Sequence Models for Computational Etymology of Borrowings

Winston Wu **Kevin Duh** **David Yarowsky**
Center for Language and Speech Processing
Department of Computer Science
Johns Hopkins University
{wswu,yarowsky}@jhu.edu,kevinduh@cs.jhu.edu

Abstract

We computationally model the processes of word borrowing from a donor word to an incorporated word, and vice versa, by answering two questions: (1) what does a word look like incorporated into another language, and in the opposite direction (2) where did a word come from? We employ neural sequence models, focusing on six specific borrowing relations: calques, partial calques, semantic loans, phono-semantic matches, transliterations, and generic borrowings. We experiment with several model variants, including LSTM encoder-decoders, copy attention, and Transformers. In both directions, we find that an LSTM model can beat strong baselines, with the quantity of data strongly influencing model performance.

1 Introduction

Words are borrowed into a language through various processes. For example, the English *internet* was incorporated into Welsh as *rhyngrwyd* (rhyng- 'between' + rhwyd 'net') through a calqueing process where each component is translated literally. In contrast, the English *chimpanzee* became the Welsh *tsimpansî* through a process of sound correspondences.

Borrowing is prevalent across the world's languages, and modeling how and from where words enter a language are interesting but understudied tasks under the umbrella of computational etymology (Wu and Yarowsky, 2020a). This is a relatively new field with many downstream applications. Perhaps the most salient is lexicon expansion: more comprehensive dictionaries will enable better communication between cultures as well as better training material for machine translation systems. Computational etymology is also important for historical linguistics, whose focus is on discovering the relationships between languages and their words. An accurate model of word borrowing can also be a boon for language preservation and revitalization, where models can help coin neologisms for modern terms.

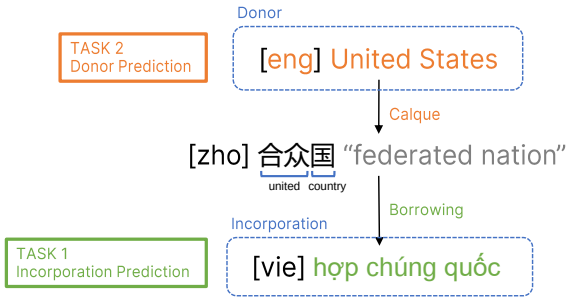

Figure 1: Two borrowing prediction tasks. The predicted output for each task is in green and orange, respectively.

Owing to recent successes of machine translation models for similar tasks (Tsvetkov and Dyer, 2015; Gorman et al., 2020; Wu and Yarowsky, 2020a,b), this paper investigates the application of neural sequence-to-sequence models for the task of etymology prediction. Specifically, we focus on word borrowings, where a word enters a language via a non-related donor language.[1] Whereas inherited words and cognates tend to follow regular sound shifts and can be modeled well with transliteration models (Beinborn et al., 2013; Wu and Yarowsky, 2018b), words borrowed from unrelated languages undergo various processes (Section 3) that may not preserve the structure or phonetics of the original word.

We propose to model borrowings in two tasks (Figure 1), motivated in Section 4. In Task 1, given a donor word and etymological relation, can we predict the form of the incorporated word in the borrowing language? In the opposite direction, in

[1] This is in contrast to other etymological relations, such as inheritance, where words enter through a related language, e.g. from Latin to French.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 4032–4037

Task 2, given the incorporated word, can we predict the donor word and language? Our experiments across several experimental scenarios on these two tasks using data from Wiktionary indicate that modeling borrowings is a challenging task with much room for future research.

2 Related Work

Though the tasks defined in this paper are new, there are several related threads of work. In the task of *cognate transliteration*, a system is trained to generate cognates in a different language (Beinborn et al., 2013; Wu and Yarowsky, 2018b). This paper uses a multilingual cognate transliteration approach applied specifically to borrowings. Similar approaches have also been applied to the task of proto-language reconstruction (Meloni et al., 2021). Related to cognate transliteration is the task of *grapheme-to-phoneme conversion*, which has a long history of research. Cognate transliteration can be viewed as G2P across languages, where the words are cognates, for example in the case of names (Waxmonsky and Reddy, 2012; Wu et al., 2018; Wu and Yarowsky, 2018a). Recently, researchers have studied massively multilingual versions of these tasks, where single (neural) models are trained on the combination of hundreds of languages (e.g. Deri and Knight, 2016; Gorman et al., 2020; Lewis et al., 2020).

