Pronoun Prediction with Latent Anaphora Resolution

Christian Hardmeier
Uppsala University
Department of Linguistics and Philology
Box 635, 751 26 Uppsala, Sweden
christian.hardmeier@lingfil.uu.se

Abstract

This paper describes the UU-Hardmeier submissions to the WMT 2016 shared task on cross-lingual pronoun prediction. Our model is a system combination of two different approaches, one based on a neural network with latent anaphora resolution and the other one on an n-gram model with an additional dependency on the source pronoun. The combination of the two models results in an improvement over each individual system, but it appears that the contribution of the neural network is more likely due to its context modelling capacities than to the anaphora resolution subnetwork.

1 Introduction

The primary submission of the UU-Hardmeier system to the pronoun prediction shared task at WMT 2016 (Guillou et al., 2016) consists of two components. The first is a reimplementation of the pronoun prediction neural network proposed by Hardmeier et al. (2013). The other system component is based on a standard n-gram language model over the lemmas of the target side. Apart from implementation details, the main difference between this model and the official baseline provided by the shared task organisers is the integration of information about the pronoun found on the source side, which allows the model to recognise whether a given pronoun was singular or plural in the source.

2 Neural Network Component

The first component of our model is a modified reimplementation of the pronoun prediction network introduced by Hardmeier et al. (2013). The main differences between the model used in this work and the previous implementation are the following:

- A complete reimplementation of the neural network code based on Theano (The Theano Development Team, 2016) and Keras (Chollet, 2016).
- Substitution of the coreference preprocessing component by CORT (Martschat and Strube, 2015).
- Inclusion of target-language context lemma and part-of-speech features.
- (Accidental) omission of a hidden layer in the submitted systems.
- Substitution of the internal softmax layer (**V**) by a sigmoid layer.

The overall structure of the network is shown in figure 1. To create input data for the network, we first generate a set of antecedent candidates for a given pronoun by running the preprocessing pipeline of the coreference resolution system CORT (Martschat and Strube, 2015). Each training example for our network can have an arbitrary number of antecedent candidates. Next, we prepare four types of features. *Anaphor source context features* describe the source language (SL) pronoun (**P**) and its immediate context consisting of three words to its left (**L1** to **L3**) and three words to its right (**R1** to **R3**), encoded as one-hot vectors. *Anaphor target context features* cover a window of three TL lemmas and part-of-speech tags to the left and to the right of the pronoun, each encoded as a one-hot vector.

Antecedent features (**A**) describe an antecedent candidate. Candidates are represented by the TL words aligned to the syntactic head of the source language markable noun phrase, again represented as one-hot vectors. These vectors cannot be fed into the network directly because their number depends on the number of antecedent candidates and on the

Proceedings of the First Conference on Machine Translation, Volume 2: Shared Task Papers, pages 576–580,
Berlin, Germany, August 11-12, 2016. ©2016 Association for Computational Linguistics

