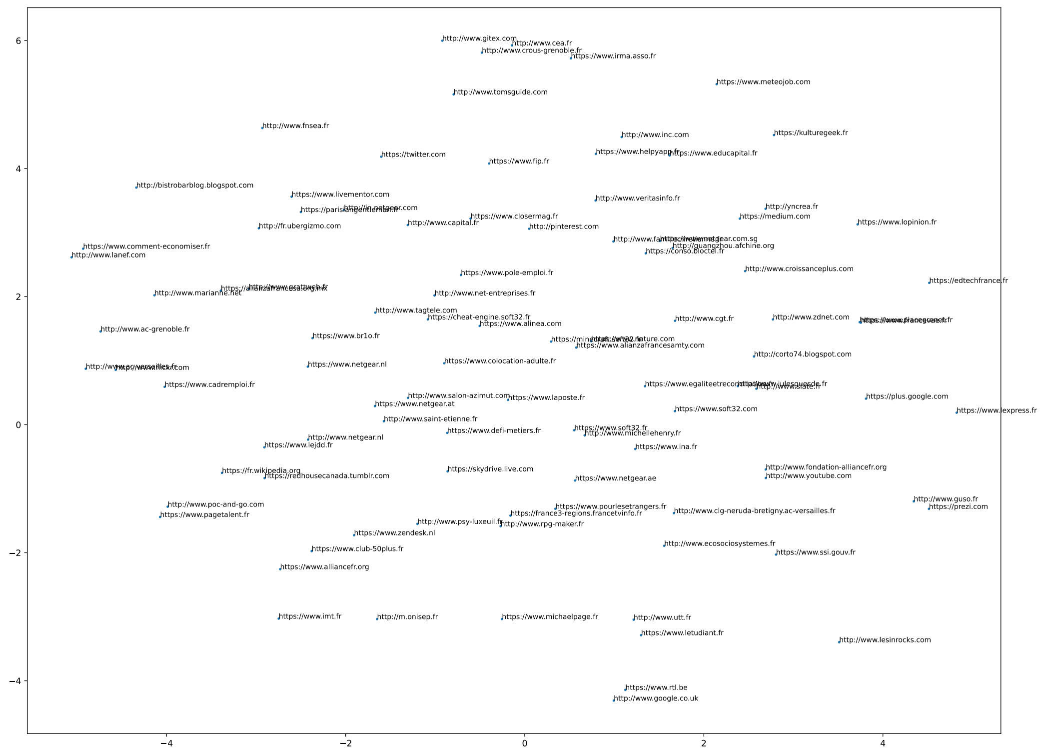
Matthieu Boyer
matthieu.boyer@ens.fr



# 1 Task 4

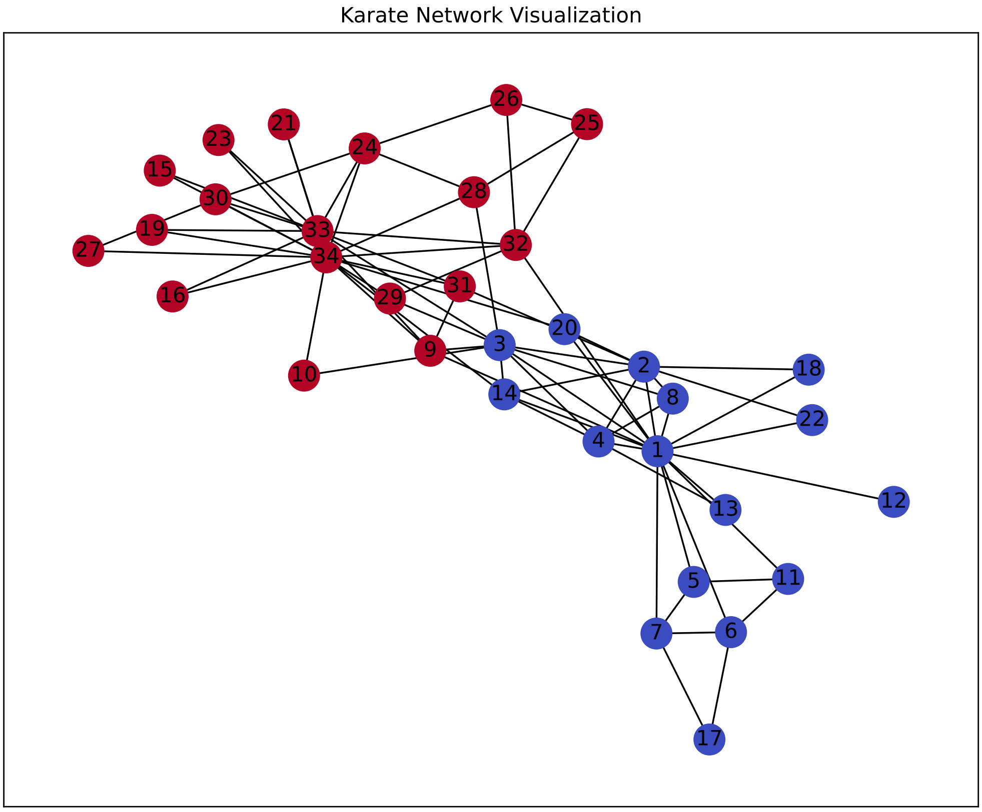
t-SNE visualization of node embeddings



# 2 Question 1

As linguists have the traditional maxim you shall know a work by its company, here, given a random walk (which can be seen as a (near-)randomly generated sequence of words), we increase the context considered to understand the meaning (or in this case the probability) that a word (vertex) is generated. The window size maps to the size of the context considered to understand what the meaning of a vertex is: what the probability of it filling in the random walk is.

# 3 Task 5




# 4 Task 8

- DeepWalk Embeddings (using `gensim.Word2Vec`) yield 42.86% accuracy.
- Sklearn's Spectral Embeddings (using `sklearn.manifold.SpectralEmbedding`) yield the same 42.86% accuracy.
- Lrw Spectral Embeddings yield 85.71% accuracy.

# 5 Question 2

The two embeddings provided show an important similarity: $X_1$ and $X_2$ define the exact same space as the second column of $X_1$ is the opposite of the second column of $X_2$. What this means is that $X_2 = X_1 R$ where $R$ is the reflection around the $y$-axis $\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$

# 6 Question 3

Since $P$ is a permutation matrix, $P_{ij} \in \{0, 1\}$. Moreover, for a permutation, each row and column has exactly one 1. The degree of node $i$ in the permuted graph equals the degree of the node that was mapped to position $i$. Therefore:

$$\tilde{D}' = P\tilde{D}P^T$$

Now we can compute the normalized adjacency matrix $\hat{A}'$:

$$\begin{aligned}
\hat{A}' &= (\tilde{D}')^{-\frac{1}{2}}\tilde{A}'(\tilde{D}')^{-\frac{1}{2}} \\
&= (P\tilde{D}P^T)^{-\frac{1}{2}}(P\tilde{A}P^T)(P\tilde{D}P^T)^{-\frac{1}{2}} \\
&= P\tilde{D}^{-\frac{1}{2}}P^T \cdot P\tilde{A}P^T \cdot P\tilde{D}^{-\frac{1}{2}}P^T \\
&= P\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}P^T \\
&= P\hat{A}P^T
\end{aligned}$$

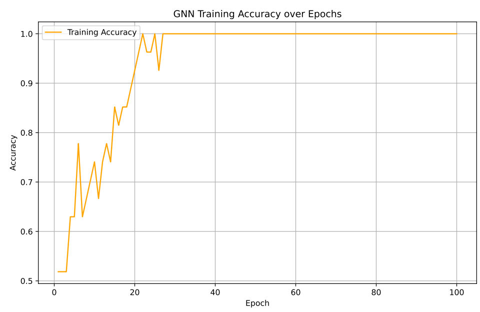
Finally, we compute the GNN output for the permuted inputs:

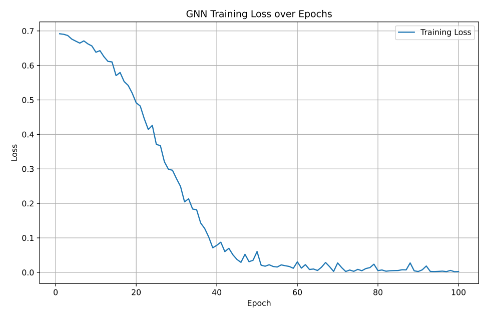
$$\begin{aligned}
\mathrm{GNN}(PAP^T, PX) &= f(\hat{A}'(PX)W) \\
&= f(P\hat{A}P^T \cdot PX \cdot W) \\
&= f(P\hat{A}(P^TP)XW) \\
&= f(P\hat{A}XW) \quad (\text{since } P^TP = I) \\
&= P \cdot f(\hat{A}XW) \quad (\text{since } f \text{ is applied element-wise}) \\
&= P \cdot \mathrm{GNN}(A, X)
\end{aligned}$$

Therefore, the GNN layer is permutation equivariant. □

# 7 Task 11

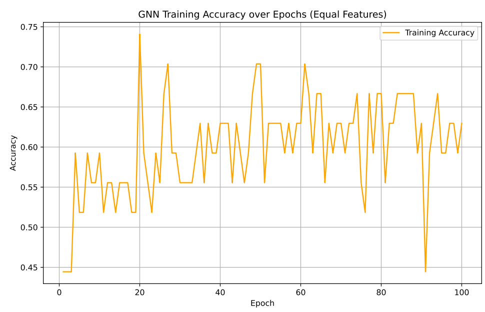
Test set results: loss= 0.0002 accuracy= 1.0000

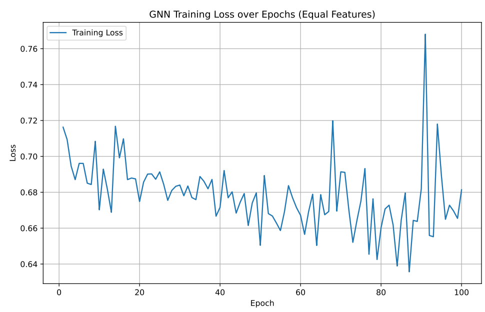



# 8 Task 12

Test set results: loss= 0.8166 accuracy= 0.2857




# 9 Question 4

We analyze a linear GCN with $k$ layers defined as:

$$Z^{(k)} = \hat{A}^k XW$$

where $\hat{A} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$ is the normalized adjacency matrix with self-loops, and the graph is connected and non-bipartite.

