Reconstruction of historical word forms using Deep Learning

Mosolova A. V.[1] (a.mosolova333gmail.com), Kutuzov A. B.[2] (andreku@ifi.uio.no)

[1]Novosibirsk State University, Novosibirsk, Russia
[2]University of Oslo, Oslo, Norway

In this paper, we present an algorithm for proto-form reconstruction for several language families. The model is based on a neural architecture and it can reconstruct the proto-forms of four language families represented by corresponding datasets. The paper includes an evaluation that shows that the algorithm can perform above the baseline at the task of proto-language reconstruction for various language families. The datasets created by the authors can be used by other researchers to create and evaluate new models. Besides, the presented algorithm can be used by linguists to reconstruct forms of proto-languages of some language families.

Key words: proto-language reconstruction, deep learning, sequence-to-sequence, Indo-European language family, Semitic language family, Austronesian language family, Nakh-Daghestanian language family

Автоматическая реконструкция исторических форм слов в праязыках при помощи глубокого обучения

Мосолова А. В.[1] (a.mosolova333@gmail.com), Кутузов А. Б.[2] (andreku@ifi.uio.no)

[1] Новосибирский Государственный Университет, Новосибирск, Россия
[2] Университет Осло, Осло, Норвегия

В статье представлен алгоритм для восстановления форм праязыков некоторых языковых семей. Модель основана на нейронной архитектуре sequence-to-sequence и позволяет восстанавливать словоформы из праязыков четырех языковых семей, датасеты которых представлены в статье. В статье проведена оценка качества и показано, что архитектура справляется с задачей языковой реконструкции для различных языковых семей на уровне выше бейслайна. Созданные авторами датасеты могут быть использованы другими исследователями для создания и оценки собственных моделей. Кроме того, описанный нами алгоритм может быть использован лингвистами при работе над реконструкцией форм праязыков некоторых языковых семей.

Ключевые слова: реконструкция праязыка, глубокое обучение, sequence-to-sequence, индоевропейская языковая семья, семитская языковая семья, австронезийская языковая семья, нахско-дагестанская языковая семья

1 Introduction

The reconstruction of forms of proto-languages is one of the main tasks in comparative linguistics. In this paper we present an algorithm that would help them reconstruct the proto-forms of words existing in modern languages because it can propose a reconstructed form of a word and this variant would be either correct or require some small corrections.

This topic was mainly researched by historical linguists, while automatic methods for proto-form reconstruction are still not widely presented.

This paper is organised as follows. In the section 2 we describe previous work conducted in this field by historical and computational linguists, in the section 3 we describe the datasets. After this, the model is introduced in the section 4 and then we analyse the results of our architecture in the section 5.

2 Previous Work

Proto-form reconstruction goes back in centuries in traditional linguistics. Even researchers of XVI [17], XVII [19] and XVIII [7] centuries were trying to find the ancient ancestor of Greek, Latin, Gothic, Celtic, and Persian languages. The first comparative grammar of Indo-European languages was created by F. Bopp in 1816 [9] where he compared verb conjugation systems of Sanskrit, Greek, Latin, Persian, and German languages.

The new approach to the proto-language reconstruction was proposed by F. de Saussure in his Mémoire in 1879 [8]. It applies the knowledge of the internal structure and relations between the elements of the grammatical system of a proto-language to the reconstruction of its phonetics, morphology, and other levels.

The end of the XX century was marked by the interest of computational linguists in the field of historical linguistics and their first attempts to create automatic systems for searching the phonetic laws and proto-forms of the words [11].

One of the most prominent papers about the automatic reconstruction of proto-language is the work by A. Bouchard-Côté et al. [2] where they propose to reconstruct proto-languages while building the phylogenetic tree of a language family. Each modern language in this tree is its leaf, all inner nodes are the unattested languages and its root is a proto-language. The evolution of phonemes of all the languages in the tree is modelled by means of transducers. The parameters of a model and its outputs are updated with the EM algorithm and Monte Carlo algorithm.

