Poincaré Embeddings for Learning Hierarchical Representations

by M. Nickel and D. Kiela

presented by Natalia Skachkova and Tatiana Anikina

3 July 2019

Seminar "Embeddings for NLP and IR"
University of Saarland

Table of contents

1. Introduction
2. Poincaré Embeddings
3. Tasks and Experiments
4. Conclusion

Introduction

Motivation

Objective of embedding: place object vectors in the embedding space s.t. their proximity reflects their similarity.

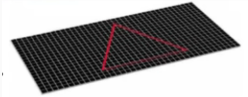
Challenge: datasets with a latent hierarchical structure (graph- and tree-structured data).

Why: embedding such vectors into traditional Euclidean space requires huge dimensionality $\Rightarrow$ computationally infeasible.

Solution: embed them into hyperbolic space, e.g. n-dimensional Poincaré ball.

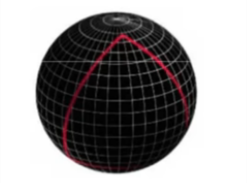
Hyperbolic Space

Euclidean space



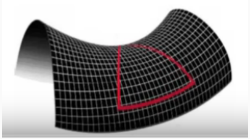
Zero curvature: The surface is flat.

Spherical space



Positive curvature: The surface curves the same way in every direction.

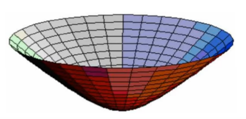
Hyperbolic space



Negative curvature: The surface curves differently depending on the direction.

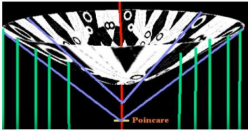
Hyperbolic Geometry

Weierstrass Model



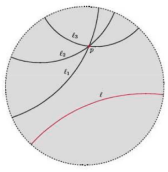
An infinite model where the entire hyperbolic space is represented on the surface of a hyperboloid.

Poincaré disk



A projection of a Weierstrass Model.

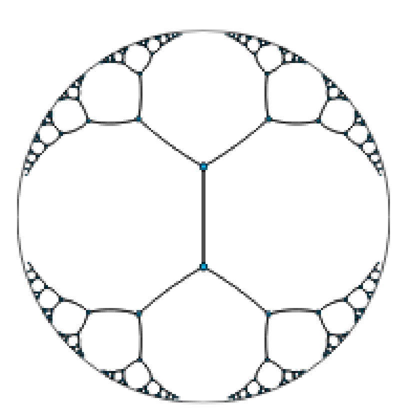
Parallel postulate



There are at least 2 lines passing through point *p* that are parallel to the line *l*.

Example: Embedding a tree into hyperbolic space

Embedding of a tree in $\mathcal{B}^2$



Perfect match:

- **Number of children** grows exponentially with their distance from the root.
- **Disk area** and **circle length** grow exponentially with the radius.

$\Rightarrow$ Trees can be thought as "discrete hyperbolic spaces"

Poincaré Embeddings

Poincaré Ball Model: Preliminaries

Poincaré Ball Model $\mathcal{B}^d$ is one of the possibilities to model hyperbolic space.

Definition: $\mathcal{B}^d = \{x \in \mathbb{R}^d \mid \|x\| < 1\}$,
where $\|x\|$ is a Euclidean norm $\sqrt{\sum_{i=1}^{d} x_i^2}$.

Distance between $u, v \in \mathcal{B}^d$: $d(u,v) = arcosh\Big(1 + 2 \cdot \frac{\|u-v\|^2}{(1-\|u\|^2)(1-\|v\|^2)}\Big)$.

We have a set of symbols (e.g. tokens) $S = \{x_i\}_{i=1}^{n}$.

We want to find **embeddings** $\Theta = \{\theta_i\}_{i=1}^{n}$, s.t. they

- capture similarities between symbols.
- infer latent hierarchy of the data.
- have feasible runtime and memory complexity.

Poincaré Ball Model: Training

Optimization problem:

Objective: $\Theta^{'} \leftarrow \arg\min_{\Theta} L(\Theta)$, where $L(\Theta)$ is a task-specific loss function.

Constraints: $\forall \theta_i \in \Theta : \|\theta_i\| < 1$.

