# Cross-lingual Natural Language Modeling

*Grandee Lee*

National University of Singapore, Singapore

grandee.lee@u.nus.edu

## 1. Motivation

We focus on a particular type of language modeling which is known as cross-lingual language modeling because it suffers both from data sparsity and domain adaption making it very challenging and meaningful as a research topic. While the neural network brings about performance gains, it has not successfully addressed these two issues since the approach is essentially data-driven. In cross-lingual language modeling, we seek to find the common structures in both languages, so that learning one language also helps us in learning another language which could potentially be under-resourced. We focus our work on the study of code-switching (CS) language whereby the speaker is free to use either language, we seek to find the common structure or mapping space which help us to model both languages effectively. In both cases, the data is usually sparse, due to the lack of high-quality alignment or the low occurrence of such data under the natural circumstance. Code-switching or code-mixing is an increasingly common linguistic behavior among bilingual speakers [1].

Intra-sentential CS speech poses a significant challenge to ASR systems [2]. This is because CS introduces more vocabulary choices at each prediction step due to words from another language, at the same time, it occurs sparingly and freely without adhering to rigid syntactic or grammatical rules [3]. Speaker may choose when to and not to switch given the same preceding context. The challenge is further exacerbated because there are far less CS linguistic resources than monolingual ones. In general, we rely on large text corpora in written form, such as newspapers and books, for monolingual language modeling. As CS takes place mostly in spoken form, we cannot find as much documented CS text as monolingual text for language modeling.

## 2. Contributions

### 2.1. Code-switch Language Model

Since the data is sparse and the domains are different, the current work [4] focuses on drawing strength from transfer learning and making use of synthetic data to supplement the real data. For brevity, in transfer learning, under the setting of CS language modeling, we adopt a method of pre-training a cross-lingual embedding space which seeks to map the words in both languages to a common space. Upon this space, we perform clustering to supplement downstream tasks. Mathematically, we seek to address this problem by combining the strength of auxiliary class and back-off scheme as summarized in the formulation below.

$$p(w_{t+1}|w_{<t}) = p(w_{t+1}|w_{<t}, c_{t+1})p(c_{t+1}|c_{<t}) \quad (1)$$

We realize the above formulation using a recurrent neural network and would expect the predicted $c_{t+1}$, which is more reliable than word prediction with limited data, to provide stronger back-off to the word prediction network. We implement this model through 1) sharing a common embedding space which allows for an architectural improvement over the traditional multi-task based language model and at the same time 2) using the predicted embedding of the auxiliary class $c_{t+1}$ as the input to the word prediction network together with $w_t$ (Fig 1). There

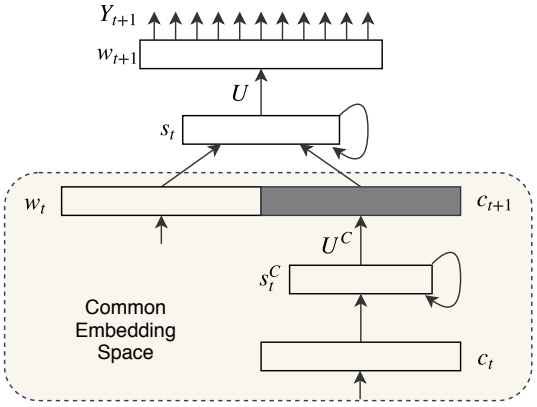


Figure 1: *Common Space embedding and class back-off Language Model (CSLM). $U$ and $U^C$ are the projection matrices.*

| Model | PPL Dev | PPL Eval |
|---|---|---|
| RNNLM† [5] | 246.60 | 287.88 |
| FL + OF† [5] | 219.85 | 239.21 |
| FLM† [6] | 177.79 | 192.08 |
| LSTM* [7] | 150.65 | 153.06 |
| Multi-task* [7] | 141.86 | 141.71 |
| CSLM | 128.12 | 129.85 |
| CSLM + Multi-task | 128.54 | 128.02 |

Table 1: *Language model baseline on SEAME [8] test set. Models marked with † indicate that training and testing are done on SEAME Phase I which approximate to 60% of SEAME Phase II in term of total tokens. Models marked with * indicate training and testing done on SEAME Phase II.*

is 9.7% perplexity reduction between *CSLM + Multi-task* over the previous state-of-the-art CS language model *Multi-task* in Table 1. CSLM draws strength from pre-trained cross-lingual word embedding, the successful sharing of this information from the auxiliary class prediction to the lexicon prediction with minimal loss and providing strong back-off. Compared to previous models which use multi-task objective, there is generally lack of strong back-off scheme. Without a common embedding space, the lexicon model would have to solve the additional task of feature extraction since the useful feature from auxiliary class

and lexicon embedding are from different space.

