Hyperbolic Embedding for Finding Syntax in BERT

Temirlan Auyespek[1], Thomas Mach[2] and Zhenisbek Assylbekov[1]

[1]*Nazarbayev University, 53 Kabanbay Batyr Ave, Nur-Sultan, Kazakhstan*
[2]*University of Potsdam, Karl-Liebknecht-Str. 24–25, Potsdam, Germany*

Abstract

Recent advances in natural language processing have improved our understanding of what kind of linguistic knowledge is encoded in modern word representations. For example, methods for testing the ability to extract syntax trees from a language model architecture were developed by Hewitt and Manning (2019)—they project word vectors into Euclidean subspace in such a way that the corresponding *squared* Euclidean distance approximates the tree distance between words in the syntax tree. This work proposes a method for assessing whether embedding word representations in hyperbolic space can better reflect the graph structure of syntax trees. We show that the tree distance between words in a syntax tree can be approximated well by the hyperbolic distance between corresponding word vectors.

Keywords

BERT, Poincaré ball, Structural probe

1. Introduction

Recent advances in natural language processing (NLP) such as contextualized word embeddings obtained from language models [1] gave significant advancements on natural language understanding tasks. It is important to understand what kind of linguistic knowledge can be encoded in these representations. There are several works that explore specific types of linguistic knowledge, such as part-of-speech [2], morphology [3, 4], and syntax [5, 6, 7].

On one hand the paper is inspired by [5] and [7], who proposed a method for recovering syntactic dependencies under squared Euclidean distance and squared Poincaré distance respectively. In this work we propose methods for extracting syntactic dependencies under Poincaré distance *without* squaring. On the other hand the paper is motivated by the observation that one cannot draw a tree in the Euclidean space with unit distance between all neighboring nodes and without overlap, since there is not enough room for nodes, see Fig. 1 for a visualization of the problem. Mathematically, one could argue that the number of nodes in a binary tree expands faster than the Euclidean volume as the tree depth grows, i.e. there is always a k_0 such that

$$2^k > (2k)^d \quad \text{for all } k > k_0,$$

where d is the dimension of the Euclidean space.

AIxIA 2021 Discussion Papers, December 1 - 3, 2021, Milan, Italy

temirlan.auyespek@nu.edu.kz (T. Auyespek); mach@uni-potsdam.de (T. Mach); zhassylbekov@nu.edu.kz (Z. Assylbekov)
https://github.com/TemirlanAuyespek (T. Auyespek)

CEUR Workshop Proceedings (CEUR-WS.org)

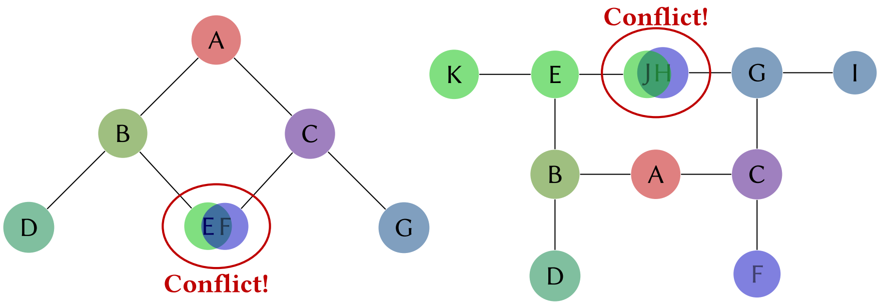


Figure 1: Trees cannot be drawn in the Euclidean space with constant distances between nodes.

2. Related work

A method, called *structural probe*, was proposed for extracting syntactic knowledge from word representations [5]. The probe identifies a linear transformation suited to use the squared Euclidean distances to represent the distance between words in the parse tree.

A second method, called *Poincaré probe*, was proposed in [7] and projects word representations into a Poincaré subspace for revealing linguistic hierarchies encoded in BERT. It can be asserted that linguistic information contained in BERT may be encoded in special metric spaces that are not necessarily Euclidean. The hyperbolic space model, in particular the Poincaré ball, is a good candidate due its tree-likeness [8, 9].