3 Data

We extract etymology information from the English edition of Wiktionary using Yawipa (Wu and Yarowsky, 2020a), a recent Wiktionary parser. We focus on six specific types of borrowings (whose Wiktionary label is in monospaced font below) across a spectrum of semantic and phonetic fidelity:

- `calque`: Also called a loan translation. Components of the original word are literally translated into the target language, e.g. the English *brainwash*, from the Chinese 洗脑 *xi* 'wash' + *nao* 'brain'.
- `partial calque`: A calque where not every component is translated, e.g. the English *apple strudel*, from the German *Apfelstrudel*.
- `semantic loan`: A sense extension is borrowed onto an existing word, e.g. the French *souris* 'mouse', which borrowed the computing sense from the English *mouse*.

Lang	Count	%
eng	23,142	0.15
lat	18,713	0.12
fra	17,556	0.11
spa	7,123	0.05
ara	6,508	0.04
san	6,393	0.04
grc	6,122	0.04
deu	5,390	0.04
rus	5,109	0.03
ita	4,660	0.03

Table 1: Distribution of top 10 languages extracted from Wiktionary.

- `psm`: Phono-semantic matching. Components of the original word are replaced with phonetically and semantically similar words, e.g. 声纳 *sheng* 'sound' + *na* 'receive', from the English *sonar*.
- `transliteration`: A deterministic process of writing script conversion that seeks to preserves a word's orthography.
- `bor`: A generic borrowing category. The overwhelming majority of borrowings in Wiktionary are labeled as such. In this paper, we distinguish between `bor`, this relation as annotated in Wiktionary, and "borrowing", the word formation process encompassing these six relations.

The data we extracted consists of over 150K ground-truth annotated borrowing relationships, spanning a total of 837 languages. The top 10 languages are shown in Table 1. Note that only 101 languages have more than 100 entries, and 260 languages have more than 10 entries. In this work, we are also specifically interested in the long tail of low-resource languages. The distribution of borrowing relations is shown in Figure 2. Note the log scale, and the fact that that the majority class (`bor`) comprises 96% of the entire dataset, which motivates several of our experimental variants.

4 Tasks

We first establish our terminology for borrowings: we say etymology is directed relation between a donor word and an incorporated word.[2] We experi-

[2]We eschew the established terms "loanword" and "borrowing" because loaning and borrowing imply an obligation to return the item being borrowed. In contrast, "borrowed" words are fully incorporated into the language.

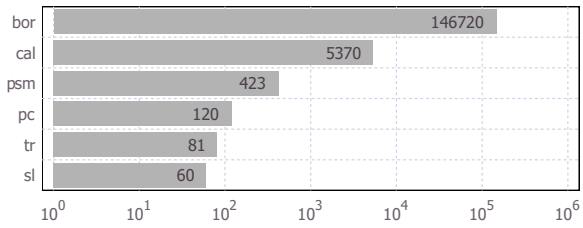

Figure 2: Distribution of borrowing relations.

ment on two tasks in etymology prediction:

4.1 Task 1: Incorporation Prediction

Given a donor word and a target language, how would the word be incorporated into that language? And by what means? This task is motivated by a real-world example: when deep learning was gaining popularity, researchers were considering how to best render the term into Japanese. Should it be a loanword and written in katakana (ディープラーニング *dīpurāningu*), or translated using a calque (深層学習 *shinsō gakushū* 'deep' + 'learning')? Besides terminology standardization, this task has applications in language revitalization and unknown word translation.

4.2 Task 2: Donor Prediction

In the opposite direction, given a word, from where and how did it come into the language? If we view Wiktionary as a directed graph, where the nodes are words and the edges are etymological relationships, there are missing edges. The task is to reconstruct these missing edges. As Wiktionary is a human-annotated resource, there is much variance in the quality and completeness of annotations, and good performance on this task can help fill in etymology even in high-resource languages like English.

