Multi-Task Student Teacher based Unsupervised Domain Adaptation for Address Parsing

Rishav Sahay, Anoop Saladi, and Prateek Sircar

Amazon.com, Bengaluru, Karnataka, India {rissahay,saladias,sircarp}@amazom

Abstract. In an e-commerce business, the ability to parse postal addresses into sub-component entities (such as building, locality) is essential to take automated actions at scale for successful delivery of shipments. The entities can be leveraged to build applications for logistics related operations, e.g. geocoding, assessing address completeness. Training an accurate address parser requires a significant number of manually labeled examples which is very expensive to create, especially when trying to build model(s) for multiple countries with unique address structure. To tackle this problem, in this paper, we present a novel Unsupervised Domain Adaptation (UDA) framework to transfer knowledge acquired by training a parser on labeled data from one country (source domain) to another (target domain) with unlabeled data. We specifically propose a multi-task student-teacher model comprising of three components: 1) specialized teachers trained on source data to create a pseudo labeled dataset, 2) consistency regularization, that uses a new data augmentation technique for sequence tagging data, and 3) boundary detection, leveraging signals in addresses like commas and text box boundaries. Multiple experiments on diverse address datasets ¹ demonstrate that our approach outperforms state-of-the-art UDA baselines for Named Entity Recognition (NER) task in terms of F1-score by 2-9%.

Keywords: Named Entity Recognition \cdot Address parsing \cdot Unsupervised Domain Adaptation

1 Introduction

Address is the most critical customer data to make successful and reliable deliveries of products in an e-commerce business. Some of the challenges we observe with respect to address quality such as misspellings, non-standard address connotations lead to delivery delays/failures, adversely impacting crucial business metrics. These problems are more pertinent to emerging countries where addresses are not standardized. Address parsing helps in identification of unique sub-components which in turn can be leveraged for multiple downstream tasks,

¹ In this paper, we do not reveal the name of the e-commerce countries on which we evaluate our models due to business confidentiality. We also mask finer address details with (XX) to preserve customer's privacy.

such as assessing completeness of addresses, predicting geocodes from address text, identifying high density communities in cities to pilot new services. For example, if you consider an address karkarbagh colony, near sbi bank, the parser can signal a missing building entity which can be used to power customer nudges to request additional information during address creation/order placement, thereby improving address quality. Address parsers are essentially NER models that require huge amounts (>50K) of labeled addresses with tags such as building name. locality, road and house unit information for model training. While address reference data from open-sources like OpenAddresses [6] and OpenStreetMap [7] are available, they differ vastly from the noisy/unstructured addresses commonly entered by customers and hence, lead to dismal performance in a practical setting. On the other hand, obtaining manually annotated address label data is expensive, time-consuming and prone to human errors. To tackle the labeled data scarcity issue, we leverage annotated data available for existing source countries when training an address parser for a new target country. There are various challenges involved while transferring knowledge from one country to another: 1) different countries have different address structures. For example, the most common address writing format in India (IN) is building name, road name followed by locality, while in the case of United Arab Emirates (UAE), it is road name followed by building name, and locality, 2) different countries may have addresses in different languages. Even if they are written in the same language (as in our experiments fixed to English), they have a very low vocabulary overlap. Hence, to solve these problems, we propose a novel multi-task student teacher based UDA architecture for address parsing that uses labeled data from source country to learn a target country address parser. Our approach involves three steps. Firstly, we perform domain adaptive pre-training for the base teacher model using both source and target addresses. We then train multiple teachers using source labeled data for address parsing to help deal with structural differences between countries. Finally, the student model is trained on two student-teacher tasks, consistency regularization and entity boundary detection. While consistency regularization task helps in making the student model robust to noise, boundary detection task provides additional information to the student about the target address structure, thereby improving overall model performance. To summarize, we make the following contributions in this work:

- We propose a novel multi-task student-teacher based UDA framework for address parsing that uses two teachers - one teacher is learned directly on source data while the other uses shuffled data.
- We present a new data augmentation technique applicable to sequence tagging data like addresses to perform consistency regularization on the student model.
- We introduce boundary detection as an additional task that leverages selfsupervised signals in addresses like customer commas and address text box boundaries.
- We evaluate our approach on proprietary e-commerce data and external datasets, and demonstrate its superiority over state-of-the-art baselines. Ab-

lation study further confirms the effectiveness of each of the proposed components.

