EFFICIENTLY LABELLING SEQUENCES USING SEMI-SUPERVISED ACTIVE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

In natural language processing, deep learning methods are popular for sequence labelling tasks but training them usually requires large amounts of labelled data. Active learning can reduce the amount of labelled training data required by iteratively acquiring labels for the data points a model is most uncertain about. However, active learning methods usually use supervised training and ignore the data points which have not yet been labelled. We propose an approach to sequence labelling using active learning which incorporates both labelled and unlabelled data. We train a locally-contextual conditional random field with deep nonlinear potentials in a semi-supervised manner, treating the missing labels of the unlabelled sentences as latent variables. Our semi-supervised active learning method is able to leverage the sentences which have not yet been labelled to improve on the performance of purely supervised active learning. We also find that using an additional, larger pool of unlabelled data provides further improvements. Across a variety of sequence labelling tasks, our method is consistently able to match 97% of the performance of state of the art models while using less than 30% of the amount of training data.

1 INTRODUCTION

In natural language processing, sequence labelling tasks such as chunking, part-of-speech tagging (POS) and named entity recognition (NER) were traditionally performed using shallow linear models such as hidden Markov models (HMMs) (Kupiec, 1992; Bikel et al., 1999) and conditional random fields (CRFs) (Lafferty et al., 2001). These approaches model the dependencies between adjacent word-level labels. However, when predicting the label for a given word, they do not directly incorporate information from the surrounding words in the sentence (known as 'context'). As a result, deeper models which do use such contextual information, for example convolutional and recurrent networks, have gained popularity (Collobert et al., 2011; Graves, 2012; Huang et al., 2015; Ma & Hovy, 2016).

For deep models to provide significant performance gains over shallow ones, large amounts of labelled data are required (Shen et al., 2018). Acquiring such labelled data is usually expensive, and can require significant manual input from trained annotators (Marcus et al., 1993; Tjong Kim Sang & De Meulder, 2003; Weischedel et al., 2011). This means that methods which can achieve strong performance with limited amounts of labelled data are of significant value.

Active learning is a promising training paradigm to reduce the amount of data required to train such models (Cohn et al., 1995). Initially, a model is trained on a small set of labelled data. Then periodically, more labelled data is added to this set by asking an 'oracle' (usually a human annotator) to label a selection of data points chosen from an unlabelled pool. The model is further trained on this updated set of labelled data and the process is repeated. Active learning has successfully been applied to sequence labelling as well as a variety of other natural language processing tasks (Ringger et al., 2007; Settles & Craven, 2008; Shen et al., 2018; Siddhant & Lipton, 2018). However, this approach usually involves training the model in a supervised fashion, meaning that the available unlabelled data is ignored.

Leveraging the vast amounts of available unlabelled data to improve performance on supervised tasks is a major goal to make progress towards artificial intelligence. Semi-supervised training can be an effective way to do this, and can be combined with active learning. Previous attempts to combine the two include graph-based methods, which can be unscalable to large datasets because

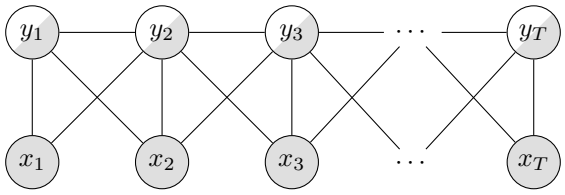


Figure 1: The graph of our locally-contextual CRF. Shaded nodes indicate observed variables and partially shaded nodes indicate variables which may be either observed or unobserved.

they involve inverting the graph Laplacian (Zhu et al., 2003). Others are based on self-training, which can be noisy and error prone since they rely on labelled data being created from the model's own highly confident predictions (Zhou et al., 2004; Tur et al., 2005).

Instead, we adopt a generative approach to semi-supervised active learning. We hypothesise that incorporating unlabelled data to account for the distribution of the sentences in a corpus can improve classification performance compared to using only labelled data. Similarly to McCallum & Nigam (1998), we treat the missing labels of the unlabelled sentences as latent variables. Using common convolutional or recurrent architectures with this approach is difficult because normalising the distribution of the unlabelled sentences is generally intractable. Instead, we use a locally-contextual CRF which directly incorporates information from neighbouring words when modelling the label for a given word. We combine this with deep nonlinear potential functions to provide a flexible model which still allows us to tractably sum over the unobserved variables.