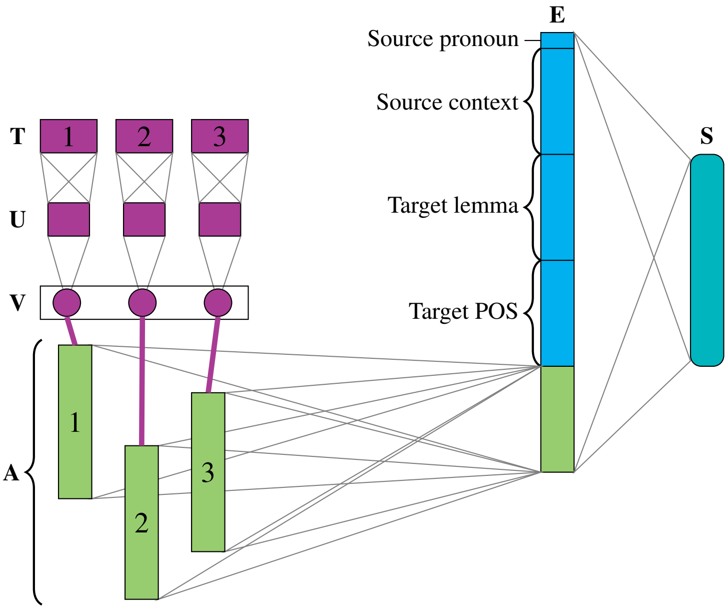

Figure 1: Neural network with latent anaphora resolution

number of TL words aligned to the head word of each antecedent. Instead, they are averaged to yield a single vector per antecedent candidate.

Finally, *anaphoric link vectors* (**T**) describe the relationship between an anaphor and a particular antecedent candidate. These vectors are generated by the feature extraction machinery in CORT and include a standard set of features for coreference resolution borrowed wholesale from the default configuration of the coreference resolution system, including a number of lexicalised feature templates that generate a large number of individual features. To increase the efficiency of the training process, all input feature sets are limited to a vocabulary consisting of the 1000 most frequent words per feature type.

In the forward propagation pass, the input word representations are mapped to a low-dimensional representation in an embedding layer (**E**). In this layer, the embedding weights for all the SL vectors (the pronoun and its 6 context words) are tied, so if two words are the same, they are mapped to the same lower-dimensional embedding regardless of their position relative to the pronoun. To process the information contained in the antecedents, the network first computes the link probability for each antecedent candidate. The anaphoric link features (**T**) are mapped to a hidden layer with logistic sigmoid units (**U**). The activations of the hidden units are then mapped to a single value, which functions as an element in an internal softmax layer over all antecedent candidates (**V**). This softmax layer assigns a probability $p_1 \dots p_n$ to each antecedent candidate. The antecedent feature vectors **A** are projected to lower-dimensional embeddings, weighted with their corresponding link probabilities and summed. The weighted sum is then concatenated with the source language embeddings in the **E** layer. To improve the training of the antecedent-related network parts, whenever a training example is presented to the network, with a probability of 20 % all source and target context features are set to zero. The **E** layer is connected to a softmax output layer predicting the pronoun class as defined by the shared task specification.

In our setup, the dimensionality of the word embeddings is 30 for the source context words, target lemmas and antecedent features and 15 for the target POS features, resulting in a total embedding layer size of 482 (two source pronoun features, six 30-dimensional source context embeddings, six 30-dimensional target lemma embeddings, six 15-dimensional target POS embeddings and one 30-dimensional antecedent feature vector). The network is regularised with an ℓ_2 penalty that was set to 10^{-6} using grid search over a held-out development set. It is trained with the ADAGRAD algorithm with minibatches of size 16 and with cross-entropy as the training objective. The gradients are computed using backpropagation. Note that the number of weights in the network is the same for all training examples even though the number of antecedent

Source:	**It** 's got these fishing lures on the bottom .
Target lemmas:	**REPLACE_0** avoir ce leurre de pêche au-dessous .
Solution:	*ils*
LM training data:	It **REPLACE_*ils*** avoir ce leurre de pêche au-dessous .
LM test data:	It **REPLACE** avoir ce leurre de pêche au-dessous .

Figure 2: Data for the source-aware language model

candidates varies because all weights related to antecedent word features and anaphoric link features are shared between all antecedent candidates. The model is trained for 60 epochs on the training data in the IWSLT set; the other training data sets are not used.

3 Source-Aware Language Model

In the pronoun prediction task at DiscoMT 2015 (Hardmeier et al., 2015), it turned out that a simple n-gram model considering only the target-side local context of the word to be predicted outperformed all submissions to the shared task. These results suggest that it is important to include strong n-gram modelling capacities into any system. The neural network system described in the previous section does not necessarily have this, so we decided to address this problem with a system combination approach.