2 Related Work

Address Parsing: In [11], the authors develop a multinational address parser using subword embeddings and recurrent neural network architecture. They build a single model capable of parsing addresses from multiple countries at the same time. Specifically, they use MultiBPEmb [4] to vectorize each word and the subword embeddings are fed into a BiLSTM encoder. The last hidden state of each word is fed to a feed-forward layer that is fed as input to a Seq2Seq module. [12] improves upon the above multi-national parser by adding an attention mechanism while label decoding, along with domain adversarial training for domain invariant features. Although the models are meant for multi-national parsing, the addresses on which they are trained with are structured addresses [10] that lack real life noise provided by customers for e-commerce delivery. Further, training a single parser model for multiple countries leads to deteriorated results for emerging countries, given the differences in address formats between countries. Also, since we do not have access to annotated data from multiple countries (~ 20 as assumed in the paper), we focus on single-source single-target address parsing in our experiments.

UDA-NER: As shown in [3], domain adaptive fine tuning using Masked language Modelling (MLM) on the target data and source data before fine tuning on the source NER data can help in boosting the zero shot performance on the target data (with a different distribution as compared to the source). In [9], the authors propose a teacher-student learning method for cross-lingual NER where the student model is trained using mean squared error loss with the teacher's output probability distribution (soft pseudo labels) as the ground truth. They also extend the methodology to multi-source cross lingual NER by weighing each of the source teachers' soft labels using the similarity between the source and target language vectors. Similarly, in [1], the authors use a student-teacher framework for cross-lingual NER, where a teacher is trained on source labeled data and, in parallel, a language discriminator and encoder are trained on the token-level adversarial task. All the above works on UDA/cross-lingual NER deal with open domain text. Addresses, on the other hand, are quite different as compared to open-domain text as they have a unique linguistic, and possess a notion of structure that can vary across countries. Hence, directly applying existing methods for our use-case is not optimal which is further validated in the results section.

3 Proposed Methodology

The core architecture of an address parser is that of a name entity recognition model, which is a token wise classifier on the top of an encoder $f(\theta)$ (RoBERTabase [5]). For a given address text $x = \{x_i\}_{i=1}^{L}$, the encoder maps it into a set of

hidden state vectors $h = \{f_{\theta}(x_i)\}_{i=1}^{L}$. For a token $x_i \in x$, its hidden state vector h_i is used to derive the probability distribution over the entity labels (see Section 4.1) using a linear classification layer and *softmax* function. We have access to a source labeled data D_{src}^l in which each source address has been assigned a label sequence, unlabeled source D_{src}^{ul} and unlabeled target D_{tgt}^{ul} addresses. We assume that source and target use the same set of entity labels.

3.1 Adaptive Pre-training using MLM

We first adapt an open source RoBERTa-base ² model to the addresses in source and target using MLM task [2]. We take 20K unlabeled addresses from both D_{src}^{ul} and D_{tgt}^{ul} after which we concatenate-shuffle them. The training procedure used here follows the same masking criteria and training settings as done in [3]. The adapted model $f(\theta_{add})$ helps to understand the linguistics within addresses and the vocabulary of both source and target data.

3.2 Student-Teacher Framework

Post adaptive pre-training, we use a student-teacher framework for unsupervised domain adaptation.

