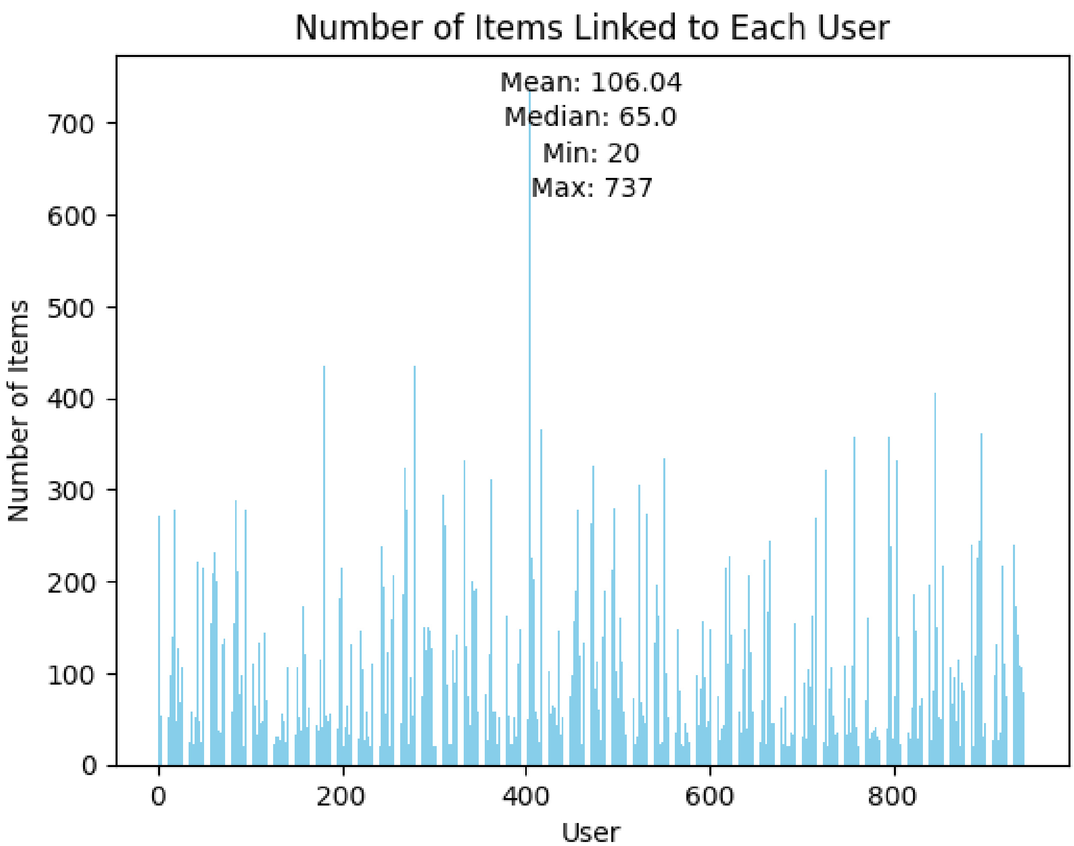
Final Report

Data exploration

With each of 943 user interacting with at least 20 items from 1682 we can rely on collaborative filtering to learn latent features of users and items



Solution exploration

Nomenclature and overview:

- Explicit feedback
- Implicit feedback
- GNN
- Content-Based approaches
- Collaborative Filtering Approaches
- Hybrid Approaches
- Matrix Factorization
  - latent factors
- GNN vs Matrix Factorization
  - GNN are able to aggregate multi-hop neighborhoods
  - Matrix representation use only direct connections
- Supervised learning on graphs
  - Labels come from external sources (predict ratings of an interaction)
  - RMSE loss
- Self-supervised learning on graphs
  - Signals come from graphs themselves (predict if two nodes are connected)

  - BPR (Bayesian Personalized Ranking) loss
- Graph representation
  - Adjacency matrix
  - COO format - a memory efficient approach to store sparse matrices
- Adjacency matrix from bipartite graph

Two approaches that we can consider for the task

- Collaborative filtering using matrix factorization
- GNN using LightGCN

Let's use supervised LightGCN in order to solve this problem. The supervised version can be classified as a collaborative filtering approach since it uses only the interactions between users and items, without the consideration of metadata.

For a given graph structure the model will try to predict a rating that the user would give to every item that they have an edge with.

Training process

- Loss:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y})^2}$$

- Optimizer: Adam
- ITERATIONS = 1_000
- EPOCHS = 10
- BATCH_SIZE = 1024
- LR = 1e-3
- ITERS_PER_EVAL = 200
- ITERS_PER_LR_DECAY = 200
- K = 10
- LAMBDA = 1e-6

Evaluation

The evaluation uses recall and precision. By top-k highest predicted item's per user.

TODO:

- implement dataloader to train in batches
- save weight
- evaluate