The official baseline of the current shared task is identical to that of the previous year, but the task is different in that the target language words are provided in lemmatised form only. Lemmatisation deprives the language model of important morphological information about the context words, in particular about their number. As a result, we observe much lower scores with the official baseline than in the 2015 shared task. Frequently, however, a look at the source pronoun would be entirely sufficient to supply the required information for the source language at least, and while the correspondence of number marking across languages is not perfect, the number of the pronoun in the source language is a strong hint.

Our source-aware language model is an n-gram model trained on an artificial corpus generated from the target lemmas of the parallel training (Figure 2). Before every REPLACE tag occurring in the data, we insert the source pronoun aligned to the tag (without lowercasing or any other processing). The alignment information attached to the REPLACE tag in the shared task data files is stripped off. In the training data, we instead add the pronoun class to be predicted. The n-gram model used for this component is a 6-gram model with modified Kneser-Ney smoothing (Chen and Goodman, 1998) trained with the KenLM toolkit (Heafield, 2011) on the complete set of training data provided for the shared task.

To predict classes for an unseen test set, we first convert it to a format matching that of the training data, but with a uniform, unannotated REPLACE tag used for all classes. We then recover the tag annotated with the correct solution using the `disambig` tool of the SRILM language modelling toolkit (Stolcke et al., 2011). This tool runs the Viterbi algorithm to select the most probable mapping of each token from among a set of possible alternatives. The map used for this task trivially maps all tokens to themselves with the exception of the REPLACE tags, which are mapped to the set of annotated REPLACE tags found in the training data.

The source-aware language model described here is identical to the base model of the UUPPSALA system (Loáiciga et al., 2016). Its output was submitted to the shared task as the UUPPSALA primary submission for English–German, German–English and French–English and as the UUPPSALA contrastive submission for English–French.

4 System Combination

To combine the neural predictor with the source-aware language model, we linearly interpolated the probabilities assigned to each class by each model. The class finally predicted was the one that scored highest according to the interpolated probability distribution.

The neural network prediction probabilities are obtained trivially as the posterior distribution of the final softmax layer **S**. For the source-aware language model, we run SRILM's `disambig` tool with the `-posteriors` option, which causes it

English–French

Class	NN	LM	NN+LM
ce	0.865	0.825	0.855
elle	–	0.483	0.325
elles	–	0.167	0.143
il	0.624	0.667	0.677
ils	0.787	0.747	0.810
cela	0.603	0.542	0.679
on	–	0.400	0.444
OTHER	0.873	0.889	0.905
Macro-F	0.469	0.590	0.605
Macro-R	0.508	0.598	0.606

English–German

Class	NN	LM	NN+LM
er	–	0.091	–
sie	0.793	0.716	0.788
es	0.684	0.688	0.718
man	–	0.182	0.222
OTHER	0.756	0.729	0.800
Macro-F	0.447	0.481	0.506
Macro-R	0.466	0.474	0.504

NN: neural network (Section 2; contrastive submission)
LM: source-aware language model (Section 3)
NN+LM: interpolated model (Section 4; primary submission)

Table 1: F-scores per class and macro-averaged F-score and recall for component and combined systems

to output an approximate posterior distribution derived from information collected during the Viterbi decoding pass. For all classes i, the probability $p_{\text{NN}}(i)$ predicted by the neural network and the probability $p_{\text{LM}}(i)$ predicted by the source-aware language model were combined as follows:

$$p(i) = \lambda\, p_{\text{NN}}(i) + (1-\lambda)\, p_{\text{LM}}(i) \qquad (1)$$

The single weight λ ($0 \leq \lambda \leq 1$) was determined by grid search on a linearly spaced grid of step size 0.1 to maximise the macro-averaged recall score for the *DiscoMT2015.test* corpus (for English–French) and the *TEDdev* corpus (for English–German). The weights used by the submitted systems are $\lambda = 0.5$ for English–French and $\lambda = 0.6$ for English–German. The fact that the optimal weight setting assigns close to equal weight to the two systems for both language pairs demonstrates that both systems have complementary information to contribute and both of them are useful to improve the overall result.