We train the model using active learning with a semi-supervised objective, and acquire sentences using a simple uncertainty based approach. Empirically, we show that this method is able to leverage the data which has not yet been labelled to improve on the performance of purely supervised active learning while having significantly fewer parameters. We observe further performance improvements when using an additional, larger pool of unlabelled data. We also find that our method is less prone to overfitting, and works well when only a limited number of data points can be acquired per round of training. Across all tasks we evaluate on, our method is able to match 97% of the performance of state of the art models with less than 30% of the amount of training data.

The remainder of this paper is structured as follows: the model and training algorithm are presented in Section 2, we compare our approach against prior work in Section 3, the experiments are presented in Section 4 and we make concluding remarks and discuss future work in Section 5.

2 MODEL

In this section, we describe the model which we use for sequence labelling tasks. These include chunking, POS and NER, and involve labelling every word in a sentence according to a predefined set of labels. Throughout, we use the following notation:

- $\mathbf{x} = x_1, \ldots, x_T$ is the sequence of words.
  - $\mathcal{X}$ is the vocabulary, i.e. $x_t \in \mathcal{X} \; \forall \; t$.
- $\mathbf{y} = y_1, \ldots, y_T$ is the corresponding sequence of labels.
  - $\mathcal{Y}$ is the set of possible labels, i.e. $y_t \in \mathcal{Y} \; \forall \; t$.
- θ is the set of parameters to be learned.

We approach sequence labelling using a locally-contextual CRF. Intuitively, it extends the vanilla CRF by using deep nonlinear potential functions, and by directly incorporating information from neighbouring words (local context) when modelling the label for a given word.

The graphical model is shown in Figure 1 and the joint distribution of the sentence $\mathbf{x}$ and labels $\mathbf{y}$ is parametrised as follows:

$$p(\mathbf{x}, \mathbf{y}|\theta) = \frac{\prod_t \psi(y_{t-1}, y_t; \theta)\phi(y_t, x_{t-1}; \theta)\eta(y_t, x_t; \theta)\xi(y_t, x_{t+1}; \theta)}{\sum_{\mathbf{x},\mathbf{y}} \prod_t \psi(y_{t-1}, y_t; \theta)\phi(y_t, x_{t-1}; \theta)\eta(y_t, x_t; \theta)\xi(y_t, x_{t+1}; \theta)} \tag{1}$$

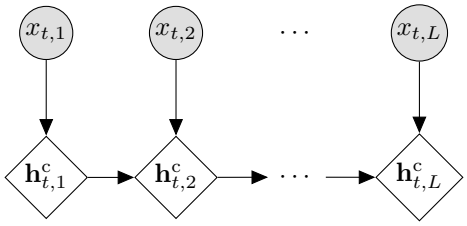


Figure 2: The character-level LSTM used to encode each word. $x_{t,l}$ denotes the l^{th} character of word x_t.

Henceforth, we drop the dependence on θ for brevity. The terms $\psi(y_{t-1}, y_t)$, $\phi(y_t, x_{t-1})$, $\eta(y_t, x_t)$ and $\xi(y_t, x_{t+1})$ are the potentials. The model structure means that the words x_{t-1} and x_{t+1}, in addition to x_t, are directly used when modelling the label y_t.

The potentials are constrained to be positive, so we parametrise them in $\log$-space. Firstly, since the labels y_t are discrete, $\log \psi(y_{t-1}, y_t)$ is simply parametrised using a $|\mathcal{Y}| \times |\mathcal{Y}|$ transition matrix.

We encode the words x_t using a character-level LSTM (Hochreiter & Schmidhuber, 1997) as shown in Figure 2. We take the final state $\mathbf{h}^{\mathrm{c}}_{t,L}$ and concatenate it to a word embedding vector (which we denote as $\mathbf{h}^{\mathrm{w}}(x_t)$) to give the encoding of word x_t:

$$\mathbf{h}(x_t) = [\mathbf{h}^{\mathrm{c}}_{t,L}; \mathbf{h}^{\mathrm{w}}(x_t)] \tag{2}$$

Then, similarly to conditional neural fields (CNFs) (Peng et al., 2009), $\log \phi(y_t, x_{t-1})$, $\log \eta(y_t, x_t)$ and $\log \xi(y_t, x_{t+1})$ are each parametrised by feedforward networks which take as input $\mathbf{h}(x_{t-1})$, $\mathbf{h}(x_t)$ and $\mathbf{h}(x_{t+1})$ respectively. Each network outputs a $|\mathcal{Y}|$-dimensional vector whose inner product is taken with the one-hot encoding of y_t (which we denote as $\mathbf{e}(y_t)$). Specifically:

$$\log \phi(y_t, x_{t-1}) = \mathbf{e}(y_t) \cdot f^{\phi}(\mathbf{h}(x_{t-1})) \tag{3}$$

$$\log \eta(y_t, x_t) = \mathbf{e}(y_t) \cdot f^{\eta}(\mathbf{h}(x_t)) \tag{4}$$

$$\log \xi(y_t, x_{t+1}) = \mathbf{e}(y_t) \cdot f^{\xi}(\mathbf{h}(x_{t+1})) \tag{5}$$

where f^{ϕ}, f^{η} and f^{ξ} are the feedforward networks.

