

Introduction

Identifying **super-spreaders: individuals who can trigger large-scale information diffusion** is a key problem in network science and influence maximisation. Traditionally, this task has been studied in the context of single-layer networks. However, real-world systems often involve multiple types of interactions, such as physical or digital communication, which are better captured by multilayer networks. **In this work, we focus on identifying agents which, when used as a single source of diffusion, enable the most effective propagation of information within multilayer networks.**

Contribution

- An inductive GNN for super-spreader identification in multilayer networks: *TopSpreadersNetwork*, capable of generalising to previously unseen graphs;
- The *TopSpreadersDataset*, derived from simulations on synthetic and real-world networks, capturing the spreading potential of individual actors;
- Alternative data transformation and embedding fusion methods, enhancing robustness in the ranking prediction task from heterogeneous data.

Problem

Definition. The task of identifying the top- k spreaders in a network M , given a ground-truth ranking \mathbf{R} , can be formulated as finding a function ϕ that predicts $\hat{\mathbf{R}}$, such that the cumulative spreading potential score (y_{rel}) of the identified top- k super-spreaders is as close as possible to that of the ground truth:

$$\phi(M) \rightarrow \hat{\mathbf{R}} : \min |1 - y_{rel}(\mathbf{R}, \hat{\mathbf{R}}, k)|$$

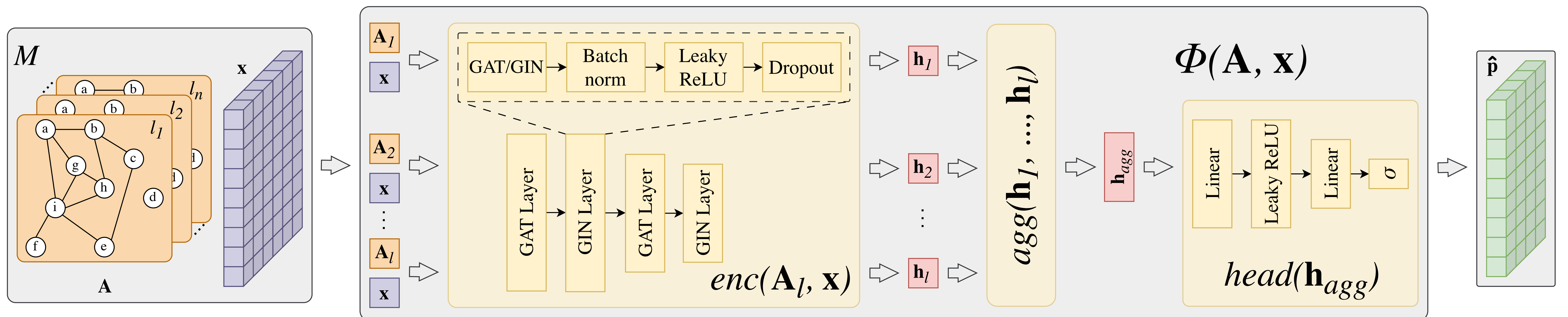


Figure 1. A schematic illustration of the *ts-net* architecture processing a multilayer network M . Each layer is encoded independently using a shared encoder composed of interleaved GAT and GIN blocks. The resulting representations are then aggregated by a trainable aggregation layer to produce actor embeddings. Finally, a vector of spreading potential is generated using a sequence of MLP layers.

Proposed model

The *TopSpreadersNetwork* is built upon three main components:

- a shared encoder that produces actor embeddings for each layer of M ;
- a trainable aggregation module that combines these representations to derive final actor embeddings;
- a prediction head that yields the output vector $\hat{\mathbf{p}}$ for each actor.

The strength of *TopSpreadersNetwork* lies in its task-specific design. **Actor features are zeroed out** to emphasise network structure. **Nonlinear label transformations are applied** to highlight high-impact spreaders during training. **A customised soft attention mechanism adaptively merges embeddings across layers**, capturing each actor's varying role in the network. To handle large graphs and to learn from local substructures, **the model uses neighbour sampling**.

