Signed Graph Autoencoder for Explainable and Polarization-Aware Network Embeddings

Nikolaos Nakis[†] **Chrysoula Kosma**[‡] **Giannis Nikolentzos**[♠]
Michail Chatzianastasis[†] **Iakovos Evdaimon**[†] **Michalis Vazirgiannis**[†♣]
[†]LIX, École Polytechnique, Institute Polytechnique de Paris, France
[‡]Université Paris Saclay, Université Paris Cité, ENS Paris Saclay, CNRS, SSA, INSERM, Centre Borelli, France
[♠]Department of Informatics and Telecommunications, University of Peloponnese, Greece
[♣]Mohamed bin Zayed University of Artificial Intelligence, United Arab Emirates

Abstract

Autoencoders based on Graph Neural Networks (GNNs) have garnered significant attention in recent years for their ability to learn informative latent representations of complex topologies, such as graphs. Despite the prevalence of Graph Autoencoders, there has been limited focus on developing and evaluating explainable neural-based graph generative models specifically designed for signed networks. To address this gap, we propose the Signed Graph Archetypal Autoencoder (SGAAE) framework. SGAAE extracts node-level representations that express node memberships over distinct extreme profiles, referred to as archetypes, within the network. This is achieved by projecting the graph onto a learned polytope, which governs its polarization. The framework employs the Skellam distribution for analyzing signed networks combined with relational archetypal analysis and GNNs. Our experimental evaluation demonstrates the SGAAE's capability to successfully infer node memberships over underlying latent structures while extracting competing communities. Additionally, we introduce the 2-LEVEL network polarization problem and show how SGAAE is able to characterize such a setting. The proposed model achieves high performance in different tasks of signed link prediction across four real-world datasets, outperforming several baseline models. Finally, SGAAE allows for interpretable visualizations in the polytope space, revealing the distinct aspects of the network, as well as, how nodes are expressing them. (*Code available at*: `https://github.com/Nicknakis/SGAAE`).

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

1 Introduction

Graphs are commonly used to model complex relations and interactions between objects. Thus, different types of real-world data, such as molecules and social networks, can be naturally modeled as graphs. In many applications, we need to apply machine learning techniques on graphs. For instance, predicting various properties of molecules (e. g., quantum mechanical properties) (Gilmer et al., 2017) has become an emerging topic in chemoinformatics. This need for machine learning methods that operate on graphs led to the development of the field of graph representation learning (Hamilton, 2020). In the past few years, interest in this field has flourished. Graph representation learning is mainly comprised of Graph Neural Networks (GNNs) (Wu et al., 2020). These models have become the standard tool for performing machine learning tasks on graphs. Roughly speaking, GNNs learn vector representations of nodes (and potentially of graphs) in a supervised, end-to-end fashion. So far, GNNs have been mainly evaluated in supervised learning tasks while unsupervised learning of node representations, however, has not received the same amount of attention.

Existing methods for unsupervised learning of node representations typically employ an autoencoder (AE) framework. In AEs, the encoder corresponds to a GNN that aggregates the local information of nodes, while the decoder reconstructs the entire graph from the learned node representations. Most models are actually Variational Autoencoders (VAEs) which utilize latent variables. The latent variables are usually formulated as Gaussian variables, while some loss term

is employed to encourage them to be similar to some predefined prior distribution, typically an isotropic Gaussian distribution with diagonal covariance matrix. To make models more interpretable, some works have replaced the Gaussian distribution with others, e. g., Dirichlet distributions such that the latent variables describe graph cluster memberships (Li et al., 2020a). Due to their success for standard graphs, graph AEs and graph VAEs have been generalized to other types of graphs such as directed graphs (Salha et al., 2019; Kollias et al., 2022) and hypergraphs (Fan et al., 2021). However, prior work has mainly focused on unsigned graphs (where all edges are positive edges) even though signed graphs (i. e., graphs in which each edge has a positive or negative sign) are ubiquitous in the real world (Leskovec et al., 2010b).

In this paper, we propose a new model, the so-called Signed Graph Archetypal Autoencoder (SGAAE), which can embed nodes of signed graphs into vectors. Those representations capture the polarization that occurs within the graph. The different poles correspond to the corners of a polytope. To encode the polarization dynamics, the model employs a likelihood function for signed edges based on the Skellam distribution (Skellam, 1946), i. e., the discrete probability distribution of the difference between two independent Poisson random variables.