### 2.2. Synthetic Data Pre-training

For the synthetic data generation, we use linguistic rule which is analogous to physical rules of a simulator to generate synthetic data close to the real domain to supplement training [9]. We show that by constraining the word embedding in a common space we can make better use of the predicted cluster for word prediction which significantly reduces the perplexity. This work is also motivated by language model fine-tuning [10], where a pre-trained language model is later fine-tuned for downstream tasks much like the case of pre-training on ImageNet in vision. Although we do not follow strictly the proposed method such as using the slanted triangular learning rates and adding a new task-specific layer, we are motivated by the idea of proposing a good initial prior so that subsequent task can be improved or achieve comparable performance with much fewer data. Such pre-training and fine-tuning technique also have the additional advantage of faster convergence.

The insertional assumption in Matrix Language Frame theory strongly motivates the use of aligned parallel data. Given a pair of aligned sentences as shown in Fig. 2, we can randomly select a few words in Chinese and substitute them with their aligned counterparts to synthesize a Chinese matrix and English embedded CS sentence, and vice versa for English.

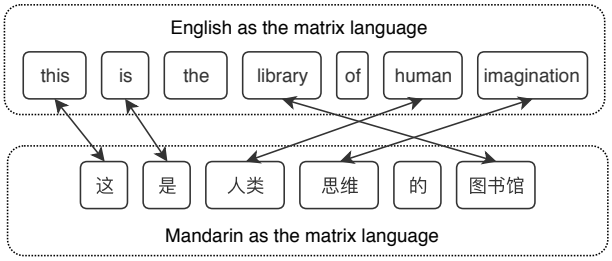


Figure 2: *An example of aligned parallel sentences. In the naive approach, the aligned words are randomly inserted into each of the matrix language.*

The baseline model is the one trained from scratch with SEAME *Monolingual* data first and then *Train* to ensure the only difference between the models is synthetic data pre-training.

Table 2: *Perplexity of the model under the various training scenarios. The synthetic corpus used for pre-training is phrase aligned with switch probability* $p_{cs} = 0.7$.

| Model | Pre-training | Training | Perplexity |
|---|---|---|---|
| Baseline | No | SEAME *Train* | 219 |
| *Pre*CS1 | *Synthetic* | No | 359 |
| *Pre*CS2 | *Synthetic* | SEAME *Train* | **173** |
| *No*CS | *Parallel* | SEAME *Train* | 223 |

The perplexity reduction of the model *Pre*CS2, which is pre-trained on the *Synthetic* CS corpus and fine-tuned on *Train*, over the Baseline model in Table 2 is $21\%$. This significant improvement in perplexity is a positive indication of the effectiveness of the proposed synthetic CS pre-training framework. Additionally, we tested a model pre-trained with the original parallel corpora, *No*CS, adapted with SEAME *Monolingual* and then fine-tuned with *Train*, which only differs from the proposed fine-tuned model by the data augmentation process. Its perplexity is 223, that is a bit worse than the baseline. This shows that data augmentation is necessary and without it, the mixed-domain data will hurt the target domain model. Furthermore, the pre-trained model without fine-tuning, i.e. *Pre*CS1, can still give a perplexity of 359, indicating it to be a good prior.

We also conduct ASR experiment on the SEAME database with $101.1hrs$ of training and $11.5hrs$ of evaluation. The ASR system is set up according to [11], whereby the acoustic model is based on time-delay neural network and the language model is trigram. The best WER for the system is $25.25\%$. To show that the reduction in perplexity also translates to reduction in WER, we perform lattice rescoring using the Synthetic CS model. Our pre-trained language model, without the adaptation phase, is fine-tuned on the Train transcription used in the ASR. The WER dropped from $25.25\%$ to $23.80\%$, an absolute improvement of $1.45\%$. To take away the improvement due to RNN language model, we also perform lattice rescoring using a RNN language model without pre-training. Its best WER is $24.11\%$ which is higher than the WER using Synthetic CS model and this shows that the proposed method has practical benefit to the downstream tasks such as ASR.