The first approach to solve this task using machine learning was the system proposed in [6]. The authors used the Needleman-Wunsch algorithm for aligning words with their proto-forms and then used Conditional Random Fields (CRF) for predicting the probability of sound changes. An output was produced through the ensemble of classifiers each of which was trained to reconstruct the proto-form from its form in one of the modern languages.

Another system that used the same dataset was described in the paper by C. Meloni et al. [12]. The authors implemented a model traditionally used in machine translation for the reconstruction of Latin words using its forms in French, Portuguese, Italian, Romanian, and Spanish. This architecture consists of an encoder and a decoder both of which are recurrent neural networks (specifically, GRU). Additionally, the decoder uses the mechanism of attention. The input for this algorithm is character embeddings concatenated with the vector representations of a language.

3 Data

We collected several word lists from the 4 language families: Indo-European, Nakh-Daghestanian, Semitic and Austronesian for solving the task of proto-language reconstruction. Another source for evaluating our algorithms was the corpus of Romance languages.

The standard form of our dataset is a table the first column of which is the word of the proto-language that is to be reconstructed, and the next columns are its forms in attested languages. Below, we provide some insights into the content of each dataset; exact statistics is presented in Table 1.

Indo-European language family. 26 languages are presented in the dataset of this family. The full dataset contains 7820 examples. The models were trained separately on the languages with more than 50 words included in the dataset; there was also a model for all the languages in the dataset (about 323 languages) and a model for the most presented languages (about 26 languages).

The data for this dataset was parsed from the dictionary of J. Pokorny [16], where the forms of Proto-Indo-European are presented along with their reflections in its daughter lan-

guages. All the languages except Greek, Latin, Lithuanian, Latvian, Albanian, and Middle Irish were transliterated. The aforementioned six exceptions were presented in the original orthography.

We also employed a special dataset for the Romance languages introduced in [5] and further enriched in [12] for the comparison of our algorithm with the previous works. This dataset consists of French (3154 words), Portuguese (3442 words), Italian (4012 words), Romanian (1506 words), Spanish (3817 words), and their common ancestor, Latin. All the forms are written with graphemes and phonemes (the IPA transcription).

Nakh-Daghestanian language family. The dataset for this language family consists of three languages: Batsbi (460 words), Ingush (664 words), and Chechen (784 words). This data was parsed from "A North Caucasian etymological dictionary" by S. Nikolayev and S. Starostin [15] which could be found on the website "The Towel of Babel"[1]. All the forms from the dictionary are written with the special transcription system developed for this dictionary.

Semitic language family. The word lists for Semitic language family were parsed from "Semitic etymological dictionary" by A. Militarev and L. Kogan [14], [13] published on the website "The Towel of Babel". This dataset consists of 29 languages the words of which are written with the transcription system developed especially for the dictionary mentioned above.

Austronesian language family. This dataset contains 659 languages and it was used in the paper of A. Bouchard-Côté et al. [2]. The authors conducted some experiments for the reconstruction of Proto-Austronesian and Proto-Oceanic (there are 196 languages that were derived from Proto-Oceanic in the dataset) with the usage of the phylogenetic tree of Austronesian languages from Ethnologue [10]. All the forms in this dataset are the IPA transcriptions of the words.

Table 1: List of language families used in this work

Family	Phonetic representation type	Word number
Indo-European	orthography, transliteration	1993
Romance languages	orthography, transcription	5419
Nakh-Daghestanian	transcription	863
Semitic	transcription	2691
Austronesian	transcription	7707

4 Model

For reconstructing the proto-forms of the modern languages' words we used the sequence-to-sequence model proposed in [4] and [18]. This type of models is usually employed for solving the task of machine translation.

The algorithm consists of an encoder for transforming the input into a vector of a fixed size and a decoder that generates an output from the obtained vector. We used a GRU recurrent neural network [3] as an encoder and a GRU recurrent neural network with attention mechanism as a decoder [1].