The paper also introduces the concept of 2-LEVEL polarization. Typically, polarization is understood as a simple "for or against" dynamic, which we characterize in terms of our work as a traditional 1-LEVEL view. In this perspective, people are grouped into two opposing sides (e.g., left vs. right in a political debate), and polarization is defined by strong agreement within each group. For instance, members of team A generally like and agree with each other while disagree with members of team B. However, reality is much more nuanced. Imagine characterizing dynamics in a political debate: while we can easily identify left and right clusters, focusing only on this first level of polarization misses an important layer of complexity. The second level of polarization highlights how negative connections or disagreements, and animosities form their own distinct structures. This is not just about team A disapproving team B. Within team A, for example, there could be subgroups that are characterized with mutual animosity over something else entirely, even if they broadly agree on the main issue. In other words, a hidden network within the network emerges, fueled by negativity. This second level reveals a deeper community structure, formed not by agreement, but by shared animosity. These hidden layers of negativity play a critical role in shaping the overall dynamics of polarization.

The contributions of this work are summarized as follows:

- *A novel AE for signed networks.* We introduce a carefully designed signed graph AE that exploits polarization to encode the underlying interactions between nodes in the input graphs. The choice of the latent space dynamics provides an identifiable, interpretable and natural representation of node memberships, which is suitable for social network settings, and especially for addressing multiple levels of polarization.
- *Superiority in real-world downstream tasks and visualization tasks.* We experimentally demonstrate that SGAAE significantly outperforms several baselines in the task of signed link prediction on real-world networks, while the learned polytope space allows for successful archetypal extraction and characterization.
- *Introduction of the* 2-LEVEL *network polarization.* Real signed networks contain polarized groups formed independently through positive or negative link structures, yielding a 2-LEVEL polarization problem. Existing models typically focus on polarization scenarios where groups are constrained to be formed uniquely based on very dense intra-group positive ties and at the same time dense negative inter-group ties (we refer to this as 1-LEVEL polarization). In the proposed 2-LEVEL polarization scenario, this constraint is lifted, allowing polarization to emerge separately from the structures defined by positive and negative links.
- *We show that the porposed AE can capture the* 2-LEVEL *network polarization.* We demonstrate that our proposed framework enables a node to possess two sets of embedding vectors describing positive and negative group memberships, effectively capturing both 1-LEVEL and 2-LEVEL polarization settings.

2 Related Work

Signed graph representation learning. Initial attempts for learning node representations in signed networks were inspired by advancements in the field of natural language processing (NLP), through embeddings on some vector space in an unsupervised manner. For example, SNE adapts the log-bilinear model from the field of NLP such that the learned node representations capture node's path and sign information (Yuan et al., 2017). Other embedding methods were designed to maintain structural balance, based on the assumption that triangles with an odd number of positive edges are more plausible than those with an even number. For instance, SIGNET builds upon the traditional

word2vec family of embedding approaches (Mikolov et al., 2013), but replaces the standard negative sampling approach with a new method which maintains structural balance in higher-order neighborhoods (Islam et al., 2018). Likewise, SiNE employs a multi-layer neural network to learn the node representations by optimizing an objective function satisfying structural balance theory (Wang et al., 2017). Based on ideas from balance theory, SGCN generalizes the prominent Graph Convolutional Network (Kipf and Welling, 2017) to signed graphs (Derr et al., 2018). SNEA replaces the mean-pooling strategy of SGCN with an attention mechanism which allows it to aggregate more important information from neighboring nodes based on balance theory (Li et al., 2020b). SIDE extends random walk-based embedding algorithms to signed graphs (Kim et al., 2018). Embeddings are learned by maximizing the likelihood over both direct and indirect signed links. POLE puts more focus on negative links than other works (Huang et al., 2022) and uses signed autocovariance to capture topological and signed similarities. To address some limitations of structural balance theory (e. g., it cannot handle directed graphs), SSNE leverages the status theory and uses different translation strategies in the embedding space for positive and negative links (Lu et al., 2019). SIGAT is a motif-based variant of the Graph Attention Network where the extracted motifs model the two aforementioned classic theories in sociology, i. e., balance theory and status theory (Huang et al., 2019). Finally, several recent studies in signed graph representation learning have integrated the signed graph Laplacian with GNNs, leading to the development of signed-spectral GNNs (Li et al., 2023; Singh and Chen, 2022; He et al., 2022; Fiorini et al., 2023).