3. Methods

We start by briefly introducing hyperbolic geometry following notation from [10]. Hyperbolic geometry is a geometry with a constant negative curvature. There are five isometric models [11] and we choose the Poincaré ball as in [7]. The Poincaré ball with negative curvature 1 is defined as $\mathbb{D}^n = \left\{ \mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\|^2 < 1 \right\}$. The distance between two points $\mathbf{u}, \mathbf{v} \in \mathbb{D}^n$ is given by

$$d_{\mathbb{D}}(\mathbf{u}, \mathbf{v}) = \cosh^{-1}\left(1 + 2\frac{\|\mathbf{u} - \mathbf{v}\|^2}{(1 - \|\mathbf{u}\|^2)(1 - \|\mathbf{v}\|^2)}\right).$$

For projecting points to the Poincaré ball we consider two mappings called gnomonic mapping and hyperboloid mapping denoted by $g(\cdot)$ and $h(\cdot)$, respectively. Closed-form formulas are

$$g(\mathbf{x}) = \frac{\mathbf{x}}{\sqrt{1 + \|\mathbf{x}\|^2}}, \qquad h(\mathbf{x}) = \frac{\mathbf{x}}{1 + \sqrt{1 + \|\mathbf{x}\|^2}}$$

Both of them map points from the Euclidean space to the unit ball, which is considered as Poincaré ball. Additionally, we use the Möbius matrix-vector multiplication defined as

$$\mathbf{M} \otimes \mathbf{x} = \tanh\left(\frac{\|\mathbf{M}\mathbf{x}\|}{\|\mathbf{x}\|}\tanh^{-1}(\|\mathbf{x}\|)\right)\frac{\mathbf{M}\mathbf{x}}{\|\mathbf{M}\mathbf{x}\|},$$

which is the hyperbolic analogue of the Euclidean linear transformation.

Our method consists of three steps as in [7], but with different ways of mapping into Poincaré ball. The method is applied to word representations $\mathbf{h}_{1:t}$ obtained from one of the BERT's layers [1] for a sentence $w_{1:t}$ consisting of words $[w_1, \ldots, w_t] =: w_{1:t}$. The first step is applying a linear transformation $\mathbf{B} : \mathbb{R}^n \mapsto \mathbb{R}^m$, where n is the dimension of word representations and m is the embedding dimension. Using this step we receive a set of vectors

$$\mathbf{x}_i = \mathbf{B}\mathbf{h}_i$$

The second step is applying gnomonic or hyperboloid mapping for obtaining vector reprentations in the Poincaré ball denoted as $\mathbf{y}_i$:

$$\mathbf{y}_i = g(\mathbf{x}_i) \qquad \text{or} \qquad \mathbf{y}_i = h(\mathbf{x}_i)$$

The final step is applying Möbius matrix-vector multiplication $\mathbf{M} : \mathbb{D}^m \mapsto \mathbb{D}^m$ for obtaining final vector representations denoted as $\mathbf{z}_i$:

$$\mathbf{z}_i = \mathbf{M} \otimes \mathbf{y}_i$$

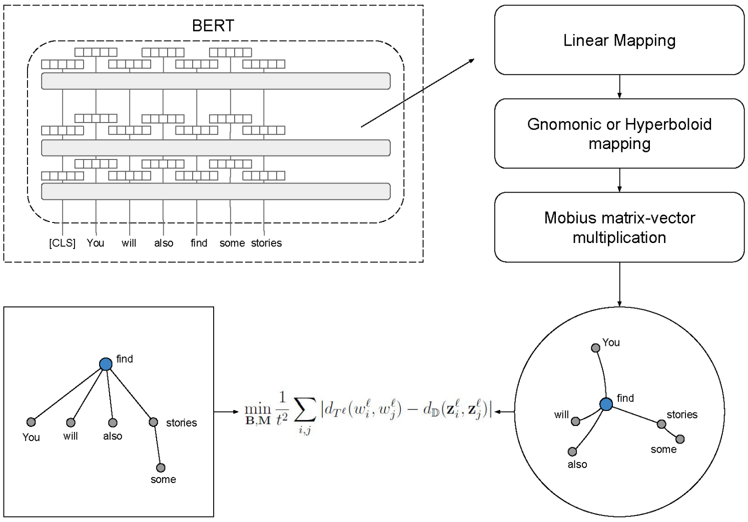

BERT
[CLS]
You
will
also
find
some
stories
Linear Mapping
Gnomonic or Hyperboloid mapping
Mobius matrix-vector multiplication
$\min_{\mathbf{B},\mathbf{M}} \frac{1}{t^2} \sum_{i,j} |d_{T^\ell}(w_i^\ell, w_j^\ell) - d_{\mathbb{D}}(\mathbf{z}_i^\ell, \mathbf{z}_j^\ell)|$
find
You
will
also
stories
some
You
find
will
stories
some
also

Figure 2: Illustration of our method.

The matrices $\mathbf{B}$ and $\mathbf{M}$ are trained in such way that the hyperbolic distance $d_{\mathbb{D}}(\mathbf{z}_i, \mathbf{z}_j)$ resembles the graph distance $d_T(w_i, w_j)$ between w_i and w_j in a syntax tree. The training objective for one sentence is

$$\ell(w_{1:t}; \mathbf{B}, \mathbf{M}) := \frac{1}{t^2} \sum_{i,j} |d_T(w_i, w_j) - d_{\mathbb{D}}(\mathbf{z}_i, \mathbf{z}_j)|. \tag{1}$$

Our approach is illustrated in Fig. 2.

4. Experiments

The main purpose of the performed experiments is to show that the usual (non-squared) Poincaré distances can encode tree distances[1].

4.1. Setup

The training objective (1) is averaged over a set of sentences (corpus) and is minimized w.r.t. $\mathbf{B}$ and $\mathbf{M}$ in the same way as in [7]. We use the Adam optimizer [12] with the learning rate 0.001. In this work we use the English Universal Dependencies dataset [13] for optimizing (1). For evaluation of the performance we report Undirected Unlabeled Attachment Score (UUAS) and average Spearman correlation (DSpr.). UUAS is the percentage of undirected edges placed correctly against the syntax tree and DSpr. is the Spearman correlation between true and predicted distances for each word in each sentence.

4.2. Results

Results for different embedding dimensionalities m, and for different layers of BERT are given in Fig. 3, where we also show the results of the Poincaré probe from [7] trained without squaring for comparison. Table 1 shows the best result per each method.

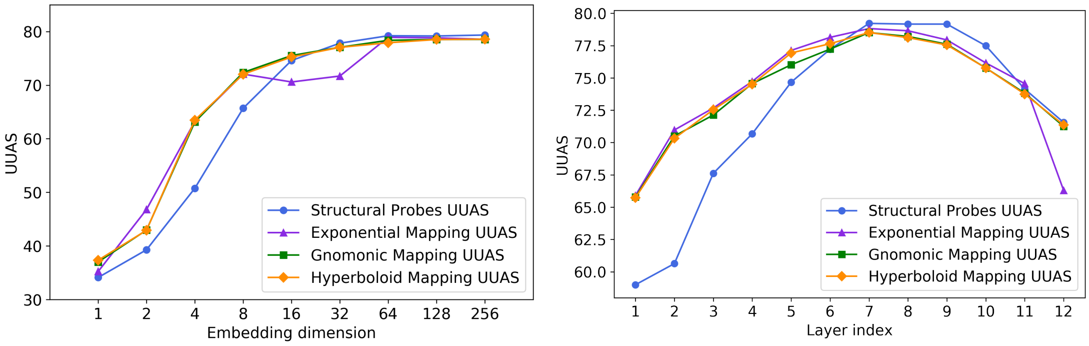


Figure 3: Probing results with respect to embedding dimensionality (left) and with respect to BERT's layer index.

Table 1
Best results per each method.