2.1 Training

2.1.1 Active learning

We train the model using active learning, which has been shown to reduce the amount of training data required to achieve good performance on sequence labelling tasks (Ringger et al., 2007; Shen et al., 2018; Siddhant & Lipton, 2018).

During each round of training, the active learning algorithm chooses a fixed number m of sentences to be labelled. These sentences are removed from the unlabelled dataset, and together with their newly acquired labels, are added to the labelled dataset. The model parameters are updated for a fixed number of iterations, then the next round begins.

To select which sentences to label next, we use the 'least confidence' strategy (Culotta & McCallum, 2005). That is, the m sentences with the largest value of $1 - p(\mathbf{y}^*|\mathbf{x})$ are chosen, where:

$$p(\mathbf{y}^*|\mathbf{x}) = \max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x}) \tag{6}$$

Empirically, we find this strategy consistently works well with our model across several different datasets.

2.1.2 Objective

At a given iteration, the model has access to all of the sentences which have been labelled so far, as well as the remaining unlabelled sentences. We denote $\tilde{p}^l(\mathbf{x}, \mathbf{y})$ and $\tilde{p}^u(\mathbf{x})$ as the empirical distributions over the labelled and unlabelled data respectively. Then, using $\mathcal{L}^l_\theta(\mathbf{x}, \mathbf{y})$ and $\mathcal{L}^u_\theta(\mathbf{x})$ as

the objectives for the labelled and unlabelled data respectively, we maximise the following semi-supervised objective with respect to the parameters θ:

$$\mathcal{L}_\theta = \sum_{(\mathbf{x},\mathbf{y})\sim\tilde{p}^l} \mathcal{L}^l_\theta(\mathbf{x},\mathbf{y}) + \sum_{\mathbf{x}\sim\tilde{p}^u} \mathcal{L}^u_\theta(\mathbf{x}) \tag{7}$$

Supervised training

For the labelled data, the natural objective to maximise is $\mathcal{L}^l_\theta(\mathbf{x},\mathbf{y}) = \log p(\mathbf{y}|\mathbf{x})$:

$$\begin{aligned}\mathcal{L}^l_\theta(\mathbf{x},\mathbf{y}) = \log p(\mathbf{y}|\mathbf{x}) =& \sum_t \left[\log\psi(y_{t-1},y_t) + \log\phi(y_t,x_{t-1}) + \log\eta(y_t,x_t) + \log\xi(y_t,x_{t+1})\right] \\ &- \log\sum_{\mathbf{y}}\prod_t \psi(y_{t-1},y_t)\phi(y_t,x_{t-1})\eta(y_t,x_t)\xi(y_t,x_{t+1})\end{aligned} \tag{8}$$

The first term is straightforward to compute. The second term can be computed using a recursion analogous to the Baum-Welch algorithm for HMMs (Baum et al., 1970). We define:

$$\alpha^{\mathbf{y}}_t = \sum_{y_{t-1}} \alpha^{\mathbf{y}}_{t-1}\psi(y_{t-1},y_t)\phi(y_t,x_{t-1})\eta(y_t,x_t)\xi(y_{t-1},x_t) \tag{9}$$

Then:

$$\sum_{\mathbf{y}}\prod_t \psi(y_{t-1},y_t)\phi(y_t,x_{t-1})\eta(y_t,x_t)\xi(y_t,x_{t+1}) = \sum_{y_T}\alpha^{\mathbf{y}}_T \tag{10}$$

Unsupervised training

For the unlabelled data, we maximise $\mathcal{L}^u_\theta(\mathbf{x}) = \log p(\mathbf{x})$:

$$\begin{aligned}\log p(\mathbf{x}) =& \log\sum_{\mathbf{y}}\prod_t \psi(y_{t-1},y_t)\phi(y_t,x_{t-1})\eta(y_t,x_t)\xi(y_t,x_{t+1}) \\ &- \log\sum_{\mathbf{x},\mathbf{y}}\prod_t \psi(y_{t-1},y_t)\phi(y_t,x_{t-1})\eta(y_t,x_t)\xi(y_t,x_{t+1})\end{aligned} \tag{11}$$