Graph autoencoders. Owing to their simple structure, graph autoencoders (AEs) have been widely used in many domains to learn representations of nodes and/or graphs in an unsupervised manner. Here, we focus on node-level representations. We should mention though that there is a line of work that focuses on learning embeddings of graphs (Winter et al., 2021) or specific classes of graphs (Zhang et al., 2019), mostly applied to molecular generation (Simonovsky and Komodakis, 2018; Jin et al., 2018). Graph variational autoencoders (VAEs), instead of embedding each node into a vector as standard AEs, embed each node into a distribution. The prior distribution over the latent features of nodes needs to be chosen a priori. Most models employ the standard isotropic Gaussian distribution as their prior (Kipf and Welling, 2016b; Grover et al., 2019) or a Gaussian mixture distribution (Yang et al., 2019). Other models utilize more complex distributions such as Dirichlet distributions (Li et al., 2020a) or the Gamma distribution (Sarkar et al., 2020). The latent variables that emerge in those cases are more interpretable. In the case of the Dirichlet distributions, they correspond to graph cluster memberships, while Gamma-distributed latent variables result in non-negativity and sparsity of the learned embeddings and can also be considered as community memberships. Typically, graph AEs and VAEs employ some GNN as their encoder (Kipf and Welling, 2016b; Grover et al., 2019). These can be modified to better capture specific properties of graphs, such as the community structure (Salha-Galvan et al., 2022).

The closest work to the proposed SGAAE is the Signed relational Latent dIstance Model (SLIM) (Nakis et al., 2023) which is a latent distance-based model that extracts a unified embedding space based on the Skellam likelihood. Differently from our method, SLIM does not consider neural network-based representations, and it is not able to characterize 2-LEVEL polarization scenarios. Moreover, our work combines graph neural networks with classical statistical models. To the best of our knowledge our paper acts as the first signed graph autoencoder that defines a self-explainable latent space.

3 Proposed Method

Preliminaries. Let $\mathcal{G} = (\mathcal{V}, \mathcal{Y})$ be a *signed graph*, where $\mathcal{V} = \{1, \dots, N\}$ represents the set of nodes, and $\mathcal{Y} : \mathcal{V}^2 \to \mathsf{X} \subseteq \mathbb{R}$ is the map that assigns weights to pairs of nodes. An edge $(i, j) \in \mathcal{V}^2$ exists if the weight $\mathcal{Y}(i, j)$ is non-zero. Thus, the set of edges in the network is given by $\mathcal{E} := \{(i, j) \in \mathcal{V}^2 : \mathcal{Y}(i, j) \neq 0\}$. Considering the general case, we assume that a node pair relationship is characterized by integer values, i. e., $\mathsf{X} \subset \mathbb{Z}$. For simplicity, we focus on undirected graphs yielding $\mathcal{Y}(i, j)$=$\mathcal{Y}(j, i)$, but we note here that our method easily generalizes to directed networks as well. We denote by $\mathcal{E}^+$ the positive edge set $\{(i, j) \in \mathcal{V}^2 : \mathcal{Y}(i, j) > 0\}$ and by $\mathcal{E}^-$ the negative edge set $\{(i, j) \in \mathcal{V}^2 : \mathcal{Y}(i, j) < 0\}$. Finally, we introduce the adjacency matrices $\boldsymbol{Y}$, $\boldsymbol{Y}^+$, and $\boldsymbol{Y}^- \in \mathbb{Z}^{N \times N}$ with $\boldsymbol{Y}_{ij} = \mathcal{Y}(i, j)$ if $(i, j) \in \mathcal{E}$ and 0 otherwise, while $\boldsymbol{Y}^+_{ij} = \max(\boldsymbol{Y}_{ij}, 0)$ and $\boldsymbol{Y}^-_{ij} = \min(\boldsymbol{Y}_{ij}, 0)$.