Initialization of Θ: All embeddings are initialized randomly using uniform distribution $\mathcal{U}(-0.001, 0.001)$ close to the origin of $\mathcal{B}^d$.

Update Θ: $\theta_{t+1} \leftarrow proj(\theta_t - \eta_t \cdot \frac{(1-\|\theta_t\|^2)^2}{4} \nabla_E)$.

Euclidean gradient of $L(\theta)$: $\nabla_E = \frac{\partial L(\theta)}{\partial d(\theta,x)} \cdot \frac{\partial d(\theta,x)}{\partial \theta}$.

A projection function with $\epsilon = 10^{-5}$ is used to apply constraints.

$$proj(\theta) = \begin{cases} \theta/\|\theta\| - \epsilon & \text{if } \|\theta\| \geq 1 \\ \theta & \text{otherwise} \end{cases}$$

Tasks and Experiments

Tasks & Objectives

What do we want to test?

- **representation capacity**:
  Can we capture complex relations with small dimensionality?
- **generalization performance**:
  Can we construct parsimonious embeddings without overfitting to the training data?

Tasks & Objectives

What other methods do we use for the comparison with Poincaré embeddings?

We train different models with **3 distance measures**:

- **Poincaré distance:** $d(u, v) = arcosh\left(1 + 2 \cdot \frac{\|u-v\|^2}{(1-\|u\|^2)(1-\|v\|^2)}\right)$.
- **Euclidean distance:** $d(u, v) = \|u - v\|^2$.
- **Translational distance:** $d(u, v) = \|u - v + r\|^2$
  where r is the global translational vector learned during training.

Tasks & Objectives

Which tasks do we use for evaluation?

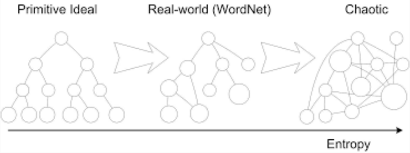
- **Embedding of Taxonomies**
- **Network Embeddings**
- **Lexical Entailment**

The tasks are performed in 2 settings:

- **Reconstruction:** Embed data then reconstruct it from the embedding.
- **Link Prediction:** Estimate the probability of inner nodes in a tree or graph.

Embedding of Taxonomies

Taxonomy is a **hierarchical system** which describes general principles of scientific classification:



In (Nickel and Kiela, 2017) Poincaré embeddings were designed to represent hierarchical data.

They worked with the *transitive closure* of the **WordNet noun hierarchy**.

Embedding of Taxonomies

What is a *transitive closure*?

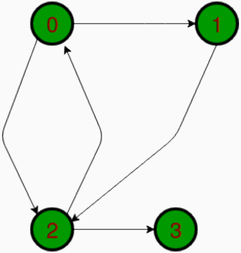
Given a directed graph G, we want to find out whether a node j is reachable from another node i for all pairs (i, j) in G.

Here *reachable* means that there exists a path from i to j.

The reachability matrix is called *transitive closure* of a graph.

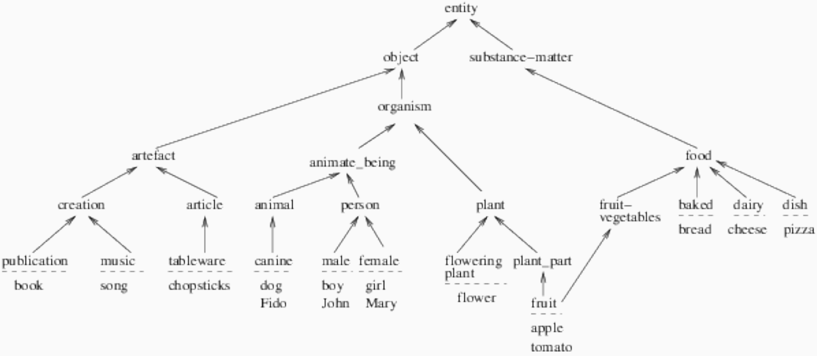
Transitive closure of G:

	0	1	2	3
0	1	1	1	1
1	1	1	1	1
2	1	1	1	1
3	0	0	0	1



Embedding of Taxonomies

The transitive closure of the WordNet noun hierarchy consists of 82,115 nouns and 743,241 hypernymy relations.



Embedding of Taxonomies

What do we want to test?