The computation for the first term is shown in Equations (9) and (10). For the second term, we use a similar recursion. We define:

$$\alpha^{\mathbf{x}}_t = \sum_{x_{t-1},y_{t-1}} \alpha^{\mathbf{x}}_{t-1}\psi(y_{t-1},y_t)\phi(y_t,x_{t-1})\eta(y_t,x_t)\xi(y_{t-1},x_t) \tag{12}$$

Then:

$$\sum_{\mathbf{x},\mathbf{y}}\prod_t \psi(y_{t-1},y_t)\phi(y_t,x_{t-1})\eta(y_t,x_t)\xi(y_t,x_{t+1}) = \sum_{x_T,y_T}\alpha^{\mathbf{x}}_T \tag{13}$$

Note that with $\mathcal{L}^l_\theta(\mathbf{x},\mathbf{y}) = \log p(\mathbf{y}|\mathbf{x})$ and $\mathcal{L}^u_\theta(\mathbf{x}) = \log p(\mathbf{x})$, the objective function in Equation (7) allows us to form a consistent estimator for θ. That is, for data $(\mathbf{x}^{(n)},\mathbf{y}^{(n)}) \sim p(\mathbf{x},\mathbf{y}|\theta_0)$, the objective has an optimum at $\theta = \theta_0$ as $n \to \infty$.

Inference

During inference, we want to find $\mathbf{y}^*$ such that:

$$\mathbf{y}^* = \arg\max_{\mathbf{y}} \ \log p(\mathbf{y}|\mathbf{x}) \tag{14}$$

This can be done using the Viterbi algorithm (Viterbi, 1967), which simply replaces the sum in Equation (9) with a $\max$ operation.

DATASET	TASK	LABELS	TRAIN	VALIDATION	TEST
CoNLL 2000	Chunking	11	7,936*	1,000*	2,012
Penn Treebank	POS	45	38,219	5,527	5,462
CoNLL 2003	NER	4	14,987	3,466	3,684
OntoNotes	NER	18	59,924	8,528	8,262

Table 1: Statistics of each of the datasets used. We use the standard splits for all datasets.

* The CoNLL 2000 dataset does not include a validation set. We therefore randomly sample 1,000 sentences from the training set to use for validation.

3 RELATED WORK

Although a somewhat unexplored area, there have been some successful applications of active learning to sequence labelling. Ringger et al. (2007) use a maximum entropy Markov model for POS, achieving strong performance with small amounts of labelled data. For NER, Culotta & McCallum (2005) use a CRF and acquire data with the least confidence strategy while Shen et al. (2004) use an SVM with a combination of multiple acquisition strategies. Shen et al. (2018) perform active learning for NER with a deep model combining convolution and recurrent layers, obtaining results competitive with the state of the art with relatively small amounts of labelled data.

Generative semi-supervised sequence labelling approaches include that of Mohit & Hwa (2005) who perform syntax-based NER. This is done by training a naive Bayes classification model using an expectation maximisation algorithm. Other semi-supervised sequence labelling approaches include structural learning (Ando & Zhang, 2005), generalised expectation criteria (Mann & McCallum, 2008), maximum discriminant functions (Suzuki & Isozaki, 2008), self-learned features (Qi et al., 2009), cross-view training (Clark et al., 2018), moment matching (Marinho et al., 2016) and unsupervised pre-training (Peters et al., 2017; Akbik et al., 2018; Devlin et al., 2019).

Among semi-supervised active learning approaches, ours is most similar to that of McCallum & Nigam (1998). The missing labels of the unlabelled sentences are treated as latent variables, and semi-supervised learning is performed by combining the query-by-committee method with expectation maximisation. Alternative approaches include graph-based methods, such as that of Zhu et al. (2003). This method requires inverting the graph Laplacian, which scales quadratically with the size of the dataset, making it impractical to run at scale. Self-training methods (Zhou et al., 2004; Tur et al., 2005) rely on labelled data being created from the model's own highly confident predictions. These methods introduce additional noise into the training process and can be error prone since they can reinforce the model's own incorrect predictions.

4 EXPERIMENTS

In this section, we evaluate the performance of our model on chunking, POS and NER. We use the following datasets:

Chunking We use the CoNLL 2000 dataset (Tjong Kim Sang & Buchholz, 2000), which consists of dividing sentences into syntactically correlated parts of words according to a set of predefined chunk types. This task is evaluated using the F1 score.

POS We use the Wall Street Journal portion of the Penn Treebank dataset (Marcus et al., 1993), which consists of labelling each word in a sentence according to a set of predefined part-of-speech tags. This task is evaluated using accuracy.