5 Experiments

To tackle these two tasks, we employ character neural sequence-to-sequence models. For Task 1, predicting the incorporated word, the input is a sequence containing: the donor language, each character of the donor word, the etymological relation, and the target language. The output is the characters of the incorporated word.

In: `eng c a b b a g e bor abe`
Out: `k a b i j`

For Task 2, the input is a sequence containing the word's language and each character of the word, while the output is the donor language, donor word characters, and relation.

In: `abe k a b i j`
Out: `eng c a b b a g e bor`

For Task 1, we experiment with separate LSTM models trained for each borrowing relation (LSTM-sep), a single multi-task LSTM model trained on the combined data (LSTM), the same model trained with both the source and target data preprocessed by the unigram SentencePiece method (Kudo and Richardson, 2018) with a vocabulary size of 4000 (LSTM-spm), the same model with copy attention (See et al., 2017) (LSTM-copy), a Transformer (Vaswani et al., 2017) model (TF), and an ensembling method (Ensemble). This method is a score-based voting procedure that combines the output of the LSTM-sep, LSTM, and TF models. Each model gives 5 votes for their top prediction, 4 votes for their second place prediction, and so on (1 vote for fifth place). For each test instance, the votes are tallied up, and the prediction with the highest number of votes is the prediction of the ensemble. Ties are broken by picking the prediction with the highest model decoder score among all the models.

For Task 2, we experiment with a baseline LSTM model and the same model with copy attention.

All models were trained using the OpenNMT-py framework (Klein et al., 2020). The LSTM models are two-layer encoder-decoders with 500-dimension hidden state, trained with the ADAM optimizer. The Transformer model has a 6-layer encoder and decoder with 8 heads, trained with ADAM with learning rate scheduling. For reproducibility, we provide the training scripts which include the full model details. Accounting for the extreme imbalance in our dataset, we performed a stratified split of the dataset into a 80-10-10 train-dev-test split, where each split contains the same proportion of languages and borrowing relations.

6 Results and Analysis

6.1 Task 1

We evaluate each model on a held-out 15,288 example test set. Table 2 presents character BLEU (computed with SacreBLEU Post (2018)) as well as accuracy and character edit distance from the gold (CED). We also report 5-best results for accuracy (was the correct answer in the top 5 results?) and CED (within the top 5 results, what is the minimum edit distance to the correct answer?)

At a cursory glance, the single models trained on all the data performs slightly better compared to the separate relation-specific models, following a

Model	BLEU	Acc	CED	5Acc	5CED
LSTM-sep	53.77	20.00	2.42	33.51	1.82
LSTM	55.83	21.43	2.31	34.98	1.71
LSTM-copy	55.90	19.92	2.32	34.46	1.69
LSTM-spm	45.62	10.68	2.85	20.31	2.13
Transformer	61.30	22.19	2.06	41.54	1.43
Ensemble	60.32	25.67	2.05	49.24	1.18

Table 2: Results for Task 1. Acc is accuracy (higher is better), CED is average character edit distance (lower is better). 5 indicates 5-best results.

trend of multi-task training performing better than models trained on a single task. The Transformer model performs the best, likely due to its innovative attention mechanism that has proven successful in other tasks. However, by examining the results for each borrowing relation, we see that the successes of the models are largely on the `bor` relations. All the models perform poorly in correctly predicting any non-`bor` relations, though we find that the calque-specific model performs slightly better than the jointly trained LSTM on calques. For example, the separate calque model correctly predicted the German *vollschlank* borrowed into Dutch as *volslank*, which the LSTM model could not do. And even when it generates incorrect answers, often the predictions look like "good attempts" at calqueing. For example, the French *Pays d'en Haut* gets translated as *Land of the Roud* (correct is *upcountry*), whereas the jointly trained models often do character substitutions instead.