## Part 1: Proving $u$ is an eigenvector of $\hat{A}$ with eigenvalue $\lambda = 1$

Let $u \in \mathbb{R}^n$ be a vector where $u_i = \sqrt{\tilde{d}_i}$ (the square root of the degree of node $i$ including the self-loop). We compute $\hat{A}u$:

$$\begin{aligned}
(\hat{A}u)_i &= \sum_{j=1}^{n} \hat{A}_{ij} u_j \\
&= \sum_{j=1}^{n} \left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}\right)_{ij} u_j \\
&= \sum_{j=1}^{n} \frac{1}{\sqrt{\tilde{d}_i}} \tilde{A}_{ij} \frac{1}{\sqrt{\tilde{d}_j}} u_j \\
&= \sum_{j=1}^{n} \frac{1}{\sqrt{\tilde{d}_i}} \tilde{A}_{ij} \frac{1}{\sqrt{\tilde{d}_j}} \sqrt{\tilde{d}_j} \\
&= \sum_{j=1}^{n} \frac{1}{\sqrt{\tilde{d}_i}} \tilde{A}_{ij} \\
&= \frac{1}{\sqrt{\tilde{d}_i}} \sum_{j=1}^{n} \tilde{A}_{ij} \\
&= \frac{1}{\sqrt{\tilde{d}_i}} \cdot \tilde{d}_i \quad \text{(by definition of degree)} \\
&= \sqrt{\tilde{d}_i} \\
&= u_i
\end{aligned}$$

Therefore, $\hat{A}u = u$, which proves that $u$ is an eigenvector of $\hat{A}$ corresponding to eigenvalue $\lambda = 1$. $\square$

## Part 2: Deriving the limit of $Z^{(k)}$ as $k \to \infty$

Since $\hat{A}$ is symmetric, it has a spectral decomposition:

$$\hat{A} = \sum_{i=1}^{n} \lambda_i v_i v_i^T$$

where $\lambda_1 = 1 > |\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_n|$ and $\{v_1, v_2, \ldots, v_n\}$ are orthonormal eigenvectors.
From Part 1, we know that $v_1 = \frac{u}{\|u\|} = \frac{u}{\sqrt{\sum_i \tilde{d}_i}}$ is the eigenvector corresponding to $\lambda_1 = 1$.

Taking powers of $\hat{A}$:

$$\begin{aligned}
\hat{A}^k &= \sum_{i=1}^{n} \lambda_i^k v_i v_i^T \\
&= \lambda_1^k v_1 v_1^T + \sum_{i=2}^{n} \lambda_i^k v_i v_i^T \\
&= v_1 v_1^T + \sum_{i=2}^{n} \lambda_i^k v_i v_i^T
\end{aligned}$$

Since $|\lambda_i| < 1$ for $i \geq 2$, we have $\lambda_i^k \to 0$ as $k \to \infty$. Therefore:

$$\lim_{k\to\infty} \hat{A}^k = v_1 v_1^T$$

The node representations become:

$$\begin{aligned}
\lim_{k\to\infty} Z^{(k)} &= \lim_{k\to\infty} \hat{A}^k X W \\
&= v_1 v_1^T X W \\
&= v_1 (v_1^T X) W
\end{aligned}$$

Note that $v_1^T X$ is a row vector (representing a global aggregation of features weighted by normalized degrees), and the result $v_1(v_1^T X)W$ assigns the same linear combination to every node, scaled by the normalized degree component $v_{1,i} = \frac{\sqrt{\tilde{d}_i}}{\sqrt{\sum_j \tilde{d}_j}}$.

More explicitly, the $i$-th row of $Z^{(k)}$ converges to:

$$Z_i^{(k)} \to \frac{\sqrt{\tilde{d}_i}}{\sqrt{\sum_j \tilde{d}_j}} \cdot (v_1^T X)W$$

This means all node representations become proportional to $v_1$, differing only by their degree-dependent scaling factor.

### Part 3: Explanation of failure to distinguish nodes

Nodes with the same degree have identical entries in $v_1$, so their representations converge to exactly the same vector regardless of their initial features $X$, causing complete oversmoothing.

## 10 Task 13

Test set results: loss= 0.6081 accuracy= 0.8376