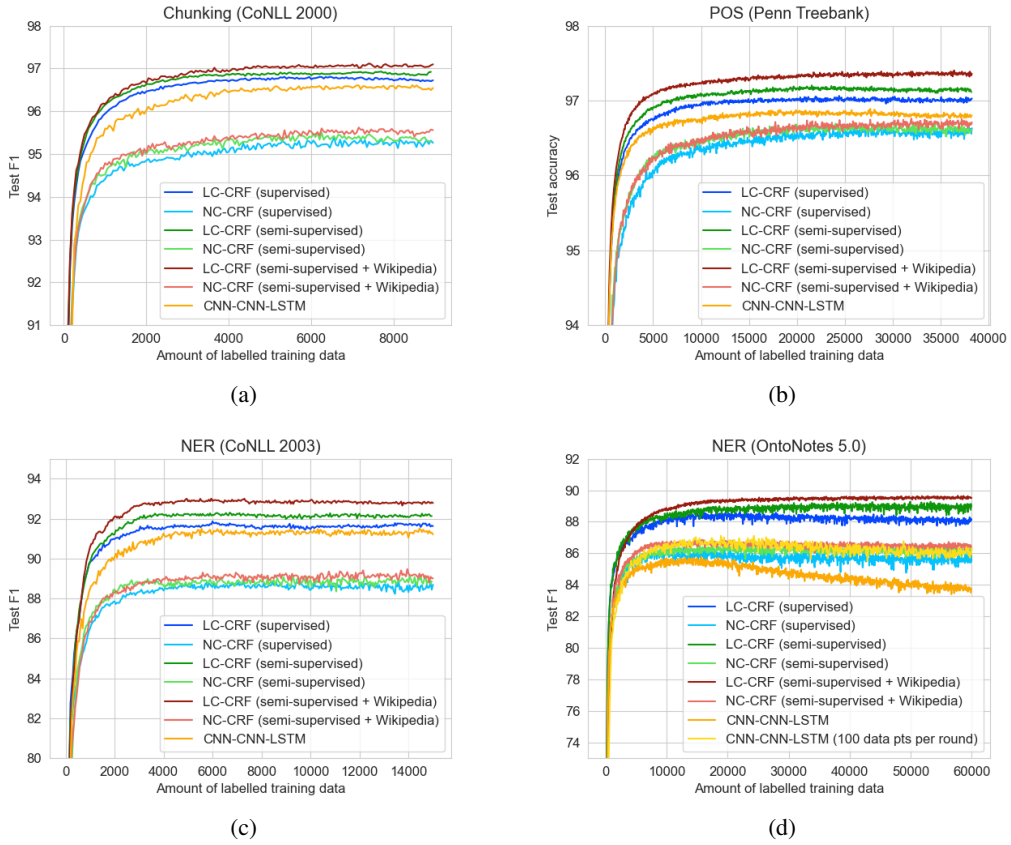
NER We use the CoNLL 2003 English (Tjong Kim Sang & De Meulder, 2003) and OntoNotes 5.0 English (Pradhan et al., 2013) datasets. Each of these datasets consists of labelling each word in a sentence according to a set of predefined entity types. This task is evaluated using the F1 score.

Statistics for all of the datasets are shown in Table 1.

We train both a supervised and a semi-supervised version of our model. In order to assess the effect of additional unlabelled data, we also train a semi-supervised version of our model with a larger

MODEL	PARAMETERS
NC-CRF	0.8M
LC-CRF	2.4M
CNN-CNN-LSTM (Shen et al., 2018)	7.3M

Table 2: The number of parameters in each of the models we train.



(a) (b) (c) (d)

Figure 3: Active learning performance of the NC-CRF, LC-CRF and CNN-CNN-LSTM (Shen et al., 2018) on the test set of each task. The x-axes show the number of sentences which have been labelled so far, and the y-axes show the model performance at that amount of labelled training data. For all tasks/metrics, higher is better.

pool of unlabelled data. For this extra unlabelled data, we use the English portion of Wikipedia. We exclude sentences whose length exceeds the maximum sentence length of, and which contain words outside of the vocabulary of the respective supervised datasets.

In addition, in order to evaluate the benefit of using the local context, we also train a non-contextual CRF. This is the same as the model described in Section 2 but with the $\phi(y_t, x_{t-1})$ and $\xi(y_t, x_{t+1})$ potentials removed. We abbreviate the locally-contextual CRF to LC-CRF, and the non-contextual CRF to NC-CRF. We compare the performance of our model to a reimplementation of the CNN-CNN-LSTM presented by Shen et al. (2018), using the reported hyperparameters. This is trained using purely supervised active learning. For all of our experiments, we report the average results over 3 runs.

