Target-Bidirectional Neural Models for Machine Transliteration

Andrew Finch and **Lemao Liu** and **Xiaolin Wang** and **Eiichiro Sumita**
National Institute of Information and Communications Technology (NICT)
Advanced Translation Technology Laboratory
3-5 Hikaridai
Keihanna Science City
619-0289 JAPAN
{andrew.finch,lmliu,xiaolin.wang,eiichiro.sumita}@nict.go.jp

Abstract

Our purely neural network-based system represents a paradigm shift away from the techniques based on phrase-based statistical machine translation we have used in the past. The approach exploits the agreement between a pair of target-bidirectional LSTMs, in order to generate balanced targets with both good suffixes and good prefixes. The evaluation results show that the method is able to match and even surpass the current state-of-the-art on most language pairs, but also exposes weaknesses on some tasks motivating further study. The Janus toolkit that was used to build the systems used in the evaluation is publicly available at https://github.com/lemaoliu/Agtarbidir.

1 Introduction

Our primary system for the NEWS shared evaluation on transliteration generation is different in character from all our previous systems. In past years, all our systems have been based on phrase-based statistical machine translation (PB-SMT) techniques, stemming from the system proposed in (Finch and Sumita, 2008). This year's system is a pure end-to-end neural network transducer. In (Finch et al., 2012) auxiliary neural network language models (both monolingual and bilingual (Li et al., 2004)) were introduced as features to augment the log-linear model of a phrase-based transduction system, and led to modest gains in system performance. In the NEWS 2015 workshop (Finch et al., 2015) neural transliteration systems using attention-based sequence-to-sequence neural network transducers (Bahdanau et al., 2014) were applied to transliteration generation. In isolation, the performance was found to be lower than that of the phrase-based system on all of the tasks, however we observed that the neural network transducer was very effective when used as a model for re-scoring the output of the phrase-based transduction process, and this led to respectable improvements relative to previous systems on most of the tasks.

Our focus this year has been on the development of an end-to-end purely neural network-based system capable of competitive performance. The changes and improvements over the sequence-to-sequence neural transducer used in NEWS2015 are as follows:

- A target-bidirectional agreement model was employed.
- Ensembles of neural networks were used rather than just a single network.
- The ensembles were selected from different training runs and different training epochs according to their performance on development (and test) data.

In all our experiments we have taken a strictly language independent approach. Each of the language pairs was processed automatically from the character sequence representation supplied for the shared tasks, with no language specific treatment for any of the language pairs. Furthermore no preprocessing was performed on any of the data with the exception of uppercasing the English to ensure consistency among the data sets.

2 System Description

2.1 Target-bidirectional Models

Our system uses the target-bidirectional approach proposed in (Liu et al., 2016), and the reader is referred to this paper for a full description of the method we employ. In brief, we use pairs of LSTM RNN sequence-to-sequence transducers that first

Proceedings of the Sixth Named Entity Workshop, joint with 54th ACL, pages 78–82,
Berlin, Germany, August 12, 2016.

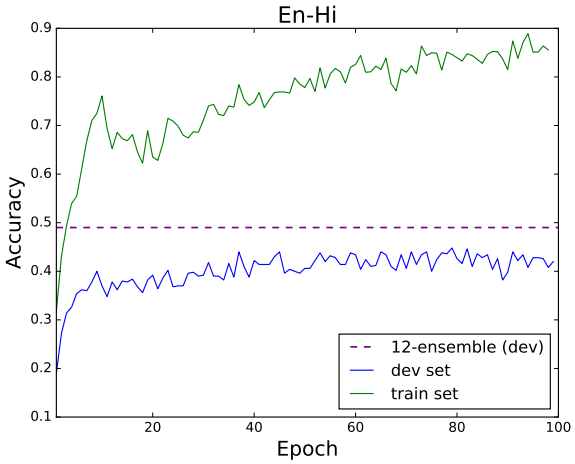

Figure 1: En-Hi training performance.