Multi-Teacher Training As discussed earlier, source and target can have very different structures. For e.g. in UAE, road occurs at the start for most of the addresses, while in India building occurs at the start. Training a teacher directly on source data will force the model to memorize such source-specific address structure patterns and produce noisy outputs on target data. This leads to a poor quality student model. To tackle this issue, we introduce an additional teacher trained on an entity level shuffled source data D_{shuf}^l . To create D_{shuf}^l , we randomly pick up annotated entity ent at index i from a source address with tags $\{x_{src}, y_{src}\} \in D^l_{src}$ and place it at another random position $j \ (j \neq i)$ to obtain a new shuffled address with tags $\{x_{shuf}, y_{shuf}\}$. Since the entities are shuffled, the source addresses will have higher variance in terms of structure, thereby enforcing the teacher model to pay attention to an entity itself without getting affected by its neighbours. Thus, we train two teacher models, main teacher $f(\theta_T)$ and shuffled-data teacher $f(\theta_{T_{shuf}})$ on D_{src}^l and D_{shuf}^l respectively using a cross-entropy loss function. The two teacher models during training are initialized using the weights of $f(\theta_{add})$ and the word embeddings layer is frozen to ensure that the models do not forget the target information that was learnt during MLM training.

Student-Teacher tasks Using each of the two teachers, for a given address x' where $x' \in D_{tgt}^{ul}$, we obtain two sets of pseudo labels for *i*-token x'_i : output probability distribution of the entity labels $P(x'_i; \theta_T)$ from the main teacher

² https://huggingface.co/docs/transformers/model_doc/roberta

and $P(x'_i; \theta_{T_{shuf}})$ from the shuffled-data teacher. The student-teacher loss is formulated as the mean squared error (MSE) between the output distributions of the entity labels by the student model $f(\theta_S)$ and pseudo labels generated by the teacher as done in [9]. For each of the losses, those tokens are only considered on which the corresponding teacher has a maximum output probability more than a certain threshold t (set to 0.85). The shuffled-data teacher is a structure agnostic model whose pseudo labels when combined with the main teacher's pseudo labels enhance the quality of the student model.

Algorithm 1: Data Augmentation for CR
Input: Unlabeled target data D_{tgt}^{ul}
1 Main teacher model $f(\theta_T)$
Output: Augmented Target data D_{aug}^{ul}
2 $EntDic \leftarrow \{\}$
3 $Y_{HL} \leftarrow []$
4 for x' in D_{tgt}^{ul} do
5 $y_{hl}, y_{probs} \leftarrow \text{get hard labels with max prob. with } f(\theta_T)$
$\textbf{6} \qquad types, starts, ends, probs \leftarrow$
find all entities with avg probs from y_{hard} and y_{probs}
7 for $ind \leftarrow 0$ to $len(types)$ do
s if $probs[ind] \ge 0.90$ then
9 $EntDic[types[ind]].append(x'[starts[ind]:ends[ind])$
10 $Y_{HL}.append(y_{hl})$
11 $D_{aug}^{ul} \leftarrow []$
12 for (x', y_{hl}) in (D^{ul}_{tgt}, Y_{HL}) do
13 $types, starts, ends \leftarrow$
find type, start ind, end ind of all entities from y_{hl}
14 for $ind \leftarrow 0$ to $len(types)$ do
15 $type \leftarrow types[ind]$
16 $randEnt \leftarrow \phi$
17 while $(end[ind] - start[ind]) \neq len(randEnt)$ do
18 $randEnt \leftarrow sample an entity from EntDic[type]$
19 $x' \leftarrow$ replace the entity at <i>ind</i> by <i>randEnt</i>
20 $x'_{aug} \leftarrow x'$
21 $D_{aug}^{ul}.append(x'_{aug})$

3.3 Consistency Regularisation task

Consistency regularization (CR) [8] is a well-studied technique used in semisupervised and self-supervised settings that encourages the prediction of the network to be similar in the vicinity of the observed training examples. We leverage this technique to make the student model more robust to noisy pseudo