4.1 ARCHITECTURE

For the character level LSTM, we use a single layer with 50 units. We initialise the word embeddings using 300-dimensional GloVe embeddings (Pennington et al., 2014) and update them during

MODEL	CHUNKING F1	POS ACC.	NER F1	
			CONLL 2003	ONTONOTES
NC-CRF – Active (sup.)	95.26	96.56	88.69	85.48
NC-CRF – Active (semi-sup.)	95.29	96.61	88.87	86.25
NC-CRF – Active (semi-sup. + Wiki)	96.56	96.71	89.01	86.36
LC-CRF – Active (sup.)	96.72	97.03	91.67	88.13
LC-CRF – Active (semi-sup.)	96.92	97.14	92.13	88.96
LC-CRF – Active (semi-sup. + Wiki)	97.10	97.36	92.78	89.52
CNN-CNN-LSTM (Shen et al., 2018)	96.54	96.80	91.29	83.69
CNN-CNN-LSTM (Shen et al., 2018) – 100 data points per round	—	—	—	86.33
Best published	97.30 (Liu et al., 2019)	97.96 (Bohnet et al., 2018)	93.50 (Baevski et al., 2019)	92.07 (Li et al., 2020b)
	97.00 (Clark et al., 2018)	97.85 (Akbik et al., 2018)	93.47 (Liu et al., 2019)	91.11 (Li et al., 2020a)
	96.72 (Akbik et al., 2018)	97.78 (Ling et al., 2015)	93.47 (Jiang et al., 2019)	90.30 (Luo et al., 2020)

Table 3: Final results of the NC-CRF, LC-CRF and CNN-CNN-LSTM (Shen et al., 2018) on the test set of each task. The best published results for each task are included in the bottom part of the table. For all tasks/metrics, higher is better.

training. Out-of-vocabulary words are replaced by an unknown token. For the feedforward networks f^{ϕ}, f^{η} and f^{ξ} referred to in Equations (3), (4) and (5), we use 2 layers with skip connections, 600 units and ReLU nonlinearities (Glorot et al., 2011).

Table 2 shows the number of parameters in each of the models we train using active learning. Both the NC-CRF and LC-CRF have significantly fewer parameters than the CNN-CNN-LSTM.

4.2 OPTIMISATION

During each round of training, we acquire 50 labelled sentences according to the strategy described in Section 2.1.1 and perform 50 iterations of stochastic gradient descent with a learning rate of 0.001 and Nesterov momentum of 0.9 (Nesterov, 1983). For the supervised version, we use mini-batches with 128 labelled sentences. For the semi-supervised version, we use mini-batches with 128 labelled and 128 unlabelled sentences.

In the objective function in Equation (7), the unsupervised component $\mathcal{L}_{\theta}^{u}(\mathbf{x})$ is typically much larger (in absolute value) than the supervised component $\mathcal{L}_{\theta}^{l}(\mathbf{x}, \mathbf{y})$. This means that early in training, parameter updates favour improving $\mathcal{L}_{\theta}^{u}(\mathbf{x})$ at the expense of $\mathcal{L}_{\theta}^{l}(\mathbf{x}, \mathbf{y})$. To alleviate this problem, we multiply the unsupervised part of the objective by a constant prefactor, which we set to 0.1. For stability, we set this prefactor to 0 when there are fewer than 500 unlabelled sentences remaining.

% PERFORMANCE OF BEST REPORTED	% TRAINING DATA REQUIRED			
	CHUNKING F1	POS ACC.	NER F1	
			CONLL 2003	ONTONOTES
95%	1.84%	0.69%	4.85%	7.34%
97%	3.47%	1.43%	6.94%	26.87%
99%	14.36%	9.27%	18.40%	—*

Table 4: The percentage of training data required for the LC-CRF (Active (semi-supervised + Wiki)) to match 95%, 97% and 99% performance of the best reported method on each task.

* Performance not reached.

4.3 RESULTS

The active learning training curves for the NC-CRF, LC-CRF and CNN-CNN-LSTM are shown in Figure 3, and the final results are shown in Table 3. For context, the best published results for each task are also included.

