3, 1 (2021), 1–23.

[20] Michael A Hedderich, Lukas Lange, and Dietrich Klakow. 2021. ANEA: distant supervision for low-resource named entity recognition. *arXiv preprint arXiv:2102.13129* (2021).

[21] Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *arXiv preprint arXiv:1508.01991* (2015).

[22] John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. (2001).

[23] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360* (2016).

[24] 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.

[25] Jing Li, Billy Chiu, Shanshan Feng, and Hao Wang. 2020. Few-shot named entity recognition via meta-learning. *IEEE Transactions on Knowledge and Data Engineering* 34, 9 (2020), 4245–4256.

[26] Jing Li, Shuo Shang, and Ling Shao. 2020. Metaner: Named entity recognition with meta-learning. In *Proceedings of The Web Conference 2020*. 429–440.

[27] Qi Li, Haibo Li, Heng Ji, Wen Wang, Jing Zheng, and Fei Huang. 2012. Joint bilingual name tagging for parallel corpora. In *Proceedings of the 21st ACM international conference on Information and knowledge management*. 1727–1731.

[28] Yinghao Li, Pranav Shetty, Lucas Liu, Chao Zhang, and Le Song. 2021. BERTifying the hidden Markov model for multi-source weakly supervised named entity recognition. *arXiv preprint arXiv:2105.12848* (2021).

[29] Nut Limsopatham and Nigel Henry Collier. 2016. Bidirectional LSTM for named entity recognition in Twitter messages. (2016).

[30] Pierre Lison, Aliaksandr Hubin, Jeremy Barnes, and Samia Touileb. 2020. Named entity recognition without labelled data: A weak supervision approach. *arXiv preprint arXiv:2004.14723* (2020).

[31] Debanjan Mahata, John Kuriakose, Rajiv Shah, and Roger Zimmermann. 2018. Key2vec: Automatic ranked keyphrase extraction from scientific articles using phrase embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. 634–639.

[32] Gideon S Mann and Andrew McCallum. 2010. Generalized Expectation Criteria for Semi-Supervised Learning with Weakly Labeled Data. *Journal of machine learning research* 11, 2 (2010).

[33] Diego Mollá, Menno Van Zaanen, and Daniel Smith. 2006. Named entity recognition for question answering. In *Proceedings of the Australasian language technology workshop 2006*. 51–58.

[34] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* 21, 140 (2020), 1–67.

[35] Jonathan Raiman and Olivier Raiman. 2018. Deeptype: multilingual entity linking by neural type system evolution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[36] Erik F Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050* (2003).

[37] Jingbo Shang, Liyuan Liu, Xiang Ren, Xiaotao Gu, Teng Ren, and Jiawei Han. 2018. Learning named entity tagger using domain-specific dictionary. *arXiv preprint arXiv:1809.03599* (2018).

[38] Edwin Simpson, Jonas Pfeiffer, and Iryna Gurevych. 2020. Low resource sequence tagging with weak labels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 8862–8869.

[39] Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. *Advances in neural information processing systems* 30 (2017).

[40] Amber Stubbs, Christopher Kotfila, and Özlem Uzuner. 2015. Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/UTHealth shared task Track 1. *Journal of biomedical informatics* 58 (2015), S11–S19.

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[42] Rui Wang, Wei Liu, and Chris McDonald. 2014. Corpus-independent generic keyphrase extraction using word embedding vectors. In *Software engineering research conference*, Vol. 39. 1–8.

[43] Zihan Wang, Jingbo Shang, Liyuan Liu, Lihao Lu, Jiacheng Liu, and Jiawei Han. 2019. Crossweigh: Training named entity tagger from imperfect annotations. *arXiv preprint arXiv:1909.01441* (2019).

[44] Rebecka Weegar, Arantza Casillas, Arantza Díaz de Ilarraza, Maite Oronoz, Alicia Pérez, and Koldo Gojenola. 2016. The impact of simple feature engineering in multilingual medical NER. In *Proceedings of the Clinical Natural Language Processing Workshop (ClinicalNLP)*. 1–6.

[45] Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Jiao Li, Thomas C Wiegers, and Zhiyong Lu. 2016. Assessing the state of the art in biomedical relation extraction: overview of the BioCreative V chemical-disease relation (CDR) task. *Database* 2016 (2016).

[46] Yi Yang and Arzoo Katiyar. 2020. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. *arXiv preprint arXiv:2010.02405* (2020).

[47] Bianca Zadrozny and Charles Elkan. 2001. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Icml*, Vol. 1. 609–616.

[48] Taolin Zhang, Chengyu Wang, Nan Hu, Minghui Qiu, Chengguang Tang, Xiaofeng He, and Jun Huang. 2022. Dkplm: Decomposable knowledge-enhanced pre-trained language model for natural language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 11703–11711.