labels predicted by the main teacher $f(\theta_T)$. Here, we introduce a new data augmentation technique which allows us to create synthetic target addresses with the same labels as pseudo labels produced by $f(\theta_T)$ for the original target address. Basically, we first create a dictionary with all the high confidence (> 0.90) entities predicted by $f(\theta_T)$ on D_{tgt}^{ul} . The dictionary contains an entity type mapped to list of confident predicted entities. The confidence of an entity is measured by the average maximum probability of the first sub-word of each of the entity tokens. Then for each pseudo-labeled target address $x' \in D_{tat}^{ul}$, we replace every entity within the address with another random entity of the same entity type (also same length to ensure label consistency) in the dictionary to get x'_{aug} . See algorithm 1 for the pseudo code to perform the data augmentation. The loss between the probability distribution of i token of x' denoted as $P(x'_i; \theta_S)$ and i token of x'_{aug} denoted as $P(x'_{iaug}; \theta_S)$ is formulated as a MSE loss. The synthetic address provides an entity level viewpoint to the student model. In other words, if a particular entity occurs at a differently in another address, it still refers to the same entity. The loss function for the above task is given by

$$L_{CR} = \sum_{x', x'_{aug} \in D^{ul}_{tgt}} \sum_{i=1}^{L} MSE(P(x'_i; \theta_S), P(x'_{iaug}; \theta_S))$$
(1)

3.4 Boundary Detection task

Address inputs have self-supervised signals like commas provided by customers while entering address text. In our case, we also have access to data in separate text fields (line 1, line 2 etc.) as entered during address creation, which automatically provides logical boundaries within the address. The key motivation behind introducing this module is that such implicit signals separating entities within addresses can potentially help the student model to identify correct entity spans in the target domain. Specifically, we sample an equal amount of addresses with boundary signals from the target address database as that of D_{tgt}^{ul} . Since these boundary signals can be noisy because of insufficient commas entered by customers, we only consider those addresses with more than two commas. We refer to this data as D_{bs} . Note that the separation between text fields is also converted to comma during pre-processing. We then define the boundary detection task on the student model as token level binary classification task to predict commas (labeled as 1) after a token in an address:

$$L_{BS} = \sum_{x^{c} \in D_{bs}} \sum_{i=1}^{L} BCE(P(x_{i}^{c}; \theta_{SC}), y_{i}^{c}))$$
(2)

where x^c is an address text, x_i^c is the token at *i* index, $y_i \in \{0, 1\}$, θ_{SC} refers to the parameters of the student encoder model along with the dense layer of binary classification and *BCE* refers to binary cross entropy loss function. Note that both the NER tasks and the boundary detection task share a common encoder.

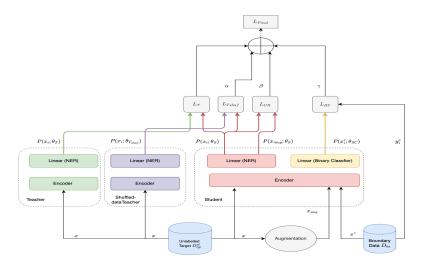


Fig. 1. Student training using multiple tasks

The student model is trained in a multi-task fashion on 4 losses - 1) studentteacher loss with main teacher pseudo labels, 2) student-teacher loss with shuffleddata teacher pseudo labels, 3) consistency regularization loss, and 4) boundary detection task loss. We weigh the 3 new tasks with α , β and γ respectively. Fig. 1 depicts the training of the student model using multiple tasks.

$$L_{Final} = L_T + \alpha * L_{T_{shuf}} + \beta * L_{CR} + \gamma * L_{BS}$$

$$\tag{3}$$

4 Experiments, Data and Results

4.1 Data

We use proprietary e-commerce addresses from 3 countries, namely C1, C2 and C3 as well as external labeled addresses [10] from USA (US), Australia (AU) and Great Britain (GB), to evaluate our model. For the in-house datasets, the train and test data are manually annotated with 4 broad entity types, namely *bld* (building name, number and apartment names), *loc* (locality, sub-locality, community), *road* (road name, road number) and *unit* (apartment number, door number and floor number) in BIO format. We also have access to unlabeled raw customer addresses (with/without boundary signals as entered by the customers) in the e-commerce databases. We test our model under 6 different source to target transfer settings. The external addresses on the other hand are tagged with 6 entity types, namely *Unit*, *StreetNumber*, *StreetName*, *Province*, *PostalCode* and *Municipality*. We intentionally choose more structured countries here to validate the robustness of our approach across emerging as well as established marketplaces. Since we do not have boundary separated customer addresses for this dataset, we synthesized boundaries after each entity in the labeled addresses

using the BIO format. Note that we only test on countries with English addresses. Dataset statistics for each of the countries are mentioned in the table 1. We reveal only the approximate size of the e-commerce datasets due to confidentiality issues.