5 Results and Discussion

Table 1 shows the F-scores per class for each of the two component systems and for the system combination that we submitted as our primary system. The most important observation that we can make is the complete failure of the neural network model to predict the infrequent classes: *elle*, *elles* and *on* for English–French and *er* and *man* for English–German. This is highly disappointing since we hoped that the neural network, with its ability to see potential antecedents, would be in a better position to make accurate predictions for these classes. Good performance for the French feminine plural class *elles* was a key motivating factor in our initial development of the pronoun prediction network (Hardmeier et al., 2013), but unfortunately we have repeatedly struggled to produce similarly good results with different data sets and tasks. In this shared task, we are forced to conclude that the effect of the neural network classifier is detrimental for the French feminine singular and plural classes and for the German masculine singular when combined with the source-aware language model.

In the system combination, we do observe improvements over the source-aware language model for all other classes, including the infrequent generic classes *on* and *man*. For the latter two classes, the neural network brings about an improvement in the combination even though it completely fails to predict the classes on its own.

In sum, the score patterns of our two component systems suggest that the value added in this task by the neural network stems from its better ability to distinguish between the various impersonal pronoun classes rather than, as we had hoped, from improved performance on anaphoric pronouns.

Acknowledgements

This work was supported by the Swedish Research Council under project 2012-916 *Discourse-Oriented Statistical Machine Translation*. The experiments were performed on resources provided by SNIC through the Uppsala Multidisciplinary Center for Advanced Computational Science (UPPMAX) under project SNIC2016/7-36.

References

Stanley F. Chen and Joshua Goodman. 1998. An empirical study of smoothing techniques for language modeling. Technical report, Computer Science Group, Harvard University, Cambridge (Mass.).

François Chollet. 2016. Keras. `https://github.com/fchollet/keras`.

Liane Guillou, Christian Hardmeier, Preslav Nakov, Sara Stymne, Jörg Tiedemann, Yannick Versley, Mauro Cettolo, Bonnie Webber, and Andrei Popescu-Belis. 2016. Findings of the 2016 WMT shared task on cross-lingual pronoun prediction. In *Proceedings of the First Conference on Machine Translation (WMT16)*. Association for Computational Linguistics, Berlin (Germany).

Christian Hardmeier, Preslav Nakov, Sara Stymne, Jörg Tiedemann, Yannick Versley, and Mauro Cettolo. 2015. Pronoun-focused MT and cross-lingual pronoun prediction: Findings of the 2015 DiscoMT shared task on pronoun translation. In *Proceedings of the 2nd Workshop on Discourse in Machine Translation (DiscoMT 2015)*. Lisbon (Portugal).

Christian Hardmeier, Jörg Tiedemann, and Joakim Nivre. 2013. Latent anaphora resolution for cross-lingual pronoun prediction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Seattle (Washington, USA), pages 380–391.

Kenneth Heafield. 2011. KenLM: faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Edinburgh (Scotland, UK), pages 187–197.

Sharid Loáiciga, Liane Guillou, and Christian Hardmeier. 2016. It-disambiguation and source-aware language models for cross-lingual pronoun prediction. In *Proceedings of the First Conference on Machine Translation (WMT16)*. Association for Computational Linguistics, Berlin (Germany).

Sebastian Martschat and Michael Strube. 2015. Latent structures for coreference resolution. *Transactions of the Association for Computational Linguistics* 3:405–418.

Andreas Stolcke, Jing Zheng, Wen Wang, and Victor Abrash. 2011. SRILM at sixteen: Update and outlook. In *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*. Waikoloa (Hawaii, USA).

The Theano Development Team. 2016. Theano: A Python framework for fast computation of mathematical expressions. *ArXiv e-prints* 1605.02688.