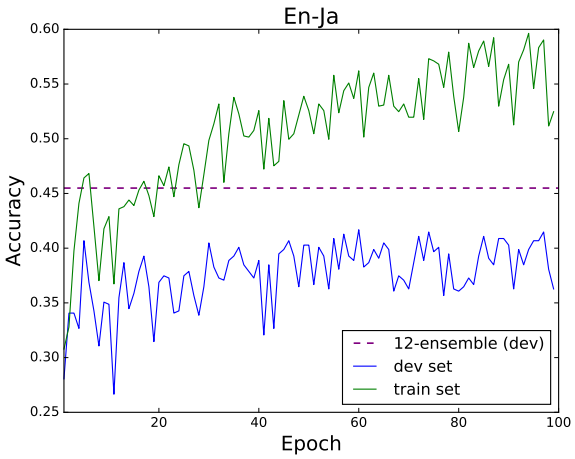

Figure 2: En-Ja training performance.

encode the input sequence into a fixed length vector, and then *decode* from this to produce the target. The method mitigates a fundamental shortcoming in neural sequence-to-sequence transduction, in which errors in prediction accumulate in the context vectors used to make the predictions, leading to progressively worse performance as the generation process proceeds. The result is unbalanced output which has high quality prefixes that degrade to lower quality suffixes. Our bidirectional agreement model overcomes this by using a pair of RNNs that generate from both left-to-right and right-to-left, producing 2 k-bests lists which are combined[1] in order to encourage agreement between the models. In (Liu et al., 2016) it is shown that the resulting output is both more balanced and of substantially higher quality than that resulting from either unidirectional model. Furthermore it

[1] In all experiments reported here we used the joint k-best approximation method.

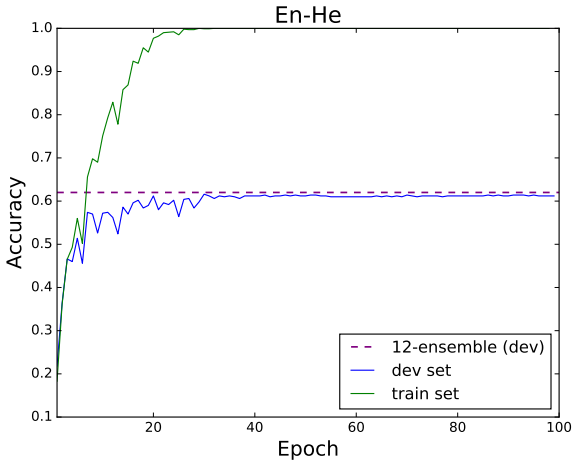

Figure 3: En-He training performance.

is shown that the gains from this method cannot be obtained from larger ensembles of unidirectional models. The approach was shown to be effective in both grapheme-to-phoneme conversion (where it set a new state-of-the-art benchmark), and in English-Japanese transliteration. This paper evaluates the method on a much wider variety of tasks highlighting some of the strengths and weaknesses of the new approach.

2.2 Ensembles

Multiple neural networks were combined into ensembles. This was done by linear interpolation (with equal weights) of the probability distributions produced by the networks over the target vocabulary during the beam search decoding.

3 Experimental Methodology

3.1 Corpora

The neural networks were trained on all of the data for each task, with the exception of 500 pairs which were used for development. The development data was used in order to determine whether or not the networks had fully trained, and also as a means of selecting the neural network models that comprised the ensembles.

In this year's workshop, 15 runs on the test data were permitted. 12 of the runs were used to evaluate the models to be used in the ensembles, and the remaining 3 runs were used to determine the ensemble size. In order to remove the advantage of using test data during system development (to maintain cross-comparability with previous years' results), one of the ensembles used was composed of all 12 of the networks ('12-ensemble' in Figure 1).

In addition, in order to observe the performance of the models on the training set during training, a sample of 1000 pairs was taken from the training data.

3.2 Training

Each of the systems was trained for 100 epochs. For all language pairs the accuracy on the development set appeared to stop improving after approximately 50 epochs. Graphs showing the performance of the systems during training are shown in Figure 1 which represent typical training runs, together with interesting exceptions in Figures 3 and 2. The green (upper) solid line on the graphs represents the accuracy on training data, the blue (lower) solid line represents the accuracy on unseen development data, and the dashed purple line represents the performance of an ensemble composed of the best performing 12 neural networks on the development set.