Table 1. Labeled, unlabeled and boundary data size (in K)

Country	C1	C2	C3	US	GB	AU
Train size	100	20	10	30	30	30
Test size	10	5	1	4	4	4
Unlabeled size	200	100	100	30	30	30
Boundary size	200	100	100	30	30	30

4.2 Experiment Setup

We use Hugging Face RoBERTa-Base as the backbone model. The maximum sequence length is fixed to 70 while batch size as 32; For MLM, the number of epochs is set to 3 while for student and teacher training, it is fixed to 7. We used early stopping with a patience of 2 to terminate the student-teacher training using the validation loss. For MLM, the learning rate is fixed at 3e-5, while for teacher a peak learning rate of 5e-5 and 1e-4 for student is used with a linear schedule and warm steps of 0.1 as in [9]. α , β and γ set to 1(default). A weight decay of 0.01 is used during training and dropout is fixed to 0.1 for student/teacher. AdamW is used as the optimizer and the model is trained on a single Nvidia Tesla T4 GPU. We use micro average F1-score across entity labels as our evaluation metric. Each number reported in section 4.4 is an average of 5 runs with different seeds.

4.3 Baselines

We compare our approach against the following baselines:

- Lower Bound (LB): We train an open-source RoBERTa model on the source data and test it on target addresses in a zero-shot fashion.
- DAPT refers to [3]. Here, we pre-train the model using 20K samples from source and target domain.
- SSTS: [9] is the teacher-student learning approach for NER in a knowledge transfer setting
- SSTS-DAPT: We first run MLM on RoBERTa (DAPT) and then use it to initialize the teacher.
- AdvP: [1] is the recent state-of-the-art for UDA-NER.

	E-commerce					DeepParse			
Country Pair	C1-C2	C1-C3	C3-C1	C3-C2	C2-C1	C2-C3	US-GB	$\operatorname{GB-US}$	AU-US
LB	0.398	0.314	0.320	0.479	0.428	0.393	0.390	0.601	0.354
DAPT	0.602	0.467	0.450	0.465	0.609	0.443	0.630	0.815	0.574
SSTS	0.434	0.391	0.383	0.490	0.453	0.432	0.372	0.598	0.364
AdvP	0.486	0.391	0.379	0.478	0.462	0.456	0.408	0.602	0.358
SSTS-DAPT	0.609	0.494	0.502	0.501	0.647	0.437	0.671	0.828	0.570
Our Approach	0.666	0.511	0.540	0.527	0.693	0.512	0.762	0.816	0.604
wo ST	0.644	0.495	0.522	0.522	0.676	0.483	0.649	0.864	0.567
wo CR	0.658	0.509	0.533	0.513	0.680	0.510	0.758	0.795	0.598
wo BS	0.653	0.491	0.528	0.523	0.677	0.505	0.735	0.818	0.589

Table 2. Performance comparison of all the methods on 9 transfer settings from both e-commerce addresses and [10] addresses measured by micro-average F1-Scores.

Table 4. Entity level results for GB-US transfer.