We next present the Signed Graph Archetypal Autoencoder (SGAAE), which is depicted in Figure 1. Our primary aim is to design a graph AE capable of learning two sets of latent embeddings that are sufficient for explaining the relationships, as well as, for characterizing the structure defined by both the positive and negative graph interactions, in a disentangled manner. Specifically, for a given signed network $\mathcal{G} = (\mathcal{V}, \mathcal{Y})$, we aim to learn two sets of low-dimensional vectors $\{\tilde{\boldsymbol{z}}_i\}_{i \in \mathcal{V}} \in \mathbb{R}^K$, $\{\tilde{\boldsymbol{w}}_i\}_{i \in \mathcal{V}} \in \mathbb{R}^K$ ($K \ll |\mathcal{V}|$), where each

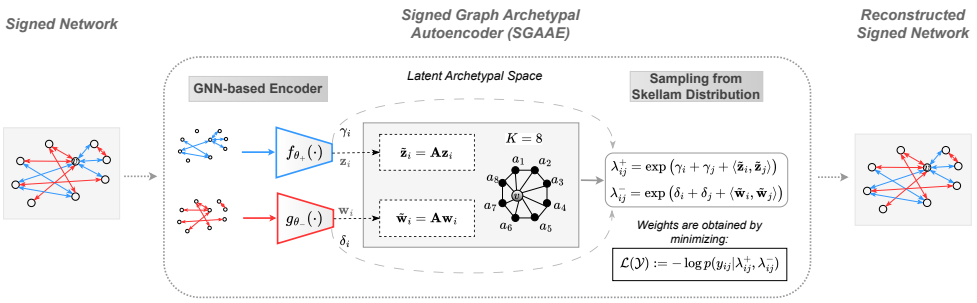


Figure 1: Framework of the proposed Signed Graph Archetypal Autoencoder (SGAAE). Given a signed network as input, the model utilizes two GNN-based encoding components working on the positive and negative interactions, respectively. The archetypal membership matrices $\boldsymbol{z}_i, \boldsymbol{w}_i$ are procured which are later multiplied with the archetypal matrix $\boldsymbol{A}$ to get the final node embeddings for the positive and negative spaces $\tilde{\boldsymbol{z}}_i, \tilde{\boldsymbol{w}}_i$. Then the final embeddings are used to calculate the Skellam rates optimizing for the Skellam log-likelihood for reconstructing the original signed graph.

is responsible for explaining the link structure as observed by $\mathcal{E}^+$ and $\mathcal{E}^-$, respectively. Importantly, the model should take into account both representations when predicting the existence of a link along its sign, or when characterizing the combined structure of the signed graph. Furthermore, such projections should express node memberships over distinct extreme profiles existing in the network, referred to as archetypes, facilitating the understanding of network polarization in a straightforward and interpretable way.

The 2-level Network Polarization. Quantifying network polarization has been long tied to antagonistic group mining in literature (Tzeng et al., 2020; Bonchi et al., 2019; Gao et al., 2015; Lo et al., 2011, 2013). Specifically, there has been great focus in community detection to discover groups defining **(i)** *strong intra-community ties*, i. e., nodes within the same community that have dense, positive connections with each other, reflecting high levels of agreement, similarity, or cooperation, and **(ii)** *weak or negative inter-community ties*, i. e., connections between nodes from different communities that are sparse, weak, or negative, indicating disagreement, conflict, or lack of interaction. We define this setting as 1-LEVEL polarization. This is illustrated in Figure 2a where the positive structure (blue points) is well separated from the negative structure (red points) that creates the 1-LEVEL polarized network. Such a case disregards potential structure in the negative ties, i. e., hidden disagreement/animosity structures, despite a uniform appearing inter-level negative tie pattern. An example of this is shown in Figure 2b where the same single network is re-ordered based on the community memberships that emerge from both the positive link structure (as in 1-LEVEL case), but also from the negative link structure. We argue that looking solely at the positive structure makes a model completely blind to the negative, highly polarized structure. We define this case as an example of 2-LEVEL polarization. A generative process of the multi-level polarized network is provided in Algorithm 1. Specifically, for a signed network with N number of nodes, we define a positive community assignment vector $\boldsymbol{\sigma}^{(+)} \in \{0, 1, 2, \ldots, K\}^N$, and a negative community assignment vector $\boldsymbol{\sigma}^{(-)} \in \{0, 1, 2, \ldots, K\}^N$. In addition, three probability matrices are given as input; **(i)** a positive community probability matrix $\boldsymbol{P}^{(+)} \in [0, 1]^{K \times K}$ that denotes the probability of a positive edge between two communities, **(ii)** the 1-LEVEL negative community probability matrix $\boldsymbol{P}^{(-)}_{1-level} \in [0, 1]^{K \times K}$ that denotes the probability of a negative edge between two communities based on the positive community assignments, and **(iii)** the 2-LEVEL negative community probability matrix $\boldsymbol{P}^{(-)}_{2-level} \in [0, 1]^{K \times K}$ that denotes the probability of a negative edge between two communities based on the negative community assignments. Finally, we introduce the polarization probability scalar $0 \leq \alpha \leq 1$ which controls the polarization level, i. e., how a negative edge is generated either based on the positive community vector $\boldsymbol{\sigma}^{(+)}$ (higher α values), or either via the independent negative community structure $\boldsymbol{\sigma}^{(-)}$ (lower α values). In Figure 2a we set $\alpha = 0.95$, and in Figure 2b we set $\alpha = 0.05$. (More details in Subsection 1.6 in the supplementary.)