- **Reconstruction** error is used to evaluate the **representation capacity** of Poincaré embeddings.
  We embed the observed data and reconstruct them from the embeddings.

- **Link prediction** is used to test **generalization performance**.
  We split the data into train/test/validation sets and randomly hold out some observed links. The task is to predict hidden links[1].

[1] Note that we do not consider leaf and root nodes here.

Embedding of Taxonomies

Procedure:

1. Let $\mathcal{D} = \{(u, v)\}$ be the set of observed hypernymy relations between noun pairs. E.g. (*mammal, cat*)

   Learn embeddings of all symbols in $\mathcal{D}$ by minimizing the **loss function**:

   $$\mathcal{L}(\Theta) = \sum_{(u,v)\in\mathcal{D}} \log \frac{e^{-d(u,v)}}{\sum_{v'\in\mathcal{N}(u)} e^{-d(u,v')}}$$

   where $\mathcal{N}(u) = \{v \mid (u, v) \notin \mathcal{D}\} \cup \{u\}$ is the set of negative examples for u (including u). We have 10 negative samples for each positive example.

Embedding of Taxonomies

Remember that we use **3 distance measures**:

- **Poincaré distance:** $d(u, v) = arcosh\left(1 + 2 \cdot \frac{\|u-v\|^2}{(1-\|u\|^2)(1-\|v\|^2)}\right).$
- **Euclidean distance:** $d(u, v) = \|u - v\|^2.$
- **Translational distance:** $d(u, v) = \|u - v + r\|^2$

$$\mathcal{L}(\Theta) = \sum_{(u,v)\in\mathcal{D}} \log \frac{e^{-d(u,v)}}{\sum_{v'\in\mathcal{N}(u)} e^{-d(u,v')}}$$

Embedding of Taxonomies

2. **Evaluate the embeddings**.

    - For each observed relationship (u, v) rank $d(u, v)$ among the ground-truth negative examples for u: $\{d(u, v') \mid (u, v') \notin \mathcal{D}\}$. In other words: rank (*mammal, cat*) with respect to (*mammal, scissors*), (*mammal, snake*), (*mammal, door*) ...

    - Compute **the mean rank** of v and **the mean average precision (MAP)** of the ranking.

Embedding of Taxonomies

What is **MAP** (Mean Average Precision)?

Mean average precision for a set of queries is the mean of the average precision scores for each query.

Let Q define the number of queries:

$$MAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q}$$

In our case, each query q is a pair like $(mammal, X)$ with X being *cat*, *dog*, *monkey* etc.

For each query we rank the pairs including the negative samples like (*mammal*, *door*) and see how often the desired pair, e.g. (*mammal*, *cat*) appears on the first place in the ranking list.

Embedding of Taxonomies: Results

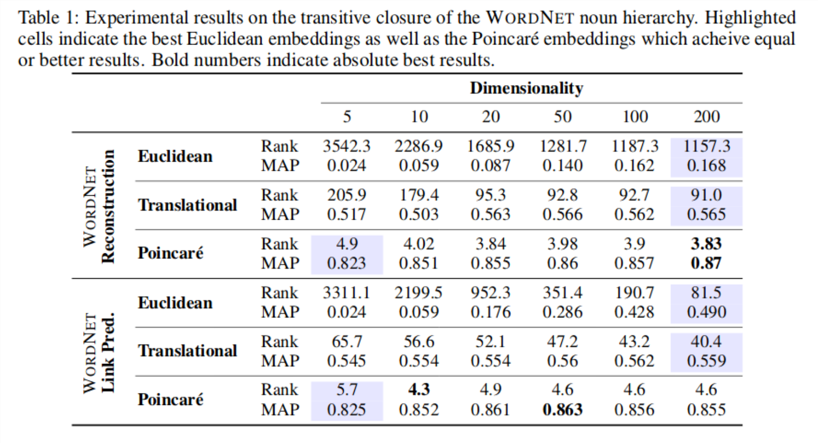


Table 1: Experimental results on the transitive closure of the WORDNET noun hierarchy. Highlighted cells indicate the best Euclidean embeddings as well as the Poincaré embeddings which acheive equal or better results. Bold numbers indicate absolute best results.