The curves in Figure 1 are typical, with the performance of the system on training data still steadily increasing at epoch 100, but with the performance on development data reaching its maximum value often by epoch 20, and almost always by epoch 50. We therefore conclude that the networks are all fully trained after 50 epochs. Furthermore, we did not observe any noticable degradation in performance after epoch 50 due to overfitting.

The curves in Figures 2 and 3 are atypical. On En-Ja the variance in accuracy from epoch to epoch was unusually high. The gains from using ensembles of networks were also larger than for other language pairs. In addition, the accuracy on training data remained lower than most language pairs. The curves for En-He show the opposite behavior. The accuracy on training data is 1.0 after about 35 epochs indicating that the neural network has effectively memorized the training data. At this point the variance in accuracy from epoch to epoch falls to almost zero. The gains from using ensembles for this language pair are very small.

We were unable to train a neural network with high accuracy on the Ar-En dataset, and as a consequence did not enter a system on this task this year. The reasons for this are not yet clear, but the system had reasonably good f-scores with very low accuracy. The networks seemed able to produce plausible output, that was rarely an exact match with the reference. We believe the neural network may have been able to generalize from the data, but was not able to memorize it well.

Training times were dependent on the language pair. Most language pairs completed the 100-epoch training on a single Tesla K40m GPU in under a day. Training for the Arabic-English task was around 10 times longer due to the larger training set.

3.3 Ensemble Selection

In order to form ensembles we need to select the ensemble size, and also the neural networks that will comprise the ensemble. In pilot experiments, we found that it is possible to obtain respectable improvement by building ensembles from the networks at different epochs during training. Our strategy was to train 5 target-bidirectional RNNs for each language pair, and select the ensemble from the epochs within these 5 runs.

In this year's workshop, 15 evaluations were permitted on test data for each task. We used 12 of these to evaluate the target-bidirectional RNNs, and 3 to select the ensemble size from $\{4, 8, 12\}$. The ensembles of size 12 were selected using development data only as follows: the best 2 target-bidirectional RNNs were selected from epochs of each of the 5 training runs, then the best 2 target-bidirectional RNNs were chosen from the remaining epochs/runs. Ensembles of 4 and 8 were selected from the candidate set of 12 (that were selected using the development set), according to their accuracy on the test data. We found a moderate positive correlation between training and development set accuracy at each epoch of the training. This suggests that the variance in the accuracy of networks from epoch to epoch during training is not simply random noise, but that 'good' and 'bad' networks exist at different epochs, and this motivated our strategy to select them based on development set accuracy.

3.4 Architecture and Parameters

The network architecture for all of the networks used in all tasks was the same, and was chosen because it has proven to be effective in other experiments. The computational expense associated with working with neural networks on this task prohibited us from running experiments to select the optimal architecture, and therefore it is possible that architectures that are considerably better than the one we have chosen exist.

The RNNs consisted of a single layer of 500 LSTMs, with 500-unit embeddings on the source and target sides. AdaDelta (Zeiler, 2012) was used for training with a minibatch size of 16. A beam

Language Pair		2012 system	2015 system	2016 Baseline	2016 12-ensemble	2016 Primary
English to Bengali	(EnBa)	0.460	0.483	0.287	**0.498**	**0.498**
Chinese to English	(ChEn)	0.203	0.184	0.098	0.211	**0.214**
English to Chinese	(EnCh)	0.311	0.313	0.193	0.309	**0.316**
English to Hebrew	(EnHe)	0.154	0.179	0.109	0.184	**0.189**
English to Hindi	(EnHi)	0.668	0.696	0.270	0.709	**0.715**
English to Japanese Katakana	(EnJa)	0.401	0.407	0.209	0.464	**0.465**
English to Kannada	(EnKa)	0.546	0.562	0.196	0.570	**0.583**
English to Korean Hangul	(EnKo)	**0.384**	0.363	0.218	0.348	0.352
English to Persian	(EnPe)	0.655	**0.697**	0.482	0.691	0.696
English to Tamil	(EnTa)	0.592	0.626	0.258	0.613	**0.629**
English to Thai	(EnTh)	0.122	0.157	0.068	0.179	**0.187**
English to Japanese Kanji	(JnJk)	0.513	**0.610**	0.461	0.327	0.327
Thai to English	(ThEn)	0.140	0.154	0.091	0.194	**0.196**

Table 1: The official evaluation results in terms of the top-1 accuracy.

search with beam width 12 was used to obtain the k-best hypotheses. Decoding was aborted, and a null hypothesis output when a target sequence was generated that was three times longer than the source (sequences of length less than 6 were not aborted).