Based on the above, we next combine and generalize recent advances in signed graph representation learning (Nakis et al., 2023) with GNNs, to design a graph AE based on an explainable latent space that confirms the effectiveness of network multi-level polarization targeting the analysis of social networks.

Archetypal Analysis. The extreme points of the convex hull enclosing the data can be referred to as archetypes. Archetypes, essentially represent the vertices of the convex hull defined by the data and

Algorithm 1 Signed Network Generation for 1-LEVEL and 2-LEVEL Polarized Networks

Input: N, $\boldsymbol{\sigma}^{(+)}$, $\boldsymbol{\sigma}^{(-)}$, $\mathbf{P}^{(+)}$,$\mathbf{P}^{(-)}_{1-level}$, $\mathbf{P}^{(-)}_{2-level}$, α.
Initialize two $N \times N$ zero matrices $\mathbf{A}^{(+)}$ and $\mathbf{A}^{(-)}$ that represent graphs $G^{(+)}$ and $G^{(-)}$.
for $i = 1$ to N **do**
  for $j = i + 1$ to N **do**
    if $rand() < \mathbf{P}^{(+)}[\sigma_i^{(+)}, \sigma_j^{(+)}]$ **then**
      $\mathbf{A}^{(+)}_{ij} \leftarrow 1$
      $\mathbf{A}^{(+)}_{ji} \leftarrow 1$
    else
      if $rand() < \alpha$ **then**
        if $rand() < \mathbf{P}^{(-)}_{1-level}[\sigma_i^{(+)}, \sigma_j^{(+)}]$ **then**
          $\mathbf{A}^{(-)}_{ij} \leftarrow 1$
          $\mathbf{A}^{(-)}_{ji} \leftarrow 1$
        end if
      else
        if $rand() < \mathbf{P}^{(-)}_{2-level}[\sigma_i^{(-)}, \sigma_j^{(-)}]$ **then**
          $\mathbf{A}^{(-)}_{ij} \leftarrow 1$
          $\mathbf{A}^{(-)}_{ji} \leftarrow 1$
        end if
      end if
    end if
  end for
end for
$\mathbf{A} \leftarrow \mathbf{A}^{(+)} - \mathbf{A}^{(-)}$
return $\mathbf{A}$ ▷ adjacency matrix of generated graph

can thus provide a detailed view of its inherent extreme structural traits. Therefore, archetypal analysis (AA) (Mørup and Kai Hansen, 2010; Cutler and Breiman, 1994) can allow us to identify and explain core data dependencies while extracting data representations as convex combinations of extremal points.

Let $\boldsymbol{X} \in \mathbb{R}^{P \times N}$ a data matrix, such that $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$. Then, the archetype matrix $\boldsymbol{A} \in \mathbb{R}^{P \times K}$, $\boldsymbol{A} = \{\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_K\}$, where $K \ll P$, can be extracted as follows:

$$\boldsymbol{\alpha}_j = \sum_{i=1}^{N} \boldsymbol{x}_i c_{ij}, \tag{1}$$

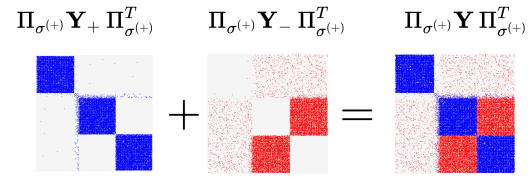
where $\boldsymbol{c}_j \in \Delta^{N-1}$ with Δ^{N-1} denoting the N-dimensional standard simplex, such as $c_{ij} \geq 0$ and $\sum_i c_{ij} = 1$. Let $\boldsymbol{A}$ be the convex hull of the data, then each point $\boldsymbol{x}_i$ is reconstructed as follows:

$$\boldsymbol{x}_i = \sum_{j=1}^{K} \boldsymbol{a}_j u_{ji} \tag{2}$$

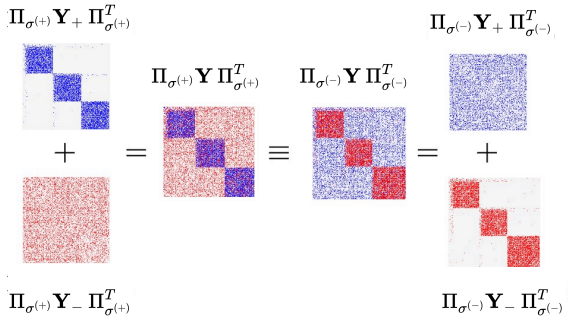
where $\boldsymbol{u}_i^T \in \Delta^{K-1}$ denoting the standard simplex in K-dimensions. Let $\boldsymbol{U} \in \mathbb{R}^{N \times K}$ the matrix formed by the K-dimensional embeddings for each data point. Then, $\boldsymbol{U}$ captures the representations as convex combinations of the archetypes in $\boldsymbol{A}$, as follows:

$$\boldsymbol{X} \approx \boldsymbol{XCU} \quad \text{s.t. } \boldsymbol{c}_j \in \Delta^{N-1} \text{ and } \boldsymbol{u}_i^T \in \Delta^{K-1}. \tag{3}$$

In the above, archetypes are represented by the corners of the convex hull, denoted as $\boldsymbol{A} = \boldsymbol{XC}$.



(a) 1-LEVEL POLARIZATION



(b) 2-LEVEL POLARIZATION

Figure 2: DIFFERENT LEVELS OF NETWORK POLARIZATION: Blue elements define positive ties whereas red define negative ties, displaying the total signed networks broken down into its two sign-specific components. **(a)** Showcases the traditional 1-LEVEL definition of network polarization focused on extracting structures with dense intra-community positive connections and negative inter-community ties. The adjacency matrices $\boldsymbol{Y}$, $\boldsymbol{Y}^+$, and $\boldsymbol{Y}^-$ and here are re-ordered based on the permutation matrix $\boldsymbol{\Pi}_{\boldsymbol{\sigma}^{(+)}}$ under the positive community memberships $\boldsymbol{\sigma}^{(+)}$. **(b)** Showcases two different permutations of the same signed network and its corresponding components that can be broken down under very dense positive communities and very dense negative communities. Importantly, the community memberships in the network for the negative and positive structures are not the same, yielding a 2-LEVEL polarization. The adjacency matrices in the left panel are re-ordered based on the permutation matrix $\boldsymbol{\Pi}_{\boldsymbol{\sigma}^{(+)}}$ under the positive community memberships $\boldsymbol{\sigma}^{(+)}$ while in the right panel we re-order $\boldsymbol{Y}$, $\boldsymbol{Y}^+$, and $\boldsymbol{Y}^-$ based on the permutation matrix $\boldsymbol{\Pi}_{\boldsymbol{\sigma}^{(-)}}$ under the negative community memberships $\boldsymbol{\sigma}^{(-)}$.

The Likelihood. Following Nakis et al. (2023), we use the Skellam distribution, which models the difference of two independent Poisson-distributed random variables ($y = N_1 - N_2 \in \mathbb{Z}$) with rates λ^+ and λ^- as:

$$P(y|\lambda^+, \lambda^-) = e^{-(\lambda^+ + \lambda^-)} \left(\frac{\lambda^+}{\lambda^-}\right)^{y/2} \mathcal{I}_{|y|}\left(2\sqrt{\lambda^+ \lambda^-}\right),$$

where $N_1 \sim Pois(\lambda^+)$ and $N_2 \sim Pois(\lambda^-)$, and $\mathcal{I}_{|y|}$ is the modified Bessel function of the first kind and order $|y|$. In particular, λ^+ generates the intensity of positive outcomes for y while λ^- generates the intensity of negative outcomes. Thus, the negative log-likelihood, used as our loss function, is computed as:

$$\mathcal{L}(\mathcal{Y}) := -\log p(y_{ij}|\lambda_{ij}^+, \lambda_{ij}^-) \tag{4}$$

$$= \sum_{i<j} (\lambda_{ij}^+ + \lambda_{ij}^-) - \frac{y_{ij}}{2} \log\left(\frac{\lambda_{ij}^+}{\lambda_{ij}^-}\right) - \log(I_{ij}^*), \tag{5}$$

where $I_{ij}^* := \mathcal{I}_{|y_{ij}|}\left(2\sqrt{\lambda_{ij}^+\lambda_{ij}^-}\right)$.