We used sequences of characters as an input for the neural network. For this purpose, we created a dictionary of all the characters in the word list and mapped each character to a unique integer identifier. These numbers were then converted into a vector of a fixed size by means of the embedding layer that was optimised along with the neural network.

[1] https://starling.rinet.ru/

Another kind of vector representations of a character that we implemented was a concatenation of a character embedding and a language embedding. This type of embeddings was employed in the models that were used for training on the examples of all the languages in a language family. All the languages were assigned a unique integer identifier and then transformed into the vector of a fixed size which was trained along with the neural network.

5 Evaluation

5.1 Experiments

We conducted several experiments to evaluate our algorithm. We trained a model for each language that had more than 50 words derived from the list roots of those presented in the dataset (except Austronesian languages). A separate model was also trained for all the languages in one family. The following models were also trained: an additional model for predicting the Proto-Oceanic roots; a model for all the Indo-European languages in a dataset; a model for all the Indo-European languages that consist of more than 50 examples.

While training we tested the models with the following hyperparameters:

- learning rate: 0.1, 0.01, 0.001, 0.0001, 0.00001;
- size of the character and language embeddings and hidden layer: 100, 200, 300, 400.

All models were trained on 20 epochs with early stopping after 3 epochs without any improvements in the loss function on the validation data. Early stopping usually occurred after 5 epochs in the models with the hyperparameters that obstructed the generalisation process (for example, learning rate 0.00001). This was not the case for the models that were trained with the parameters that allowed the models to generalise successfully.

To validate the results, we conducted cross-validation with 3 folds for each dataset. The final result of a model was the average of these 3 folds.

5.2 Metrics

As a metric for evaluating the performance, a normalised edit distance was used, a variety of the Levenshtein distance [20]. It estimates the similarity between two sequences by counting the number of insertions, deletions, and substitutions required for obtaining the first sequence from the second one. Normalisation is calculated by dividing this result by the average length of the word in the corresponding word list. It is necessary for comparing the results across the languages.

5.3 Results

In this section, we will describe the results obtained on our datasets. As a baseline, we used a straightforward algorithm that predicted the exact copy of the given word to be its proto-form.

For the Indo-European language family, the best result (0.516) was obtained with the model trained on all the languages with more than 50 words. The learning rate of this model is 0.001, the size of a hidden layer is 300. Other models obtained the best results with the learning rate 0.01 and various sizes of a hidden layer (the results of applying different learning rates and sizes of a hidden layer are shown in the table 2 in the Appendix). All the models worked better than the proposed baseline. The size of a dataset does not influence the quality of models that were trained on a language with less than 500 examples. Training

of a model employing the dataset that consists of all the languages presented in the Pokorny dictionary does not improve the quality of a model because of the languages that contain only 1 example. The model cannot learn the information about this language, but its general quality decreases because of the presence of such languages.

The minimal normalised edit distance for the Nakh-Daghestanian language family (0.503) was acquired for Chechen with the learning rate 0.01 and the size of the hidden layer 300. For example, this model reconstructed the forms *hott* and *lām* for the words *pott* and *lam* that have *pott* and *lām* proto-forms respectively. The increase of the size of the hidden layer of other models decreased the edit distance (the results of applying different learning rates and sizes of a hidden layer are shown in the table 3 in the Appendix). A comparison against the baseline showed that the baseline model quality was higher than the quality of our model. This may be caused by the huge number of words that did not changed over time. The size of a dataset did not influence the quality of a model. Usage of a language embedding in the models for all the languages did not improve the edit distance, either.

The model for Arabic achieved the best quality (0.404) among the models for the Semitic languages. Its learning rate was equal to 0.01 and the size of the hidden layer was 400. This result could be explained by the fact that the Arabic consonant system did not change significantly through the ages. The results of applying different learning rates and sizes of a hidden layer are shown in the table 4 in the Appendix. The model trained on all the languages worked better without a language embedding, which is probably due to the fact that training of the additional embedding slows down the process of generalisation of a model. A comparison with the baseline showed that copying the original word worked better than a trained model only with some of the languages that consist of less than 300 examples.

