Learning Transliteration without Parallel Data

Derick Anderson and Timothy Gillis

Spring 2019

Project Idea

- Transliterate from English to Japanese without parallel data
- Do that using recent work on word translation without parallel data

Background - Transliteration

- To represent a word in a script different from that it was originally written in
- For some script pairs it's as easy as a table lookup
  - E.x. Middle English alphabet "þe" -> Modern English alphabet "the"
- For some script pairs it requires some thinking
  - In Japanese, primarily approximate pronunciation but also consider spelling
  - Key for our project: transliterated words are written in a specific script called katakana (in modern times anyway)
  - E.x: Edinburgh -> エディンバラ, Boston -> ボストン
- Usually learning transliteration requires at least parallel data, i.e. text which means the same thing in both scripts of interest

Background - Word Translation w/o Parallel Data

Conneau et al propose the following for unsupervised word translation:

- A domain-adversarial approach to minimize $\|WX - Y\|_2$ where W is a linear mapping and X, Y are monolingual word embeddings.
- A refinement procedure where a synthetic vocabulary is constructed with the learned W to apply the Procrustes solution (i.e. min $\|WX - Y\|_2 = UV^T$, where $U\Sigma V^T = SVD(YX^T)$).
  - This can be used with ground truth data as a supervised approach as well.
- A similarity measure for word translation which mitigates hubs in nearest-neighbor graphs. In this case, target words that are translations for many source words.
- An unsupervised validation criterion for model selection: mean cosine similarity between translations of most frequent 10,000 words.

Method

- Learn word translations from Japanese to English
- Find the English translations of Japanese katakana words by translating the most frequent words in the Japanese embedding vocabulary.
- Train the transliteration model on those pairs
  - Our transliteration model is straightforward: an RNN encoder-decoder with attention
- Evaluate on katakana pairs from a dictionary Facebook Research provides

Results

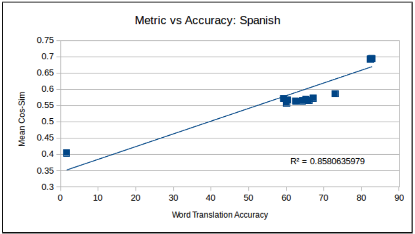
- Japanese-English word translation does not work: Purely unsupervised approach gets 0 precision at k=1, supervised approach gets 7.72 precision at k=1.
  - The domain-adversarial approach is highly sensitive to hyper-parameters.
  - The model selection metric correlates less with word translation accuracy as languages get further apart.
  - The Japanese word embedding tokens seem to be noisy to begin with.
- While Conneau et al's approach aims to mitigate hubs, we do not have sufficient resources to hold the entire vocabulary in memory, causing many words to be incorrectly translated as adverbs, prepositions, etc.
- Learning on translation pairs we generate get's us to 0.5% accuracy, adding in transfer learning pushes that to 1.2%.

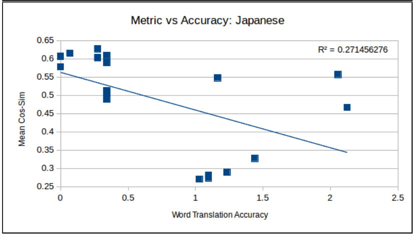
Fallback Exploration - Timothy

How well does the unsupervised metric correlate with word translation accuracy?

- Conneau et al. claim it correlates well, however they only show one example as proof without specifying the language pair.

Language	R^2	P@1
Spanish	0.8581	81.7
Korean	0.7754	13.2
French	0.7722	82.3
Chinese	0.0237	32.5
Japanese	-0.2715	0.0





Fallback Exploration - Derick

- English -> Japanese transliteration is affected by both the spelling and pronunciation of the English word
- The neural transliteration paper [1] we use as a baseline only incorporates the written form of a word
- Can we improve transliteration accuracy with a multi-task learning setup of also predicting the pronunciation of a word?
- Yes, we can: informal results suggest an about 6% accuracy jump (from the mid-forties).
- Formal results on the accuracy benefit and an analysis of the best way to do the multi-task learning are in process

[1] https://arxiv.org/abs/1610.09565

Conclusions

- Conneau et al.'s unsupervised word translation approach completely fails with the Japanese-English language pair.
- Their proposed similarity metric only correlates well with accuracy when the source/target languages are similar.
- The translation data generated from the word translation model are too noisy for our model to learn transliteration
- Multi-task learning to also predict the pronunciation of an English word improves the accuracy of English -> Japanese transliteration