Table 3. Performance of [11] on e-comm

e 3. Performance of [11] on	Entity	Precision	Recall	F1-Score	
nmerce datasets (F1-Scores)	Municipality	0.964	0.968	0.966	
	PostalCode	0.986	0.997	0.992	
Method C1 C2 C3	Province	0.969	0.982	0.975	
[11] 0.15 0.23 0.12	StreetName	0.909	0.939	0.924	
[] 0.10 0.10 0.11	StreetNumber	0.991	0.993	0.992	
	Unit	0.952	0.970	0.961	

Results, Ablation Studies, Parameter study and Case study 4.4

Performance Analysis As it can be observed from table 2, our approach (or its ablation) outperforms all the baselines on each of the 9 transfer settings. Noticeably, SOTA for UDA-NER like SSTS and AdvP are outperformed by DAPT. We do see that performing student-teacher training on a source/target domain adapted model (SSTS-DAPT) gives a performance boost, suggesting the benefits of adaptive pre-training. Our final proposed approach, however outperforms the best baselines by 2% to 9% which shows the advantages of adding the new tasks. We show the entity type level scores (precision, recall and F1-score) using our approach for one of the DeepParse addresses transfer settings (GB-US) in table 4. The scores mentioned in the table are from the seed with a max overall F1-Score out of the 5 seed runs. Lastly, we test the performance of the multinational parser mentioned in [11] on our e-commerce address dataset shown in the table 3. Since the parser emitted different set of tags as in the e-commerce dataset, we created a mapping between the tag sets (like StreetNumber, Street-Name mapped to road, municipality to loc, unit to unit and bld to O). The low F1-Scores for the datasets show the lack of generalization of the parser to real life customer addresses, thus justifying our claims in Section 2.

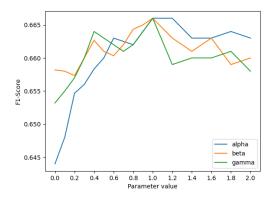


Fig. 2. F1 scores on the test data for C1-C2 when α , β or γ is varied from [0, 2]

Ablation and Parameter Studies We perform ablation studies on all the transfer settings shown in table 2 by removing each of the proposed components - 1) shuffled-data teacher wo ST, 2) consistency regularization wo CR, and 3) boundary detection wo BS. We see a consistent drop in the F1 scores for almost all of the datasets when removing a module. Specifically, the average drop in the scores for wo ST is 2.3%, wo CR is 0.8% and wo BS is 1.2% this. Thus, structural differences between the addresses of different countries can be handled well using shuffled-teacher and boundary detection, while consistency regularization helps dealing with noisy pseudo labels. We also study the impact of the 3 hyperparameters α , β and γ in range [0,2]. Here, we varied one of the hyper-parameters at a time while the others are set to default value of 1. As seen in Fig. 2, we observe an increasing trend in the F1-Score for C1 to C2 transfer with increase in parameter values till 1 and then a slight decreasing trend is observed.

Case study Finally as shown in table 5, we perform a case study for UAE to IN transfer on 2 concrete examples where we compare the parsed outputs of SSTS and our method. In case 1, SSTS wrongly labels 13 XX as road which suggests that the teacher memorized the address structure of UAE (where customers usually enter road names at the start). This was corrected by our approach that correctly labels it as *bld* which can be attributed to the structural in-variance brought in by the shuffled-teacher. Also, it detects *tarsali sussen road* partially while our approach fully recognized the entity. The boundary detection task helps here in detecting the full entity by signalling the student model right boundaries in a target address. In example 2, SSTS missed *marg* (hindi for road) which is a word used specifically in IN (not used in UAE) and mis-classifies it as *loc* while our approach correctly recognized it as *road*. This shows the importance of domain adaptive pre-training using MLM on source and target addresses.

4.5 Training/Inference time

On a total 10K sample size of source and target addresses, MLM based domain adaptive pre-training took 30 mins/epoch. For every address, we created 10 augmentations as done in [2], thus effective training size being 100K. We used 4 GPUs for this training procedure. Teacher training on 10K samples took 1 min/epoch while student training with the 3 tasks on 20K samples took 6.10 min/epoch. The evaluation time on 4K samples was completed in 9s.