Across every task, the NC-CRF performs significantly worse than the LC-CRF, demonstrating the utility of using the local context. For both the NC-CRF and LC-CRF, semi-supervised training consistently outperforms purely supervised training. Moreover, using an additional pool of unlabelled data provides a further boost to the semi-supervised performance. We also find that on all of the tasks, each version of the LC-CRF outperforms the CNN-CNN-LSTM. These results support our initial hypothesis that semi-supervised active learning can improve the performance of deep models compared to purely supervised active learning.

Figure 3d shows that when performing NER on OntoNotes 5.0, both the CNN-CNN-LSTM and, albeit to a lesser degree, the LC-CRF (supervised) suffer from overfitting. In contrast, neither of the semi-supervised versions of the LC-CRF have the same behaviour. Training an additional version of the CNN-CNN-LSTM where the number of labelled sentences acquired per round is increased from 50 to 100 reduces the overfitting. These results suggest that semi-supervised active learning can help to prevent overfitting, and that the purely supervised approach may not be suitable in scenarios where only a small amount of labelled data can be acquired between each round of training.

In Table 4, we show the percentage of training data required for the LC-CRF (Active (semi-supervised + Wiki)) to match 95%, 97% and 99% performance of the best reported method on each task. These results demonstrate the data-efficiency of our method which, across all tasks, reaches 97% of the performance achieved by state of the art models with less than 30% of the amount of labelled training data. Furthermore, on chunking, POS, and NER on CoNLL 2003, our method reaches 99% of the state of the art performance with less than 20% of the amount of labelled training data.

5 CONCLUSION

We have proposed a model and training procedure for labelling sequences using semi-supervised active learning, which treats the missing labels of the unlabelled sentences as latent variables. We find that our approach is able to leverage the data which has not yet been labelled to improve on the performance of purely supervised active learning on a variety of tasks. Moreover, using an additional larger pool of unlabelled data provides a further increase in performance. We also find that our method is less prone to overfitting, and works well when only a limited number of data points can be acquired per round of training. Across all tasks we evaluate on, our method is able to match 97% of the performance of state of the art models with less than 30% of the amount of training data.

Our method presents plenty of avenues for future work. The model parameters can be treated as latent random variables, which has been shown to improve uncertainty estimation (Gal et al., 2017). External knowledge can be incorporated into the model, which has been shown to improve performance on NER (Seyler et al., 2017). A sentence-level latent variable could be introduced to model semantic features of the text, which may further improve classification performance.

REFERENCES

Alan Akbik, Duncan Blythe, and Roland Vollgraf. Contextual String Embeddings for Sequence Labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, 2018.

Rie Ando and Tong Zhang. A High-Performance Semi-Supervised Learning Method for Text Chunking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, 2005.

Alexei Baevski, Sergey Edunov, Yinhan Liu, Luke Zettlemoyer, and Michael Auli. Cloze-driven Pretraining of Self-attention Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2019.

Leonard E. Baum, Ted Petrie, George Soules, and Norman Weiss. A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains. *Annals of Mathematical Statistics*, 41, 1970.

Daniel M. Bikel, Richard Schwartz, and Ralph M. Weischedel. An Algorithm That Learns What's in a Name. *Machine Learning*, 34, 1999.

Bernd Bohnet, Ryan McDonald, Gonçalo Simões, Daniel Andor, Emily Pitler, and Joshua Maynez. Morphosyntactic Tagging with a Meta-BiLSTM Model over Context Sensitive Token Encodings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2018.

Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc Le. Semi-Supervised Sequence Modeling with Cross-View Training. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018.

David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. Active Learning with Statistical Models. In *Advances in Neural Information Processing Systems*, 1995.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural Language Processing (Almost) from Scratch. *Journal of Machine Learning Research*, 12, 2011.

Aron Culotta and Andrew McCallum. Reducing Labeling Effort for Structured Prediction Tasks. In *Proceedings of the 20th National Conference on Artificial Intelligence*, 2005.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, 2019.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep Bayesian Active Learning with Image Data. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep Sparse Rectifier Neural Networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, 2011.

Alex Graves. *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer, 2012.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9, 1997.

Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional LSTM-CRF Models for Sequence Tagging. In *arXiv:1508.01991*, 2015.

Yufan Jiang, Chi Hu, Tong Xiao, Chunliang Zhang, and Jingbo Zhu. Improved Differentiable Architecture Search for Language Modeling and Named Entity Recognition. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2019.

Julian Kupiec. Robust Part-of-Speech Tagging Using a Hidden Markov Model. *Computer Speech & Language*, 6, 1992.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the 18th International Conference on Machine Learning*, 2001.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. A Unified MRC Framework for Named Entity Recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020a.

Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. Dice Loss for Data-imbalanced NLP Tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020b.