The best result for the Austronesian language family (0.238) is obtained with the model that was trained on all the languages without a language embedding (learning rate was 0.001, the size of a hidden layer was 400). The results of applying other learning rates and sizes of a hidden layer are shown in the table 5 in the Appendix. Meanwhile, the result for this dataset presented in [2] is 0.25, but the authors of this paper were also using the phylogenetic tree of Austronesian languages while reconstructing the proto-forms. This is probably why the results presented in this paper for reconstructing the Proto-Oceanic are higher (0.125) than ours (0.348). However, all our models work significantly better than the baseline.

Romance languages achieve the minimal edit distance (0.184) using the model trained for all languages without a vector representation of language. The learning rate of the model was 0.001, the size of the hidden layer was 400. Other models obtained the best result with the learning rate 0.001 and the sizes of the hidden layer equal to 200 and 300 (the results of applying different learning rates and sizes of a hidden layer are shown in the table 6 in the Appendix). The results of all models are higher than the performance of the baseline. It is seen that with the increase in the size of the dataset the quality of the model also increased. This dataset was also used in the paper [12] and they achieved 0.102 with the model for all languages; this is twice as good as our model's quality (0.184).

6 Conclusion

In this paper, we presented a seq-to-seq model for proto-language reconstruction. It is seen from the analysis of the results that the models with the size of a hidden layer from 200 to 400 work better than those with the hidden size equal to 100. Increase of the learning size does not change the quality, the best models use learning rate 0.01 used by default with the SGD optimiser, so it is worth using this parameter for training the models for this task.

The minimal normalised edit distance achieved by this model is 0.184 and this quality allows us to say that this model can be used by a linguist during the process of reconstruction

of a proto-language because the form reconstructed by the algorithm may differ from the original one in 1 or 2 characters, so these reconstructions need almost no corrections.

We showed that the reconstruction of forms of a proto-language is possible not only for the well-studied language families but also for certain others even with a small number of examples. In comparison with the algorithms from previous works, our system outperforms some of them or work better than our baseline. The exceptions were the Semitic language family whose average quality was negatively affected by the languages with a small number of examples (Aramaic, Biblical Aramaic, Epigraphic South Arabian and Phoenician) and the Nakh-Daghestanian language family where the same problem occurred.

The datasets used for evaluation in this paper are publicly available[2] in a unified form. Therefore, other researchers can use them for creating and implementing algorithms for proto-language reconstruction and improving the quality of our models.

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[2] Alexandre Bouchard-Côté, David Hall, Thomas L Griffiths, and Dan Klein. Automated reconstruction of ancient languages using probabilistic models of sound change. *Proceedings of the National Academy of Sciences*, 110(11):4224–4229, 2013.

[3] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[4] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, 2014.

[5] Alina Maria Ciobanu and Liviu P Dinu. Automatic detection of cognates using orthographic alignment. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 99–105, 2014.

[6] Alina Maria Ciobanu and Liviu P Dinu. Ab initio: Automatic latin proto-word reconstruction. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1604–1614, 2018.

[7] Gaston-Laurent Cœurdoux. Réponse au mémoire de m. l'abbé barthélémy. *Mémoires de Littérature, tirés des Registres de l'Académie Royale des Inscriptions et Belles-Lettres*, 49:647–667, 1808.

[8] Ferdinand De Saussure. *Mémoire sur le système primitif des voyelles dans les langues indo-européennes*. BG Teubner, 1879.

[9] Bopp Franz. Über das conjugationssystem der sanskritsprache in vergleichung mit jenem der griechischen, lateinischen, persischen und germanischen sprache, 1816.

[10] M Paul Lewis. *Ethnologue: Languages of the world*. SIL international, 2009.

[2] https://github.com/anya-bel/proto_rec

[11] John B Lowe and Martine Mazaudon. The reconstruction engine: a computer implementation of the comparative method. *Computational Linguistics*, 20(3):381–417, 1994.