Table 5. Qualitative analysis of UAE-IN parsed results

1	Ground Truth: [13 XX] _{bld} [motinagar 2] _{loc} [tarsali
	sussen road] _{road}
	SSTS: [13 XX] _{road} [motinagar 2] _{loc} [tarsali] _{loc} [sussen
	road] _{road}
	Our Approach: [13 XX] _{bld} [motinagar 2] _{loc} [tarsali
	sussen road] _{road}
	Ground Truth: [XX floor] _{unit} [a764] _{unit} [tulsi marg] _{road}
	$[\text{sector19}]_{\text{loc}}$ uttar pradesh
	SSTS: [XX floor] _{unit} [a764] _{unit} [tulsi marg] _{loc}
	[sector19] _{loc} uttar pradesh
	Our Approach: [XX floor] _{unit} [a764] _{unit} [tulsi marg] _{road}
	$[sector 19]_{loc}$ uttar pradesh

5 Industrial Usecase

The address parser is catering to two use-cases in production for an emerging country C, namely 1) address quality scoring, and 2) community identification for launching new services. We trained an address parser for country C using labeled data from another country C1 and integrated parser based address completeness detection with the existing address quality scoring model for country C. This integration led to a 3% increase in the recall of the model for detecting junk addresses. The parser also assisted in identification of high-density communities for the launch of new value added e-commerce services. Our approach resulted in 67% and 133% more communities as compared to those identified earlier by operations team via manual process.

6 Conclusion and future work

In this paper, we propose a student-teacher based framework to transfer knowledge from training address parser on source country with labeled data to a target country with unlabeled data. Our approach uses multiple techniques like training shuffled-data teacher using shuffled source data, data augmentation for sequence tagging data for consistency regularization and learning from boundary signals to improve the target parser. Experiments on multiple e-commerce datasets and external data validate the effectiveness of our approach. In future, we plan to extend the solution to handle source and target countries with different languages and also, leverage multiple sources for training the model for the target country. Also, we wish to explore cases when we do have a limited amount of labeled data for the target country and how to include it in our training framework.

References

- Chen, W., Jiang, H., Wu, Q., Karlsson, B., Guan, Y.: AdvPicker: Effectively Leveraging Unlabeled Data via Adversarial Discriminator for Cross-Lingual NER. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics (Aug 2021). https://doi.org/10.18653/v1/2021.acl-long.61, https://aclanthology.org/2021. acl-long.61
- Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2018). https://doi.org/10.48550/ARXIV.1810.04805, https://arxiv.org/abs/1810. 04805
- Han, X., Eisenstein, J.: Unsupervised domain adaptation of contextualized embeddings for sequence labeling (2019). https://doi.org/10.48550/ARXIV.1904.02817, https://arxiv.org/abs/1904.02817
- Heinzerling, B., Strube, M.: BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation. Miyazaki, Japan (2018)
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach (2019). https://doi.org/10.48550/ARXIV.1907.11692, https://arxiv. org/abs/1907.11692
- 6. openaddresses: Open addresses data. https://openaddresses.io/ (2019)
- 7. openstreetmap: Open street map data. https://www.openstreetmap.org/#map= 5/21.843/82.795 (2019)
- Sajjadi, M., Javanmardi, M., Tasdizen, T.: Regularization with stochastic transformations and perturbations for deep semi-supervised learning (2016). https://doi.org/10.48550/ARXIV.1606.04586, https://arxiv.org/abs/ 1606.04586
- Wu, Q., Lin, Z., Karlsson, B.F., Lou, J.G., Huang, B.: Single-/multi-source cross-lingual ner via teacher-student learning on unlabeled data in target language (2020). https://doi.org/10.48550/ARXIV.2004.12440, https://arxiv.org/ abs/2004.12440
- Yassine, M., Beauchemin, D.: Structured Multinational Address Data (2020), https://github.com/GRAAL-Research/deepparse-address-data
- Yassine, M., Beauchemin, D., Laviolette, F., Lamontagne, L.: Leveraging subword embeddings for multinational address parsing. CoRR abs/2006.16152 (2020), https://arxiv.org/abs/2006.16152
- Yassine, M., Beauchemin, D., Laviolette, F., Lamontagne, L.: Multinational address parsing: A zero-shot evaluation. CoRR abs/2112.04008 (2021), https://arxiv.org/abs/2112.04008

¹² Rishav Sahay et al.