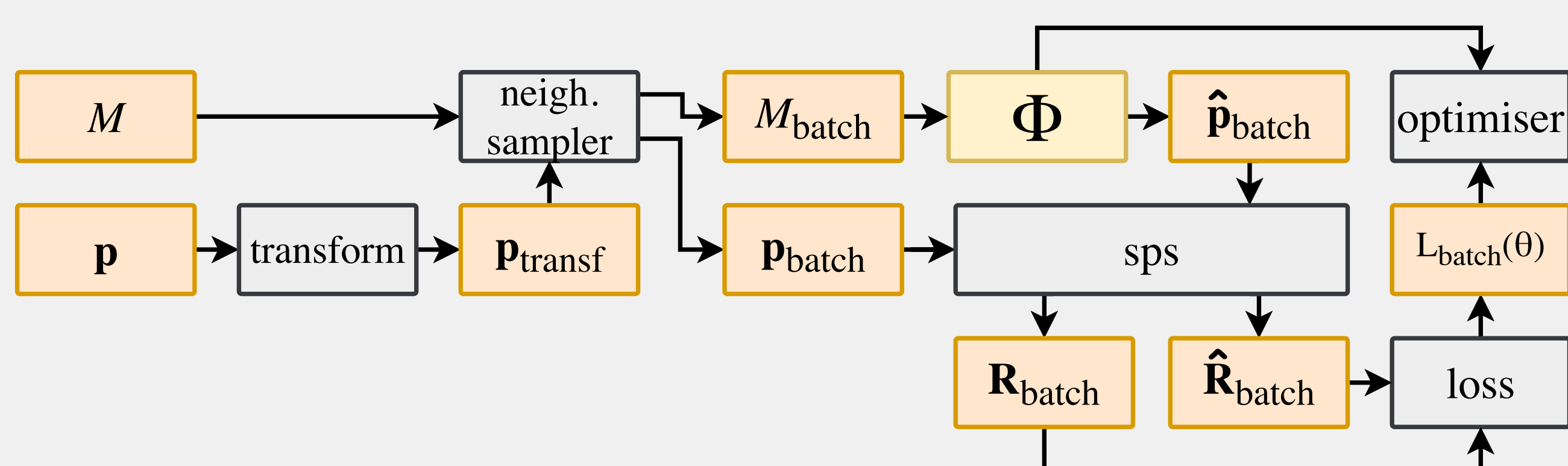


Figure 2. The pipeline used to train the *TopSpreadersNetwork*; orange blocks represent data entities, grey blocks indicate functional components, and the model is highlighted in yellow. Note that while the model is trained to optimise $\hat{\mathbf{R}}$, it is still capable of predicting the vector of spreading potential \mathbf{p} . This makes the results interpretable, in contrast to those of competitors.

Dataset

The *TopSpreadersDataset* was created by generating over two hundred artificial (Erdős-Rényi and Barabási-Albert) networks and supplementing them with selected real-world multilayer graphs. For each actor in each network, we determined their spreading potential vector \mathbf{p} through simulations of the Multilayer Independent Cascade Model (MICM) under varying spreading parameters.

Definition. A spreading potential of each actor can be expressed by a vector:

$$\mathbf{p} = [p_{ex}, p_{sl}, p_{pi}, p_{pl}]$$

where:

- p_{ex} - total number of activated actors across the network,
- p_{sl} - spreading duration, as a number of MICM iterations,
- p_{pi} - maximum number of actors activated simultaneously,
- p_{pl} - iteration at which the activation peak occurred.

Table 1. Networks comprising the *TopSpreadersDataset*, with their key parameters.

Network type	Layers	Actors	Nodes	Edges	Degree	Notes
artificial-er	3.52	558.19	1741.70	6684.00	24.13	100 graphs
artificial-pa	3.52	574.51	1976.07	42636.53	122.10	100 graphs
artificial-small	2.75	1000.00	2750.00	6609.12	13.22	8 graphs
arxiv	13	14065	26796	59026	8.39	Domenico2015
aucs	5	61	224	620	20.33	Rossi2015
ckmp	3	241	674	1370	11.37	Coleman1957
eu-trans	37	417	2034	3588	17.21	Cardillo2013
l2-course	2	41	82	297	14.49	Paradowski2024
lazega	3	71	212	1659	46.73	Snijders2006
timik	3	61702	102247	875191	28.37	Jankowski2017

Evaluation & results

The test split consists of all real-world networks and 14 artificial ones, meaning that ***TopSpreadersNetwork* is exposed only to artificial graphs during training**. This setup is designed to demonstrate the model's generalisation abilities. Transductive and non-trainable competitive methods are also evaluated on this split.

Since the number of super-spreaders in a network depends on its structural properties, we assess prediction quality by measuring y_{rel} in three k settings: T — the beginning of the ranking, i.e., the top-identified spreader; S — a saddle point separating super-spreaders from regular ones; F — the full ranking (to assess the method's stability).

By comparing raw values (*val*), we evaluate accuracy at the k -element level, whereas with the area under the curve (*auc*), we assess the stability of predictions.

Table 2. Average performance of the evaluated methods across two network types. Predictions with the highest quality (i.e. closest to 1) for each criterion are highlighted in color to indicate their rank — **first**, **second**, or **third**.

ϕ	Artificial networks				Real networks			
	T_{val}	S_{auc}	S_{val}	F_{auc}	T_{val}	S_{auc}	S_{val}	F_{auc}
random	0.538	0.672	0.759	0.875	0.520	0.629	0.749	0.863
deg-c	0.841	0.821	0.872	0.942	0.829	0.819	0.896	0.931
deg-cd	0.830	0.820	0.872	0.943	0.829	0.815	0.901	0.933
ghb-s	0.706	0.772	0.834	0.924	0.777	0.806	0.887	0.918
ghb-sd	0.662	0.771	0.830	0.920	0.777	0.801	0.886	0.919
deep-im	0.548	0.679	0.761	0.878	0.643	0.677	0.788	0.872
mn2v-km	0.611	0.711	0.790	0.903	0.586	0.658	0.786	0.881
ts-net	0.850	0.826	0.881	0.947	0.921	0.826	0.897	0.934

The results demonstrate the superiority of the *TopSpreadersNetwork* in most evaluation scenarios, outperforming both heuristic and ML-based competitors. Notably, the model remains robust when predicting on the out-of-distribution data.