For relational data, the Skellam distribution rate parameter λ_{ij}^+ models the positive interaction intensity and λ_{ij}^- parameter models negative the interaction intensity between a node pair $\{i, j\}$. We constrain the latent space into a polytope to define a convex hull of the latent representations, achieving archetypal characterization. For relational data, we aim on expressing the embedding matrices—which encode each node's position in the network—as convex combinations of nodal archetypes. Consequently, we follow a similar methodology as the SLIM model, extending the relational archetypal analysis formulation to two membership vectors in the same latent spaces, independently reconstructing positive and negative representations from the extracted archetypes. This enables us to disentangle memberships in positive and negative communities, allowing for the characterization of 2-LEVEL polarization. Thus, the Skellam rates are calculated as:

$$\lambda_{ij}^+ = \exp\left(\gamma_i + \gamma_j + \langle \tilde{\boldsymbol{z}}_i, \tilde{\boldsymbol{z}}_j \rangle\right), \tag{6}$$

$$\lambda_{ij}^- = \exp\left(\delta_i + \delta_j + \langle \tilde{\boldsymbol{w}}_i, \tilde{\boldsymbol{w}}_j \rangle\right), \tag{7}$$

where $\tilde{\boldsymbol{z}}_i, \tilde{\boldsymbol{w}}_i \in \mathbb{R}^K$ the node embedding positions, $\langle \cdot, \cdot \rangle$ denotes the inner product while $\{\gamma_i, \delta_i\}_{i \in \mathcal{V}}$ denote the node-specific random effect terms. Essentially, γ_i, γ_j represent the tendency of a node to form positive connections while δ_i, δ_j the tendency to form negative connections. In other words, it accounts for degree heterogeneity in the positive and negative sub-networks, accordingly. To further construct explainable latent spaces, defined via memberships over archetypes, the final embedding positions $\tilde{\boldsymbol{z}}_i, \tilde{\boldsymbol{w}}_i$ are obtained as follows:

$$\tilde{\boldsymbol{z}}_i = \boldsymbol{A}\boldsymbol{z}_i, \tag{8}$$

$$\tilde{\boldsymbol{w}}_i = \boldsymbol{A}\boldsymbol{w}_i, \tag{9}$$

where $\boldsymbol{z}_i, \boldsymbol{w}_i \in \Delta^{K-1}$ and $\boldsymbol{A} \in \mathbb{R}^{K \times K}$ is the matrix with columns containing the archetypes for the unified space of positive and negative representations, that are reconstructed through the membership vectors. To define memberships that utilize archetypes to the maximum, i. e., archetypes belonging to the data, we introduce a temperature parameter in the softmax function used to define $\boldsymbol{z}_i, \boldsymbol{w}_i$. There exist additional methods that define the archetypal matrix through a gate function to achieve the same effect (Nakis et al., 2023). (A comprehensive list of all introduced notation is provided in Table 1 of the supplementary material.)

The latent space is self-explainable by construction, as each node's position in the latent space describes its membership to the K introduced archetypes. This approach ensures interpretability, allowing us to understand and explain the role and position of each node within the network. Our methods build on the principles of explainability demonstrated in prior works (Chen et al., 2019; Gautam et al., 2022; Kjærsgaard et al., 2024), where self-explainability is achieved by introducing prototypes in the latent space, a concept closely related to the relational archetypal analysis adopted in our model.

Characterizing 2-level Polarization. In real signed networks, we may have communities that are created based on positive links (*friendship*) and additional communities that are created due to negative links (*animosity*). Models that exploit 1-LEVEL polarization essentially introduce only one vector of community memberships, which constrains the node to belong to one community under both the positive and negative structures. In a 2-LEVEL polarization scenario, this constraint is removed, by enabling a node to have two sets of embedding vectors, each one describing the community structure based on either positive or negative link structures, separately. This is evident in our framework since we introduce two archetype membership matrices $\mathbf{Z}, \mathbf{W}$. In Figure 2b, the same unique signed network is defined and characterized by 2-LEVEL polarization and is re-ordered in two different ways. More specifically, the left part of Figure 2b focuses on discovering structure based on finding densely positively connected communities with densely negative connections leaving the (exact same) communities. This essentially describes the 1-LEVEL polarization scenario under a unique community membership vector $\mathbf{Z}$, and the ad-