## 3. Future Works

There are various cross-lingual word embedding derivation techniques and they can be classified into two major categories. One is characterized by the use of pre-trained word embeddings in their respective languages and makes use of constraints to align the two embedding spaces. Another category is noted for data augmentation and novel training strategies to derive cross-lingual embedding from the network directly. In term of aligning the embedding spaces, we note that the current works focus on finding a transformation matrix [12] and make use of a bilingual dictionary to constrain the transformation using the squared loss of the residual matrix. Such methods assume strongly the isomorphic property of the respective language structure and this assumption rarely holds, especially in distant language pairs.

While the neural network training and data augmentation methods have shown better results and such good performance is usually attributed to their latent soft-alignment. Such claims have not been extensively investigated and we cannot gain a deeper understanding of the linguistic properties captured by such embeddings. To this end, we propose to focus on part of the future works on understanding the embedding space in term of its visualization and the latent linguistic structure. By devising a set of comprehensive linguistic probes such as the ones found in [13], we hope to observe the transformation of the probe words during training and the underlying graph structure they tend to exhibit.

## 4. Acknowledgments

I would like to thank my supervisor Porf. Li Haizhou for his guidance and support. I am supported by NUS Research Scholarship.

# 5. References


[1] P. Muysken, C. P. Díaz, P. C. Muysken *et al.*, *Bilingual speech: A typology of code-mixing*. Cambridge University Press, 2000, vol. 11.

[2] Ö. Çetinoğlu, S. Schulz, and N. T. Vu, "Challenges of computational processing of code-switching," in *Proceedings of the Second Workshop on Computational Approaches to Code Switching*. Association for Computational Linguistics, 2016, pp. 1–11. [Online]. Available: http://www.aclweb.org/anthology/W16-5801

[3] P. Auer, "From codeswitching via language mixing to fused lects: Toward a dynamic typology of bilingual speech," *International journal of bilingualism*, vol. 3, no. 4, pp. 309–332, 1999.

[4] G. Lee and H. Li, "Word and class common space embedding for code-switch language modelling," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6086–6090.

[5] H. Adel, N. T. Vu, F. Kraus, T. Schlippe, H. Li, and T. Schultz, "Recurrent neural network language modeling for code switching conversational speech," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 8411–8415.

[6] H. Adel, N. T. Vu, and T. Schultz, "Combination of recurrent neural networks and factored language models for code-switching language modeling," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, vol. 2, 2013, pp. 206–211.

[7] G. I. Winata, A. Madotto, C.-S. Wu, and P. Fung, "Code-switching language modeling using syntax-aware multi-task learning," in *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*. Association for Computational Linguistics, 2018, pp. 62–67. [Online]. Available: http://aclweb.org/anthology/W18-3207

[8] G. Lee, T.-N. Ho, E.-S. Chng, and H. Li, "A review of the Mandarin-English code-switching corpus: SEAME," in *Asian Language Processing (IALP), 2017 International Conference on*. IEEE, 2017, pp. 210–213.

[9] G. Lee, X. Yue, and H. Li, "Linguistically motivated parallel data augmentation for code-switch language modeling," in *INTERSPEECH (Accepted*, 2019.

[10] J. Howard and S. Ruder, "Universal Language Model Fine-tuning for Text Classification," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2018, pp. 328–339.

[11] P. Guo, H. Xu, L. Xie, and E. S. Chng, "Study of Semi-supervised Approaches to Improving English-Mandarin Code-Switching Speech Recognition," in *Proc. Interspeech 2018*, 2018, pp. 1928–1932. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1974

[12] M. Artetxe, G. Labaka, and E. Agirre, "Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[13] A. Kuncoro, C. Dyer, J. Hale, D. Yogatama, S. Clark, and P. Blunsom, "LSTMs can learn syntax-sensitive dependencies well, but modeling structure makes them better," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 1426–1436. [Online]. Available: https://www.aclweb.org/anthology/P18-1132