			Dimensionality					
			5	10	20	50	100	200
WORDNET Reconstruction	**Euclidean**	Rank	3542.3	2286.9	1685.9	1281.7	1187.3	1157.3
		MAP	0.024	0.059	0.087	0.140	0.162	0.168
	Translational	Rank	205.9	179.4	95.3	92.8	92.7	91.0
		MAP	0.517	0.503	0.563	0.566	0.562	0.565
	Poincaré	Rank	4.9	4.02	3.84	3.98	3.9	**3.83**
		MAP	0.823	0.851	0.855	0.86	0.857	**0.87**
WORDNET Link Pred.	**Euclidean**	Rank	3311.1	2199.5	952.3	351.4	190.7	81.5
		MAP	0.024	0.059	0.176	0.286	0.428	0.490
	Translational	Rank	65.7	56.6	52.1	47.2	43.2	40.4
		MAP	0.545	0.554	0.554	0.56	0.562	0.559
	Poincaré	Rank	5.7	**4.3**	4.9	4.6	4.6	4.6
		MAP	0.825	0.852	0.861	**0.863**	0.856	0.855

Poincaré embeddings:

- show high representation capacity and generalization performance
- are robust w.r.t. the embedding dimension

Network Embeddings

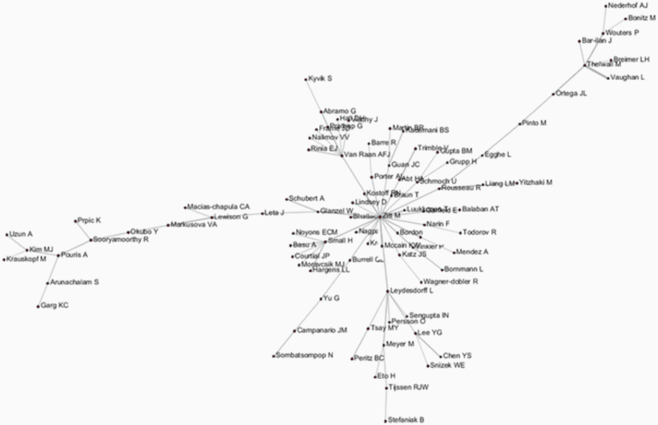
Reconstruction and link prediction in networks.

Experiments on **4 social netwoks**:

- AstroPh
- CondMat
- GrQc
- HepPh

These networks represent scientific collaborations s.t. there exists an **undirected edge** between two persons (**nodes**) if they co-authored a paper.

Network Embeddings



Author collaboration network.

Network Embeddings

Procedure:

1. Model **the probability of an edge** as

   $P((u,v)=1|\Theta)=\frac{1}{e^{(d(u,v)-r)/t}+1}$

   where Θ represents Poincaré embeddings,
   $r, t > 0$ are hyperparameters.
   r is the radius around each point u so that all points within this radius are likely to have an edge with u.
   t specifies the steepness of the logistic function, it influences clustering and degree distribution (probability distribution over node connections).

Network Embeddings

2. Split each dataset into train, validation and test sets.
   Use validation set to tune r and t.
   Learn the embeddings using **cross-entropy loss**:

   $$H(p, q) = -\sum_{x \in \mathcal{X}} p(x) \log q(x)$$

   where p and q are probability distributions over node connections.

3. **Evaluate:** compute **MAP** on the test set.

Network Embeddings: Results

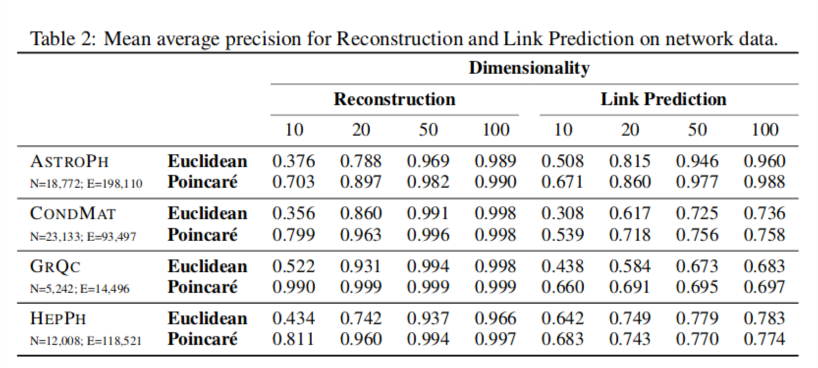


Table 2: Mean average precision for Reconstruction and Link Prediction on network data.