Wang Ling, Chris Dyer, Alan W Black, Isabel Trancoso, Ramón Fermandez, Silvio Amir, Luís Marujo, and Tiago Luís. Finding Function in Form: Compositional Character Models for Open Vocabulary Word Representation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015.

Yijin Liu, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. GCDT: A Global Context Enhanced Deep Transition Architecture for Sequence Labeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

Ying Luo, Fengshun Xiao, and Hai Zhao. Hierarchical Contextualized Representation for Named Entity Recognition. In *Proceedings of the Thrity-Fourth AAAI Conference on Artificial Intelligence*, 2020.

Xuezhe Ma and Eduard Hovy. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2016.

Gideon S. Mann and Andrew McCallum. Generalized Expectation Criteria for Semi-Supervised Learning of Conditional Random Fields. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics*, 2008.

Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19, 1993.

Zita Marinho, André F. T. Martins, Shay B. Cohen, and Noah A. Smith. Semi-Supervised Learning of Sequence Models with Method of Moments. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016.

Andrew McCallum and Kamal Nigam. Employing EM and Pool-Based Active Learning for Text Classification. In *Proceedings of the Fifteenth International Conference on Machine Learning*, 1998.

Behrang Mohit and Rebecca Hwa. Syntax-based Semi-Supervised Named Entity Tagging. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, 2005.

Y. E. Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. *Dokl. Akad. Nauk SSSR*, 269, 1983.

Jian Peng, Liefeng Bo, and Jinbo Xu. Conditional Neural Fields. In *Advances in Neural Information Processing Systems*. 2009.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 2014.

Matthew Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. Semi-supervised Sequence Tagging with Bidirectional Language Models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 2017.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. Towards Robust Linguistic Analysis using OntoNotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, 2013.

Y. Qi, P. Kuksa, R. Collobert, K. Sadamasa, K. Kavukcuoglu, and J. Weston. Semi-Supervised Sequence Labeling with Self-Learned Features. In *Ninth IEEE International Conference on Data Mining*, 2009.

Eric Ringger, Peter McClanahan, Robbie Haertel, George Busby, Marc Carmen, James Carroll, Kevin Seppi, and Deryle Lonsdale. Active Learning for Part-of-Speech Tagging: Accelerating Corpus Annotation. In *Proceedings of the Linguistic Annotation Workshop*, 2007.

Burr Settles and Mark Craven. An Analysis of Active Learning Strategies for Sequence Labeling Tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, 2008.

Dominic Seyler, Tatiana Dembelova, Luciano Del Corro, Johannes Hoffart, and Gerhard Weikum. KnowNER: Incremental Multilingual Knowledge in Named Entity Recognition. *CoRR*, abs/1709.03544, 2017.

Dan Shen, Jie Zhang, Jian Su, Guodong Zhou, and Chew-Lim Tan. Multi-Criteria-based Active Learning for Named Entity Recognition. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, 2004.

Yanyao Shen, Hyokun Yun, Zachary C. Lipton, Yakov Kronrod, and Animashree Anandkumar. Deep Active Learning for Named Entity Recognition. In *International Conference on Learning Representations*, 2018.

Aditya Siddhant and Zachary C. Lipton. Deep Bayesian Active Learning for Natural Language Processing: Results of a Large-Scale Empirical Study. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018.

Jun Suzuki and Hideki Isozaki. Semi-Supervised Sequential Labeling and Segmentation Using Giga-Word Scale Unlabeled Data. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics*, 2008.

Erik F. Tjong Kim Sang and Sabine Buchholz. Introduction to the CoNLL-2000 Shared Task Chunking. In *Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop*, 2000.

Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL*, 2003.

Gokhan Tur, Dilek Hakkani-Tür, and Robert E. Schapire. Combining active and semi-supervised learning for spoken language understanding. *Speech Communication*, 45, 2005.

Andrew J. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13, 1967.

Ralph Weischedel, Eduard Hovy, Mitchell Marcus, Martha Palmer, Robert Belvin, Sameer Pradhan, Lance Ramshaw, and Nianwen Xue. *OntoNotes: A Large Training Corpus for Enhanced Processing*. Springer, 2011.

Zhi-Hua Zhou, Ke-Jia Chen, and Yuan Jiang. Exploiting Unlabeled Data in Content-Based Image Retrieval. In *Proceedings of the 15th European Conference on Machine Learning*, 2004.

Xiaojin Zhu, John Lafferty, and Zoubin Ghahramani. Combining Active Learning and Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions. In *Proceedings of the ICML Workshop on the Continuum from Labeled to Unlabeled Data*, 2003.