1. Comparison of GCN model when using a Euclidean Space vs. Hyperbolic Space

Group members:
Jessica Ho (jessicaho44@g.ucla.edu)
Wenqi Zou (wenqizou625@g.ucla.edu)
Rosa Garza (rgarza96@g.ucla.edu)
Brian Tagle (taglebrian@gmail.com)
Mentor: Patricia Xiao (patriciaxiao@g.ucla.edu)

2. Problem and Goal

Embeddings in euclidean spaces have been extensively studied and applied to many different machine learning tasks. In contrast, research into hyperbolic embeddings is still new, experimental, and has not been used as widely as euclidean embeddings. Hyperbolic spaces have several advantages that allow them to model hierarchical data in an embedding space with greater efficiency and simplicity than euclidean embeddings. We hypothesize that Twitter's social network graph contains some hierarchical structure that will allow it to be modeled by a hyperbolic space better than a Euclidean space. Tweets can mention other users and be retweeted by many people, so we expect the space of Tweet embeddings to grow exponentially. Therefore, tweets and their connections should be better represented by the hyperbolic space, where the volume of a ball in the space increases exponentially. Euclidean GCN already performs well on node classification and link prediction for this dataset; however, we believe that hyperbolic models can produce comparable results, especially with low embedding dimensionalities where euclidean models perform poorly. Visualizations of hyperbolic embeddings also have the benefit of clarity over euclidean embeddings. The high dimensionality of most euclidean embeddings causes it to become an uninterpretable mass in 2D. However, hyperbolic embeddings need relatively low dimensionality to accurately fit the data and the manifolds used to model hyperbolic space are able to be visualized clearly as a circle in 2D, with the most significant nodes closer to the center.

3. Data Plan

Our dataset will be on politicians' 2019/2020 tweets, either provided by Patricia's repo (sans text data) or pulled fresh from Twitter's API (sans likes). We have not settled on the final dataset yet because we are still investigating what data will work best for each model (Euclidean GCN and HGCN). For instance, the vanilla GCN takes the one hot encoding of CORA content while Patricia's model doesn't consider text data. The final Twitter dataset will include mentions and retweets relationships. The nodes of the graph will be tweets and the links of the graph will be mentions and retweets. The nodes' features will include the tweet's author id/username and the actual text of the tweet. We can get each user's political party label from Patricia's repo.

4. Solution Plan

We will implement a Euclidean GCN that takes in the Twitter dataset to generate some baseline metrics in downstream tasks described in Section 5. For this part, we'll tweak the author of GCN's repo to do said downstream tasks by using portions of Patricia's multi-relational GCN. Then we'll need to compare it with a multi-relational HGCN, which we can alter in a similar fashion to perform our desired downstream tasks.

The major contributions of these models will be the comparison of:

a. embedding visualizations,
b. distances between nodes,
c. and downstream tasks' accuracy scores (Section 5)

Between the two models. The main differentiating them will be the embedding method so the aforementioned outputs will tell us how well the data fits each embedding space.

5. Evaluation Plan

One of our end goals will be to generate political polarity scores or party classifications for Twitter users. We could also predict links (retweets, mentions, and maybe likes, depending on which dataset we choose) between nodes. The subsequent accuracy scores can then be compared.

We also plan to compare visualizations between the two embedding methods to see which is more interpretable and which accounts for the increasing space size of the dataset better. We can use t-SNE to visualize Euclidean data. We are currently investigating two options to visualize hyperbolic embeddings. The first option is to train a two dimensional hyperbolic embedding to visualize. The visualization should still be clear due to hyperbolic embeddings effectiveness at low dimensions. The second option is CO-SNE, a new method proposed in a recent research paper for visualizing hyperbolic data. However since the paper is recent, implementations of CO-SNE are not currently available.

Finally, we'll generate statistics on distances between nodes in either embedding space. These might be tricky to compare since the definition of distance is different in either space, but they may still be illuminating. We can also analyze these distances to see whether HGCNs overcome the over-smoothing problem that deep Euclidean GCNs (more than 3 layers) traditionally find challenging. It's worth noting that deep Euclidean GCNs can ostensibly easily learn to anti-smooth during training and training time can be decreased via mean-subtraction, as discussed in this paper. If we have time, we can get our Euclidean GCNs to adopt mean-subtraction and again compare its performance to HGCNs.

6. Schedule

Week 4:
- Everyone: Get used to vanilla Euclidean GCNs

Week 5:
- Brian + Brooke: Looking into Patricia's code to see what we can take for our models
- Jessica + Rosa: Dipping feet into HGCN and try to apply HGCN on Twitter dataset

Week 6:
- Have our finished Euclidean and Hyperbolic GCNs perform node classification and link prediction.

Week 7:
- Create visualizations for Euclidean and Hyperbolic embeddings.

Week 8:
- Last minute fixups.

Week 9:
- Write the report and presentation.

7. References

Chami, Ines, et al. "Hyperbolic graph convolutional neural networks." *Advances in neural information processing systems* 32 (2019).

Guo, Yunhui, Haoran Guo, and Stella Yu. "CO-SNE: Dimensionality Reduction and Visualization for Hyperbolic Data." *arXiv preprint arXiv:2111.15037* (2021).

Kipf, Thomas. "Graph convolutional networks." *Tkipf. github. io [online], Uploaded on Sep* 30 (2016).

Nickel, Maximillian, and Douwe Kiela. "Poincaré embeddings for learning hierarchical representations." *Advances in neural information processing systems* 30 (2017).

Xiao, Zhiping, et al. "TIMME: Twitter ideology-detection via multi-task multi-relational embedding." *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2020.

Yang, Chaoqi, et al. “Revisiting Oversmoothing in Deep GCNs.” arXiv *preprint arXiv:2003.13663* (2020).