Supplementary Material for Learning with Structured Representations for Negation Scope Extraction

Hao Li and **Wei Lu**
Singapore University of Technology and Design
8 Somapah Road, Singapore, 487372
hao_li@mymail.sutd.edu.sg, luwei@sutd.edu.sg

1 Training and Decoding

We introduce objective functions and gradient computation as well as the inference method for Linear, Semi and Latent models as follow.

Linear and Semi

The probability of predicting a possible output $\mathbf{y}$, which is the label sequence capturing negation scope information, given an input sentence $\mathbf{x}$ is defined as:

$$p(\mathbf{y}|\mathbf{x}) = \frac{\exp\left(\mathbf{w}^T\mathbf{f}(\mathbf{x},\mathbf{y})\right)}{\sum_{\mathbf{y}'\in\mathcal{Y}(\mathbf{x})}\exp(\mathbf{w}^T\mathbf{f}(\mathbf{x},\mathbf{y}'))} \quad (1)$$

where $\mathbf{w}$ is the weight vector, and $\mathbf{f}(\mathbf{x},\mathbf{y})$ is the feature vector defined over the sentence $\mathbf{x}$, and the output structure $\mathbf{y}$. $\mathcal{Y}(\mathbf{x})$ is the space of all the possible negation scope that are compatible with input $\mathbf{x}$.

During the learning process, we aim to minimize the L_2-regularized negative joint log-likelihood of our dataset, defined as:

$$\mathcal{L}(\mathbf{w}) = \sum_i \log \sum_{\mathbf{y}'\in\mathbf{Y}(\mathbf{x}^{(i)})}\exp(\mathbf{w}^T\mathbf{f}(\mathbf{x}^{(i)},\mathbf{y}')) - \sum_i \mathbf{w}^T\mathbf{f}(\mathbf{x}^{(i)},\mathbf{y}^{(i)}) + \lambda\mathbf{w}^T\mathbf{w} \quad (2)$$

where $(\mathbf{x}^{(i)},\mathbf{y}^{(i)})$ is the i-th training instance and λ is the L_2 regularization hyper-parameter. Here the output $\mathbf{y}$ basically conveys the information annotated in the dataset, including negation scope.The objective function is convex and we use L-BFGS (Liu and Nocedal, 1989) to optimize it, where the gradient with respect to each parameter w_k can be computed as:

$$\frac{\partial\mathcal{L}(\mathbf{w})}{\partial w_k} = \sum_i \mathbf{E}_{p(\mathbf{y}'|\mathbf{x}^{(i)})}\Big[f_k(\mathbf{x}^{(i)},\mathbf{y}')\Big] - \sum_i f_k(\mathbf{x}^{(i)},\mathbf{y}^{(i)}) + 2\lambda w_k \quad (3)$$

Latent

The probability of predicting a possible output $\mathbf{y}$, which is the label sequence capturing negation scope information, given an input sentence $\mathbf{x}$ is defined as:

$$p(\mathbf{y}|\mathbf{x}) = \frac{\sum_{\mathbf{h}}\exp\left(\mathbf{w}^T\mathbf{f}(\mathbf{x},\mathbf{y},\mathbf{h})\right)}{\sum_{\mathbf{y}'\in\mathcal{Y}(\mathbf{x})}\sum_{\mathbf{h}}\exp(\mathbf{w}^T\mathbf{f}(\mathbf{x},\mathbf{y}',\mathbf{h}))}$$

where a latent variable $\mathbf{h}$ represents the underlying patterns.

During the learning process, we aim to minimize the L_2-regularized negative joint log-likelihood of our dataset, defined as:

$$\mathcal{L}(\mathbf{w}) = \sum_i \log\sum_{\mathbf{y}'}\sum_{\mathbf{h}'}\exp(\mathbf{w}^T\mathbf{f}(\mathbf{x}^{(i)},\mathbf{y}',\mathbf{h}')) - \sum_{\mathbf{i}}\log\sum_{\mathbf{h}}\mathbf{w}^T\mathbf{f}(\mathbf{x}^{(i)},\mathbf{y}^{(i)},\mathbf{h}) + \lambda\mathbf{w}^T\mathbf{w}$$

where $(\mathbf{x}^{(i)},\mathbf{y}^{(i)})$ is the i-th training instance and λ is the L_2 regularization hyper-parameter. Here the output $\mathbf{y}$ basically conveys the information annotated in the dataset, including negation scope.