Copy attention (LSTM-copy), which allows the model the option to copy characters from the source, was intended to help the model with similarly spelled borrowings, but overall it did not perform as well as a simple LSTM model. The subword model (LSTM-spm) also unexpectedly did not perform well. The goal of using subwords was to encourage the model to translate larger character sequences, the idea being that translational relations such as calques would consist of two subwords rather than several individual characters. Indeed, the LSTM-spm model treats most words as calques, often translating when it should instead perform character substitutions or sound shifts. Ensembling of three models' outputs is a simple but effective method resulting in a large increase in prediction performance. The score-based voting effectively combines the strengths of individual models, especially when all models have the same word in their n-best predictions.

Error Analysis Due to the small quantities of available training data for partial calques, semantic loans, phonosemantic matches, and transliterations, the models cannot accurately learn to predict words incorporated by the aforementioned processes. This data shortage is exacerbated for the separately trained systems. Models largely treat these translational borrowings as generic `bor`s and perform character substitutions and sound shifts. This approach, exemplified by cognate transliteration systems, works for the majority of test examples, because `bor`s are essentially cognates with small edit distance. All phonosemantic matches are Chinese, so models will output Chinese characters, but due to the sparsity of the characters, the model cannot produce the correct answer. For the remainder of this analysis, we will focus on `bor` and `cal` as the main two borrowing relations. We find all the models show similar patterns of prediction, so the following examples are from the multi-task LSTM model.

In many cases, the incorporated word is similar to the donor, so the model can correctly predict the borrowing. For example, for the Latin *vanitas* borrowed into French, the model predicts *vanita*; the correct *vanité* is its second choice. The model can also handle different writing scripts. For example, it correctly predicts the Greek πυριτις borrowed into Latin as *pyritis*. Unfortunately, sound shifts do not work for the other borrowing relations, like calques, that require translation of morphemes. In many cases, the model does not seem to distinguish between non-`bor` relations and merely performs sound shifting. For example, our model predicts that the English *shopping center* calqued into Afrikaans is *schoppingsentre* (correct is *winkelsentrum*).

When encountering calques, the model sometimes recognizes that it should translate rather than transliterate. However, the lack of sufficient training data prevents the model from learning to accurately translate component morphemes. For example, our model predicts the English *download* calqued into German is *Dunnleut* (correct is *herunterladen*). Here, we see that the model picks up on the fact that German words tend to start with a capital letter, though in this case the word in question is a verb which does not need capitalization. We also find that the model cannot get the word order correct when languages have different adjective-noun ordering. For example, our model incorrectly

predicts that the French *mariage blanc* borrowed into English is *marriage mank* (correct is *white marriage*).

Broken down by language, our data contains numerous low-resource languages, many of which have just 1-10 words. Training a single model on such data for a single language would yield low performance, but our massively multilingual borrowing models can successfully handle many of these low-resource languages.

6.2 Task 2

For Task 2, we follow Wu and Yarowsky (2020a), who used an LSTM model to predict both the language and formation mechanism of a word. While they attempted to predict broader categories of inheritance vs borrowing, we focus on six specific borrowing relations. Because many borrowings have small edit distance, we also employed an LSTM model with copy attention. This model's performance was slightly worse than the baseline LSTM, a trend we also observed in Task 1. This indicates that borrowings are fundamentally different from inherited and cognate words, where copy attention models have seen good performance. Results grouped by word, language, and relation are presented in Table 3.

The models for Task 2 are inherently multi-task: they must predict the donor language, donor word, as well as the relation. As such, prediction of donor language and relation can be evaluated as classification tasks. We found that our models were able to generate valid languages and relations in 98% cases, showing that sequence-to-sequence models can also be successful in classification tasks.

We briefly analyze the errors of the LSTM model. Perhaps unsurprisingly, the model gets over 96% accuracy on predicting the relation by always guessing `bor`, the majority class. Yet it is able to beat a strong majority baseline (always predicting `bor`, the majority class). Our model is also able to successfully predict the language of the borrowing in almost half of the test instances (guessing the majority donor language, English, would only achieve 14.8% accuracy). Thus a word's language and spelling provide sufficient information for identifying how and from where it entered the language. In terms of errors, we find some instances where the model predicts a donor language that is actually related to the correct language. For example, the Dutch *tabak* is borrowed from Spanish *tabaco*, rather than our model's French *tabac*, and many Dutch words originally from English were predicted to come from German, and vice versa. We also see several words like English *specify* were predicted to come from French, but are actually from Old French. Future work can address a custom loss function that gives "partial credit" to such predictions rather than marking them as completely incorrect.