	Structural probe	Gnomonic mapping	Hyperboloid mapping	Exponential mapping
UUAS	79.17	78.51	78.51	78.82
DSpr.	80.94	83.79	83.93	83.96

We can see that the newly proposed methods with gnomonic and hyperboloid mappings gave results that are competitive to state-of-the-art. These methods outperform the structural probes

[1] Results can be reproduced at https://github.com/TemirlanAuyespek/HyperbolicEmbedding

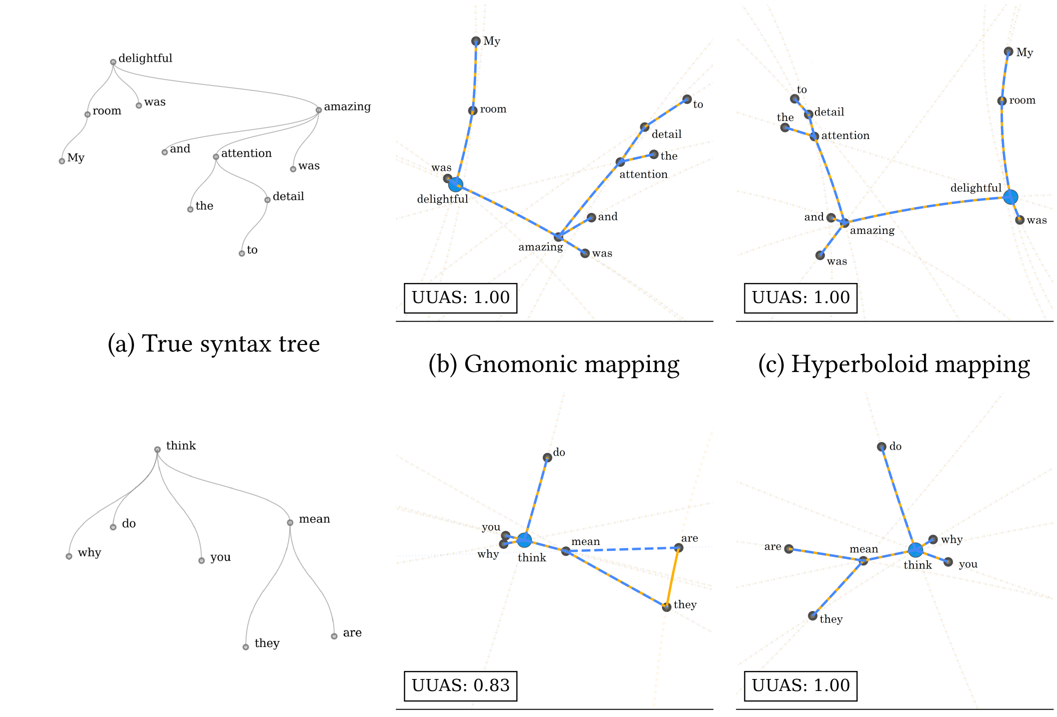
in lower dimensions. However, for dimensions higher than 16 the results become more and more similar. Results of the exponential mapping from [7] without squaring show scores similar to gnomonic and hyperboloid mappings, but there was a subsidence at embedding dimensions 16 and 32.

Fig. 3 (right) shows results of UUAS for different BERT layers with embedding dimension 128, which we consider as a trade-off between performance and computational complexity.

4.3. Visualization

We visualize recovered dependency trees in Fig. 4 using PCA projection as [7]. The results of the two methods have a similar structure, also very similar to the original syntax tree. There can be a certain level of distortion, but there is no good analogy of PCA in hyperbolic space [14].

Along with the competitive performance of our method, another important contribution of our work is the *interpretability* of the obtained Poincaré ball distances because they themselves—and not their squares—approximate syntax tree distances.



(a) True syntax tree

(b) Gnomonic mapping

(c) Hyperboloid mapping

(d) True syntax tree

(e) Gnomonic mapping

(f) Hyperboloid mapping

Figure 4: Examples of mapping using new methods. Yellow lines/geodesics show the ground truth. Blue dashed lines/geodesics show predicted dependencies and hyperbolic distances along these lines/geodesics approximate the syntax tree distances. Background gray curves are geodesics of the Poincaré ball.