4 Evaluation Results

The official scores for our system are given in Table 1, alongside the scores of our previous systems on the same test set, and the scores of the official baseline system. The highest scores are highlighted in bold, and it is clear that this year's system has attained higher accuracy than the systems from previous years on most of the language pairs. For some pairs, such as English-Katakana, English-Thai and Thai-English, the improvement is substantial. However, there are also tasks in which the neural system was not able to match the performance of the previous system, notably English-Japanese Kanji, English-Hangul and Arabic-English. The first two of these tasks have quite large vocabularies on the target side, and this may make them less suitable for a neural approach. The Arabic-English task has no such issues, and furthermore has a far larger training corpus available which ought to favor the neural method, however it differs from the other tasks in that short vowels are not represented in written Arabic, but must still be generated on the target side. Further research is necessary to determine the true cause, but our conjecture is that phrase-based systems, which effectively memorize the training data in a piecewise manner, are consequently more successful on this task than neural networks which are geared more towards generalization rather than memorization.

5 Conclusion

The system used for this year's shared evaluation signals a paradigm shift away from the phrase-based systems based on machine translation technology used by our group in earlier years. Our end-to-end neural machine transliteration system leverages the agreement between target-bidirectional RNN ensembles to improve its performance. On most of the transliteration tasks the system has shown itself to be capable of matching and even surpassing the current state-of-the-art. We believe neural networks have a bright future in the field of transliteration generation, and the experiments on the NEWS Workshop datasets have uncovered outstanding issues that will make interesting topics for future research as this technology matures.

Acknowledgements

For the English-Japanese, English-Korean and Arabic-English datasets, the reader is referred to the CJK website: http://www.cjk.org. For English-Hindi, English-Tamil, and English-Kannada, and English-Bangla the data sets originated from the work of (Kumaran and Kellner, 2007) [2]. The Chinese language corpora came from the Xinhua news agency (Xinhua News Agency, 1992). The English Persian corpus originates from the work of (Karimi et al., 2006; Karimi et al., 2007).

[2]http://research.microsoft.com/india

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*.

Andrew Finch and Eiichiro Sumita. 2008. Phrase-based machine transliteration. In *Proceedings of the 3rd International Joint Conference on Natural Language Processing (IJCNLP)*, volume 1, Hyderabad, India.

Andrew Finch, Paul Dixon, and Eiichiro Sumita. 2012. Rescoring a phrase-based machine transliteration system with recurrent neural network language models. In *Proceedings of the 4th Named Entity Workshop (NEWS) 2012*, pages 47–51, Jeju, Korea, July. Association for Computational Linguistics.

Andrew Finch, Lemao Liu, Xiaolin Wang, and Eiichiro Sumita. 2015. Neural network transduction models in transliteration generation. In *Proceedings of the Fifth Named Entity Workshop*, pages 61–66, Beijing, China, July. Association for Computational Linguistics.

Sarvnaz Karimi, Andrew Turpin, and Falk Scholer. 2006. English to persian transliteration. In *SPIRE*, pages 255–266.

Sarvnaz Karimi, Andrew Turpin, and Falk Scholer. 2007. Corpus effects on the evaluation of automated transliteration systems. *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*.

A. Kumaran and Tobias Kellner. 2007. A generic framework for machine transliteration. In *SIGIR'07*, pages 721–722.

Haizhou Li, Min Zhang, and Jian Su. 2004. A joint source-channel model for machine transliteration. In *ACL '04: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 159, Morristown, NJ, USA. Association for Computational Linguistics.

Lemao Liu, Andrew Finch, Masao Utiyama, and Eiichiro Sumita. 2016. Agreement on target-bidirectional lstms for sequence-to-sequence learning. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Xinhua News Agency. 1992. Chinese transliteration of foreign personal names. *The Commercial Press*.

Matthew D. Zeiler. 2012. ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701.