We can use standard gradient-based methods to optimize the objective function. For the Linear CRF with latent variable, we choose to use L-BFGS (Liu and Nocedal, 1989) as the optimization algorithm, which was previously shown effective in optimizing similar objective functions (Blunsom et al., 2008). The gradient with respect to each parameter w_k can be computed as:

$$\frac{\partial\mathcal{L}(\mathbf{w})}{\partial w_k} = \sum_i \mathbf{E}_{p(\mathbf{y}',\mathbf{h}'|\mathbf{x}^{(i)})}\Big[f_k(\mathbf{x}^{(i)},\mathbf{y},\mathbf{h}')\Big] - \sum_i \mathbf{E}_{p(\mathbf{h}|\mathbf{x}^{(i)},\mathbf{y}^{(i)})}\Big[f_k(\mathbf{x}^{(i)},\mathbf{y}^{(i)},\mathbf{h})\Big] + 2\lambda w_k$$

Decoding

The inference procedure involves calculating the objective function value and the gradients above. We implement the marginal inference procedure based on a generalized forward-backward style algorithm, which allows us to perform exact inference based on dynamic programming.

At the decoding phase, we adopt standard MAP inference using a procedure that is analogous to the conventional Viterbi algorithm. From the decoded structured representation, we can simply read off the predicted negation scope.

In terms of time complexity of inference procedure, the **Linear** models are $O(nT^2)$, where where n is the sentence length, and T is the number of labels, which is 2 in this case. The **Latent** models take $O(nT^2)$, where T is the number of labels times latent size, which is 4 in this case. The **Semi** models take $O(nLT^2)$, where T is 2 and L is the maximum span length according to the corpus.

2 Evaluation Metrics

We can conduct evaluations of negation scope extraction based on metrics defined at different levels, namely token-level evaluations and scope-level evaluations. There are two versions of evaluation metrics defined at the scope-level that can be used to measure the performance according to *SEM2012 shared task (Morante and Blanco, 2012). The official PERL script released in the corpus is adopted to report results in both versions: version A (used to rank the systems during *SEM2012 shared task competition) and version B (introduced in the final stage of the competition). Both versions use *precision*, *recall* and *F1-measure*, except that they differ when calculating FP for scope-level evaluations, resulting in the differences in precision calculation defined as $\frac{TP}{TP+FP}$, where TP and FP refer to "true positive" and "false positive" respectively. In version A, FP only considers the case that the predicted scope is found but no scope is found in the gold instance. However, in version B, FP considers both of the following cases: 1) the predicted scope does not exactly match with the gold scope; 2) the predicted scope is found, but the gold instance contains no scope. Note that in the *SEM 2012 shared task, the ranking of the competition remains the same when switching the evaluation metrics from version A to the version B. In this work, we report both versions of results based on the official evaluation script for fair comparisons.[1]

Besides, *percentage of correct scope* (PCS) is another metric for scope-level evaluations. It is defined as the number of TP over the total number of non-empty scope instances in the gold data, which is essentially the scope-level *recall* score.

References

Phil Blunsom, Trevor Cohn, and Miles Osborne. 2008. A discriminative latent variable model for statistical machine translation. In *Proc. of ACL*. http://www.aclweb.org/anthology/P/P08/P08-1024.

Federico Fancellu, Adam Lopez, and Bonnie L Webber. 2016. Neural networks for negation scope detection. In *Proc. of ACL*. https://doi.org/10.18653/v1/p16-1047.

Dong C Liu and Jorge Nocedal. 1989. On the limited memory bfgs method for large scale optimization. *Mathematical programming* https://link.springer.com/article/10.1007/BF01589116.

Roser Morante and Eduardo Blanco. 2012. *sem 2012 shared task: Resolving the scope and focus of negation. In *Proc. of *SEM 2012*. http://www.aclweb.org/anthology/S12-1035.

[1] We note that the official script ignores punctuation symbols, while we found some prior work does consider them in evaluations (Fancellu et al., 2016). To make a fair comparison with these works, we decide to also consider punctuation symbols in our evaluations by adapting the official script. We note that empirically, doing so can lead to slightly lower numbers than using the official script when reporting our numbers.