Model	Rel	Lang	Word	CED
Majority	96.0	14.8	–	–
LSTM	96.1	47.9	23.2	2.9
LSTM-Copy	96.1	47.7	20.8	3.0

Table 3: Results for Task 2: 1-best accuracy grouped by Relation, Language, and Word. CED is average character edit distance for Word prediction.

In terms of word prediction, the seemingly low accuracy of the model is not discouraging. Supported by the low character edit distance, we see many examples where the model's prediction is close enough to be recognized by a human. For example, the Chinese 阿卡拉 is borrowed from English *a cappella*, but our model predicts *acapara*, and the Jersey French *thiâtre* was predicted to be borrowed from Latin *thiatrum* (correct is *theātrum*). When providing new entries to an impoverished etymology dictionary, our prediction model can suggest possible etymology and even plausible unknown word forms, which can then be verified by a human lexicographer.

7 Conclusion

We model word borrowings from a donor to an incorporated word, and vice versa, using neural sequence models in a variety of experimental scenarios. We find that a single model trained to predict multiple types of borrowings performs better than separate models trained for each borrowing. A Transformer model performs better than an LSTM model, and a simple ensembling method results in superior performance, though the amount of training data is a limiting factor in the performance of these models. Predicting the donor language and word is a slightly easier task, where our LSTM model is able to beat a strong majority baseline. Source code for reproducing our experiments is available at https://github.com/wswu/borrowings.

References

Lisa Beinborn, Torsten Zesch, and Iryna Gurevych. 2013. Cognate production using character-based machine translation. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 883–891, Nagoya, Japan. Asian Federation of Natural Language Processing.

Aliya Deri and Kevin Knight. 2016. Grapheme-to-phoneme models for (almost) any language. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 399–408, Berlin, Germany. Association for Computational Linguistics.

Kyle Gorman, Lucas F.E. Ashby, Aaron Goyzueta, Arya McCarthy, Shijie Wu, and Daniel You. 2020. The SIGMORPHON 2020 shared task on multilingual grapheme-to-phoneme conversion. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 40–50, Online. Association for Computational Linguistics.

Guillaume Klein, François Hernandez, Vincent Nguyen, and Jean Senellart. 2020. The OpenNMT neural machine translation toolkit: 2020 edition. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 102–109, Virtual. Association for Machine Translation in the Americas.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Dylan Lewis, Winston Wu, Arya D. McCarthy, and David Yarowsky. 2020. Neural transduction for multilingual lexical translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4373–4384, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Carlo Meloni, Shauli Ravfogel, and Yoav Goldberg. 2021. Ab antiquo: Neural proto-language reconstruction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4460–4473, Online. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Yulia Tsvetkov and Chris Dyer. 2015. Lexicon stratification for translating out-of-vocabulary words. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 125–131, Beijing, China. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Sonjia Waxmonsky and Sravana Reddy. 2012. G2P conversion of proper names using word origin information. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 367–371, Montréal, Canada. Association for Computational Linguistics.

Winston Wu, Nidhi Vyas, and David Yarowsky. 2018. Creating a translation matrix of the Bible's names across 591 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Winston Wu and David Yarowsky. 2018a. A comparative study of extremely low-resource transliteration of the world's languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Winston Wu and David Yarowsky. 2018b. Creating large-scale multilingual cognate tables. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Winston Wu and David Yarowsky. 2020a. Computational etymology and word emergence. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3252–3259, Marseille, France. European Language Resources Association.

Winston Wu and David Yarowsky. 2020b. Wiktionary normalization of translations and morphological information. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4683–4692, Barcelona, Spain (Online). International Committee on Computational Linguistics.