Cognate Discovery to Bootstrap Lexical Resources

Shantanu Kumar - 2013EE10798
Supervisors - Prof. Sumeet Agarwal & Dr. Ashwini Vaidya

***Abstract*—The high level of linguistic diversity in South Asia poses the challenge of building lexical resources across the languages from these regions. This project is aimed at automatically discovering cognates between closely related language pairs like Hindi-Marathi, which would assist the task of lexical resource creation.**

I. Introduction

Cognates are words across different languages that are known to have originated from the same word in a common ancestral language. For example, the English word '*Night*' and the German '*Nacht*', both meaning *night* and English '*Hound*' and German '*Hund*', meaning *dog* are cognates whose origin can be traced back to Proto-Germanic [1]. Cognates are not always revealingly similar and can change substantially over time such that they do not share form similarity. Cognate information has been successfully applied to NLP tasks, like sentence alignment [2] in bitexts and improving statistical machine translation models.

II. Previous Work

Orthographic features based classifier : Hauer and Kondrak [3] use a number of basic word similarity measures as input to a SVM classifier for cognate prediction. They use features like common bigrams, longest common substring, word length difference etc.

Gap-weighted common subsequences : T. Rama [1] uses a string kernel based approach wherein he defines a vector for a word pair using all common subsequences between them and weighting the subsequence by their gaps in the strings. The subsequence based features outperform orthographic word similarity measures.

Siamese ConvNet model : In a recent work, T. Rama introduces CNN based siamese-style model [4] for the task. The model is inspired by image-similarity CNN models. By using deep learning based models, the need for external feature engineering is avoided and the system outperforms previous works for cognate detection.

For the data, we have used the IELex Database, which is an Indo-European lexical database. It contains 34,000 lexical items from 52 languages. Each entry in the table has a unique cognate class number associated with it.

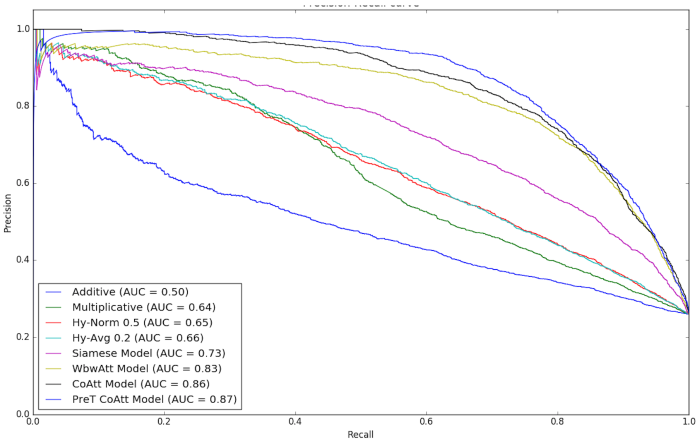
III. Models

Improving gap-weighted subsequence model : There were two fundamental drawbacks in using the common-subsequence model (Referred to as *Multiplicative* model). Firstly, since the model only looked at common subsequences, it ignored vital information about closely related subsequences such as '*fa*' in *FATHER* and '*pa*' in *PATER* which are cognate words. Secondly, this also resulted in very sparse vectors as the feature space size was huge. To overcome these problems, we tried smoothing techniques which resulted in models *Hy-Avg*, *Hy-Norm* and *Additive*.

Recurrent attention-based model : The drawback from using CNN model were also twofold. Firstly the variable length of the words is handled by clipping and padding in the network. Secondly, the character vectors used in the model are manually defined by looking at various phonetic classes and convolving over those features directly is not very intuitive. So we use a recurrent LSTM based model with attention based on the works of [5] for the cognate task (*Co-Att model*).

IV. Results

The plot below shows the PR curve for the various models. The subsequence model with maximum smoothing (*Additive*) performs the worst, however some minimum amount of smoothing helps the *Hy-Avg* and *Hy-Norm* over the original *Multiplicative* model. The LSTM based attention model (*CoAtt model*) performs significantly better than the siamese CNN model. Further improvements in the attention model are made by making the network symmetric and pre-training the attention layer weights.



V. Future Work

- We would like to perform an analysis over the performance of the Recurrent architecture and how it provides an improvement over the CNN model and Sub-sequence model. This would help to understand the underlying principles behind detecting cognate words.
- A by-product of the LSTM model is the character embeddings that are learnt simultaneously for the IPA

characters. We would like to analyze these embeddings to see if they represent any information about the different phonetic classes.

- We would like to use the model specifically to extract cognate words from Hindi-Marathi. We shall try to do this by extracting noun-pairs from aligned texts of Hindi-Marathi and testing the same on our system. The evaluation can be done manually by sampling.

References

[1] T. Rama, "Automatic cognate identification with gap-weighted string subsequences.," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, May 31–June 5, 2015 Denver, Colorado, USA*, pp. 1227–1231, 2015.

[2] M. Simard, G. F. Foster, and P. Isabelle, "Using cognates to align sentences in bilingual corpora," in *Proceedings of the 1993 conference of the Centre for Advanced Studies on Collaborative research: distributed computing-Volume 2*, pp. 1071–1082, IBM Press, 1993.

[3] B. Hauer and G. Kondrak, "Clustering semantically equivalent words into cognate sets in multilingual lists.," in *IJCNLP*, pp. 865–873, Citeseer, 2011.

[4] T. Rama, "Siamese convolutional networks based on phonetic features for cognate identification," *arXiv preprint arXiv:1605.05172*, 2016.

[5] T. Rocktäschel, E. Grefenstette, K. M. Hermann, T. Kočiskỳ, and P. Blunsom, "Reasoning about entailment with neural attention," *arXiv preprint arXiv:1509.06664*, 2015.