[12] Carlo Meloni, Shauli Ravfogel, and Yoav Goldberg. Ab antiquo: Proto-language reconstruction with rnns. *arXiv preprint arXiv:1908.02477*, 2019.

[13] A Militarev and L Kogan. Semitic etymological dictionary. vol. ii: Animal names. münster, 2005.

[14] Alexander Militarev and Leonid Kogan. Semitic etymological dictionary. vol i. anatomy of man and animals. *Münster: Ugarit-Verlag*, 2000.

[15] Sergei L Nikolayev and Sergej Anatolevič Starostin. *A North Caucasian etymological dictionary*. Asterisk Press, 1994.

[16] Julius Pokorny. *Indogermanisches etymologisches wörterbuch*, volume 2. Francke, 1969.

[17] Filippo Sassetti. *Lettere dall'India (1583-1588)*, volume 52. Salerno, 1995.

[18] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[19] Marcus Zuerius van Boxhorn. *Antwoord van Marcus Zuerius van Boxhorn, gegeven op de vraaghen, hem voorgestelt over de bediedinge van de afgodinne Nehalennia, onlancx uytghegeven: In welcke de ghemeine herkomste van der Griecken, Romeinen, ende Duytschen Tale... grondelijck ontdeckt ende verklaert worden*. by Willem Christiaens vander Boxe, 2018.

[20] Владимир Иосифович Левенштейн. Двоичные коды с исправлением выпадений, вставок и замещений символов. In *Доклады Академии наук*, volume 163, pages 845–848. Российская академия наук, 1965.

Appendix

Table 2: Average edit distance for all the languages in Indo-European language family dataset with different hyperparameters, HL - the size of a hidden layer, LR - learning rate

HL / LR	0.1	0.01	0.001	0.0001	0.00001	Baseline
100	4.991	3.572	3.943	5.658	18.52	4.505
200	8.589	3.397	3.762	5.077	16.912	4.505
300	11.55	3.429	3.702	4.677	12.647	4.505
400	13.224	**3.359**	3.713	5.006	11.257	4.505

Table 3: Average edit distance for all the languages in Nakh-Daghestanian language family dataset with different hyperparameters, HL - the size of a hidden layer, LR - learning rate

HL / LR	0.1	0.01	0.001	0.0001	0.00001	Baseline
100	4.45	2.649	3.628	3.628	3.748	**1.775**
200	13.276	2.274	3.521	3.724	4.687	1.775
300	14.753	2.131	3.343	3.712	4.987	1.775
400	15.533	2.124	3.242	3.699	4.772	1.775

Table 4: Average edit distance for all the languages in Semitic language family dataset with different hyperparameters, HL - the size of a hidden layer, LR - learning rate

HL / LR	0.1	0.01	0.001	0.0001	0.00001	Baseline
100	6.538	3.701	4.418	6.751	18.008	**3.38**
200	11.175	3.549	4.248	5.324	13.493	3.38
300	13.099	3.49	4.007	4.806	10.148	3.38
400	14.197	3.559	3.933	4.532	9.918	3.38

Table 5: Average edit distance for all the languages in Austronesian language family dataset with different hyperparameters, HL - the size of a hidden layer, LR - learning rate

HL / LR	0.1	0.01	0.001	0.0001	0.00001	Baseline
100	20.955	1.356	1.529	3.183	3.67	3.081
200	16.713	1.247	1.258	2.543	3.764	3.081
300	19.723	1.281	1.517	2.198	3.704	3.081
400	9.353	1.195	**1.164**	2.455	3.793	3.081

Table 6: Average edit distance for all the languages in Romance languages dataset with different hyperparameters, HL - the size of a hidden layer, LR - learning rate

HL / LR	0.1	0.01	0.001	0.0001	0.00001	Baseline
100	17.243	8.963	4.543	8.157	9.508	3.811
200	16.336	6.999	3.624	8.073	9.776	3.811
300	17.332	10.218	3.464	7.417	8.424	3.811
400	14.436	9.386	**3.233**	7.048	8.585	3.811