Accelerating t-SNE using a uniform grid-based approximation

Project: Approximating Nearest Neighbours in Hyperbolic Space

Milan Otten

June 24, 2024

1 Introduction

- Previous work [1] has established a way of embedding high-dimensional data into hyperbolic space for visualisation purposes.
- This is done using t-SNE: an algorithm that groups neighbours in lower-dimensional embeddings by pulling points that are neighbours in the high-dimensional data closer together.
- We explore a novel way of optimising this algorithm by the use of a uniform grid on the data.
- As opposed to the previously used quadtree, the uniform grid offers better runtime complexity.

2 Hyperbolic Space

- We use the Poincaré disk model for hyperbolic space, which is a unit disk.
- Hyperbolic space has the unique property of expanding towards the edges of the disk.
- This makes it a good candidate for embedding arbitrarily sized data, such as trees.

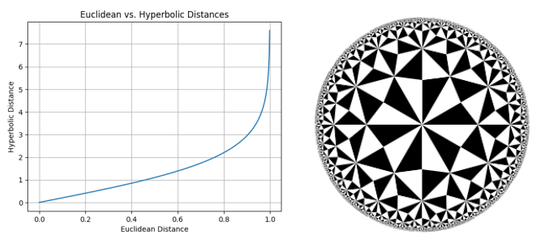

Figure: Left: Comparison of Euclidean distance from the centre of the disk compared to the hyperbolic distance at the same point. The hyperbolic distance would grow to infinity, but the Euclidean distance is stopped at 0.9999
Right: Visualisation of the hyperbolic space using a constant pattern. Source: Weisstein, Eric W. "Poincaré Hyperbolic Disk." From *MathWorld*–A Wolfram Web Resource `https://mathworld.wolfram.com/PoincareHyperbolicDisk.html`.

3 Uniform Grid

- The uniform grid splits up the space of the disk into equal-sized rectangles.
- The geometric mean of the points in a grid cell is used to approximate the underlying points.
- The gradient forces for t-SNE are then only calculated from each point to each grid cell.
- The uniform grid can be built and used in $O(m \cdot n)$ with n points and m grid cells. The quadtree solution did this in $O(n \log n)$.
- An optimisation is that the grid fits to the points, to keep as few grid cells as possible empty, which increases accuracy. See the figure below.
- Advantage: highly parallelizable.
- Disadvantage: no exact calculation.

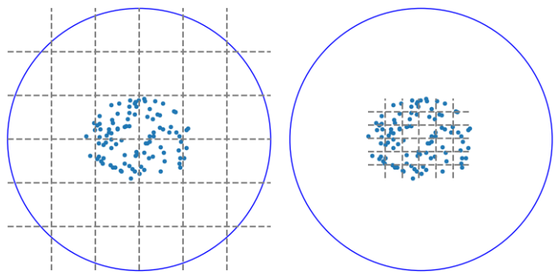

Figure: A visualisation of the uniform grid on some data. Left: illustration without the aforementioned optimisation. Right: with the optimisation.

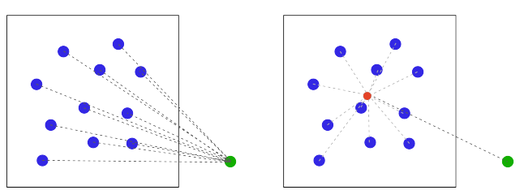

Figure: An illustration of the summarisation performed by Barnes-Hut, during the calculation of the forces acting on the green point. Left is the exact method, right is the approximation.

4 Experiments and Results

- We compare our solution with the previous solution with different datasets.
- Each algorithm is run 3 times at different dataset sizes.
- Our new algorithm provides better runtime performance and accuracy than the previous solution.
- At a set grid resolution, our algorithm is linear in the number of points, while the previous quadtree solution is log-linear.

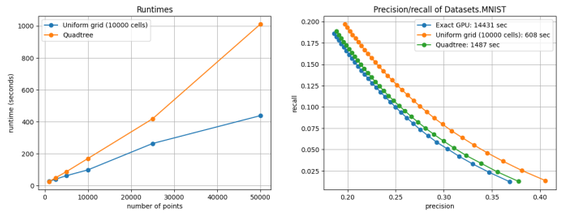

Figure: Left: Runtimes of *Uniform grid* vs. *Quadtree* (previous solution).
Right: Precision-recall graph of MNIST at 70,000 points.

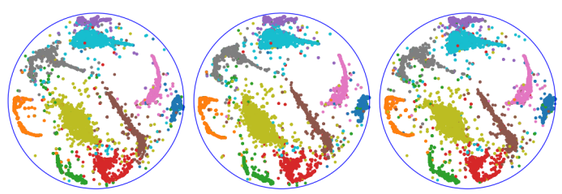

Figure: Embeddings of the MNIST dataset (70,000 data points) for *Exact*, *Quadtree*, *Uniform grid* respectively.

	Runtime (s)		
Algorithm	10,000 points	25,000 points	50,000 points
Exact	318	2,335	7,875
Quadtree	169	418	1,011
Uniform grid	98	263	438

Table: Runtimes in seconds of embeddings of MNIST by different algorithms.

Choice of Grid Resolution

- The grid resolution provides a trade-off between runtime and accuracy.
- We experimentally found the optimal size for our datasets.
- 10,000 - 15,000 is optimal for all tested datasets.

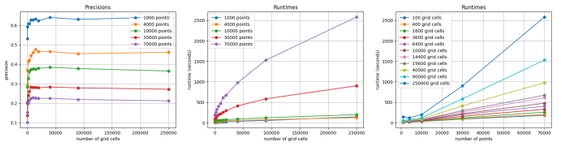

Figure: Left: precisions of MNIST embeddings for different point counts. Centre: Runtimes of MNIST embeddings at different point counts. Right: Runtimes of MNIST embeddings at different grid resolutions.

5 Conclusion

- The uniform grid is a viable alternative to the quadtree for accelerating t-SNE.
- Our proposed solution can offer better runtime performance and accuracy than previous solutions.
- Our proposed solution offers a trade-off between performance and accuracy by controlling the grid resolution.

References

[1] M. Skrodzki, H. van Geffen, N. F. Chaves-de-Plaza, T. Höllt, E. Eisemann, and K. Hildebrandt. *Accelerating hyperbolic t-SNE*. 2024. arXiv: `2401.13708 [cs.HC]`.

Github

The source code for this project can be found here: `https://github.com/Milan7843/hyperbolic-tsne`.

TU Delft Delft University of Technology