		Dimensionality							
		Reconstruction				**Link Prediction**			
		10	20	50	100	10	20	50	100
ASTROPH	**Euclidean**	0.376	0.788	0.969	0.989	0.508	0.815	0.946	0.960
N=18,772; E=198,110	**Poincaré**	0.703	0.897	0.982	0.990	0.671	0.860	0.977	0.988
CONDMAT	**Euclidean**	0.356	0.860	0.991	0.998	0.308	0.617	0.725	0.736
N=23,133; E=93,497	**Poincaré**	0.799	0.963	0.996	0.998	0.539	0.718	0.756	0.758
GRQC	**Euclidean**	0.522	0.931	0.994	0.998	0.438	0.584	0.673	0.683
N=5,242; E=14,496	**Poincaré**	0.990	0.999	0.999	0.999	0.660	0.691	0.695	0.697
HEPPH	**Euclidean**	0.434	0.742	0.937	0.966	0.642	0.749	0.779	0.783
N=12,008; E=118,521	**Poincaré**	0.811	0.960	0.994	0.997	0.683	0.743	0.770	0.774

Poincaré embeddings:

- outperform Euclidean embeddings
- show good results even in a low-dimensional setting

Lexical Entailment

Capture **graded lexical entailment** by quantifying to what degree X is a type of Y on a scale of [0, 10].

Experiments on the noun part of **HyperLex** (2163 rated pairs).

HyperLex example:	motorcycle / vehicle	9.85
	enemy / crocodile	0.33

Lexical Entailment

Procedure:

1. Take Poincaré embeddings with dimensionality 5 that were learned for the taxonomy embedding task.
2. Measure "is a type of" relation for each pair (u, v) with the **score function**:

$$\mathit{score}(\text{is-a}(u, v)) = -(1 + \alpha(\|v\| - \|u\|))d(u, v)$$

where $\alpha = 10^3$ is a hyperparameter
Here $\alpha(\|v\| - \|u\|)$ acts as a penalty when v is lower in the hierarchy, i.e. when v has a higher norm than u.

Lexical Entailment

3. Calculate **Spearman's rank correlation** with the ground-truth ranking.
   Spearman's rank correlation assesses how well the relationship between two variables can be described using a monotonic function.

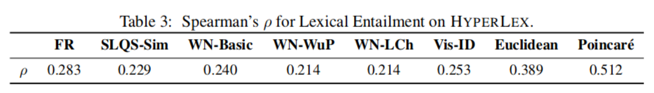


Table 3: Spearman's ρ for Lexical Entailment on HyperLex.

	FR	SLQS-Sim	WN-Basic	WN-WuP	WN-LCh	Vis-ID	Euclidean	Poincaré
ρ	0.283	0.229	0.240	0.214	0.214	0.253	0.389	0.512

The ranking based on Poincaré embeddings clearly outperformed other SOTA methods.

Conclusion

Conclusion

Poincaré embeddings:

1. are capable of inferring **latent hierarchy** and **similarities** in data.
2. outperform traditional Euclidean embeddings for the task of modeling large-scale **taxonomies** and **graph-like structures**.
3. are **compact** and **computationally cheap**.

References I

References

[1] Author collaboration. URL `journals.plos.org/plosone/article?id=10.1371/journal.pone.0187653`.

[2] Hyperbolic geometry introduction. URL `bjlkeng.github.io/posts/hyperbolic-geometry-and-poincare-embeddings/`.

[3] Transitive closure. URL `www.geeksforgeeks.org/transitive-closure-of-a-graph/`.

[4] Maximilian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 6341–6350, 2017.

References II

[5] Tiffany Sakaguchi. Hyperbolic geometry. URL https://www.youtube.com/watch?v=IMEIZdfklVA.

[6] Ivan Vulic, Daniela Gerz, Douwe Kiela, Felix Hill, and Anna Korhonen. Hyperlex: A large-scale evaluation of graded lexical entailment. *CoRR*, abs/1608.02117, 2016. URL http://dblp.uni-trier.de/db/journals/corr/corr1608.html#VulicGKHK16.