5. Conclusion

In this work, we introduced two methods for embedding word representations in the hyperbolic space model, specifically the Poincaré ball. These methods were able to recover syntactic knowledge from word representation space. The obtained results are comparable to the results by Chen et al. [7] and are sometimes better. More importantly, we showed that hyperbolic distances can encode tree distances *without* any squaring. These results also confirm that the hyperbolic spaces fit tree-structured data better than the Euclidean spaces.

Future research will be dedicated to the investigation of other configurations of the BERT model such as BERT Large and extracting other kinds of linguistic knowledge.

Acknowledgements

This work is supported by the Nazarbayev University faculty-development competitive research grants program, grant number 240919FD3921. The authors would like to thank anonymous reviewers for their feedback.

References

[1] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: Proceedings of NAACL-HLT, 2019, pp. 4171–4186. URL: https://doi.org/10.18653/v1/n19-1423. doi:10.18653/v1/n19-1423.

[2] Y. Belinkov, N. Durrani, F. Dalvi, H. Sajjad, J. R. Glass, What do neural machine translation models learn about morphology?, in: Proceedings of ACL, 2017, pp. 861–872. URL: https://doi.org/10.18653/v1/P17-1080. doi:10.18653/v1/P17-1080.

[3] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations, in: Proceedings of NAACL-HLT, 2018, pp. 2227–2237. URL: https://doi.org/10.18653/v1/n18-1202. doi:10.18653/v1/n18-1202.

[4] M. E. Peters, M. Neumann, L. Zettlemoyer, W.-T. Yih, Dissecting contextual word embeddings: Architecture and representation, in: Proceedings of EMNLP, 2018, pp. 1499–1509. URL: https://doi.org/10.18653/v1/d18-1179. doi:10.18653/v1/d18-1179.

[5] J. Hewitt, C. D. Manning, A structural probe for finding syntax in word representations, in: Proceedings of NAACL-HLT, 2019, pp. 4129–4138. URL: https://doi.org/10.18653/v1/n19-1419. doi:10.18653/v1/n19-1419.

[6] T. Linzen, E. Dupoux, Y. Goldberg, Assessing the ability of lstms to learn syntax-sensitive dependencies, Trans. Assoc. Comput. Linguistics 4 (2016) 521–535. URL: https://transacl.org/ojs/index.php/tacl/article/view/972.

[7] B. Chen, Y. Fu, G. Xu, P. Xie, C. Tan, M. Chen, L. Jing, Probing BERT in hyperbolic spaces, CoRR abs/2104.03869 (2021). URL: https://arxiv.org/abs/2104.03869. arXiv:2104.03869.

[8] M. Nickel, D. Kiela, Poincaré embeddings for learning hierarchical representations, in: Proceedings of NeurIPS, 2017, pp. 6338–6347. URL: https://proceedings.neurips.cc/paper/2017/hash/59dfa2df42d9e3d41f5b02bfc32229dd-Abstract.html.

[9] R. Sarkar, Low distortion delaunay embedding of trees in hyperbolic plane, in: Graph Drawing, 2011.

[10] O. Ganea, G. Bécigneul, T. Hofmann, Hyperbolic neural networks, in: Proceedings of NeurIPS, 2018, pp. 5350–5360. URL: https://proceedings.neurips.cc/paper/2018/hash/dbab2adc8f9d078009ee3fa810bea142-Abstract.html.

[11] J. W. Cannon, W. J. Floyd, R. Kenyon, W. R. Parry, et al., Hyperbolic geometry, Flavors of geometry 31 (1997) 59–115.

[12] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: Proceedings of ICLR, 2015. URL: http://arxiv.org/abs/1412.6980.

[13] N. Silveira, T. Dozat, M. de Marneffe, S. R. Bowman, M. Connor, J. Bauer, C. D. Manning, A gold standard dependency corpus for English, in: Proceedings of LREC, 2014, pp. 2897–2904. URL: http://www.lrec-conf.org/proceedings/lrec2014/summaries/1089.html.

[14] X. Pennec, Barycentric subspace analysis on manifolds, Annals of Statistics 46 (2018) 2711–2746.