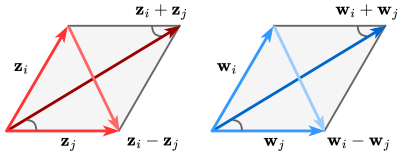


Figure 3: Polarization Identity—Inner product proximity in Skellam rates.

jacency matrices of the left panel are reordered based on exactly these community memberships (we show how both the positive and negative link structures are re-ordered as well as their combination on the same adjacency metric). As we witness, such a case finds very strong communities in the positive links but the negative structure seems spurious/random. The right panel of Figure 2b shows exactly the same network while seeking communities based on the negative link structure (by introducing a community membership vector $\mathbf{W}$), i. e., very negatively connected communities with positive links leaving the (exact same) communities. We observe that under this scenario, we successfully find very strong signals in the negative communities but the positive link structure seems random. Therefore, there is a question of choice between the panels to effectively describe this very differently structured network, based on the positive/negative structure. Motivated by this, our methodological design combines both possibilities by decoupling the community archetype memberships while accounting for this 2-LEVEL polarization problem, as we also show in section 4.

GNN-based Encoder. To handle the signed interactions in the network, we construct an encoder based on message-passing neural networks (MPNNs) for each edge type. Specifically, our approach separates the positive and negative interactions to learn distinct embeddings for each type. For the positive interactions, the encoder processes the graph containing only positive links by setting the weights of negative links to zero. For the negative interactions, we flip the sign of the negative links to treat them as positive during message passing. We use graph convolutional network (GCN) (Kipf and Welling, 2016a) with each layer in this network propagating information among connected nodes, followed by a Multilayer Perceptron (MLP) to produce the final embeddings and parameters. Thus, representations $\boldsymbol{Z}, \boldsymbol{\gamma}$ are produced for positive links, whereas $\boldsymbol{W}, \boldsymbol{\delta}$ are produced for the negative ones. (*More details are provided in the supplementary in Subsection 2.3.*)

Choosing the inner product. We here adopt the inner product as a proximity metric for the Skellam rates (see Figure 3) instead of the Euclidean distance (Nakis et al., 2023), as we argue that it can capture more complex relations since it weakly generalizes distance matrices (see supplementary). We can think of the archetype membership vectors $\boldsymbol{z}_i, \boldsymbol{w}_i$ as the participation to the extreme profiles/opinions present in the network of node i. For a real inner product space, by using the polarization identity we can express the inner product of Eq. (6) and (7) as:

$$\langle \tilde{\boldsymbol{z}}_i, \tilde{\boldsymbol{z}}_j \rangle = \frac{1}{4} \left(\|\boldsymbol{A}(\boldsymbol{z}_i + \boldsymbol{z}_j)\|^2 - \|\boldsymbol{A}(\boldsymbol{z}_i - \boldsymbol{z}_j)\|^2 \right), \quad (10)$$

$$\langle \tilde{\boldsymbol{w}}_i, \tilde{\boldsymbol{w}}_j \rangle = \frac{1}{4} \left(\|\boldsymbol{A}(\boldsymbol{w}_i + \boldsymbol{w}_j)\|^2 - \|\boldsymbol{A}(\boldsymbol{w}_i - \boldsymbol{w}_j)\|^2 \right). \quad (11)$$

The polarization identity can extend our understanding regarding the inner product as a similarity metric between two network nodes in terms of stochastic equivalence and homophily (Hoff, 2007), two fundamental properties of social network analysis. For either vector $\mathbf{z}$, and $\mathbf{w}$, the first term of the polarization identity essentially describes that the similarity of two nodes is high when $\|\boldsymbol{A}(\boldsymbol{z}_i + \boldsymbol{z}_j)\|^2$ is high, modeling stochastic equivalence. The second term denotes that a high rate can also be achieved when $\|\boldsymbol{A}(\boldsymbol{z}_i - \boldsymbol{z}_j)\|^2$ is low meaning that the two nodes are positioned close to each other and thus modeling homophily. Optimizing the model using the polarization identity expression, yields a more stable optimization due to the nice properties and convexity of the squared Euclidean norm.

All of the introduced notation can be found in Table 1 of the supplementary material. Finally, our model scales efficiently to large networks by employing a random sampling technique; for further details, see subsection 1.4 in the supplementary material.

4 Results and Discussion

We extensively evaluate the performance of our proposed method by comparing it to the prominent graph-based approaches designed for signed networks. All training details are provided in the supplementary.

Datasets and baselines. For the task of link prediction, the following *real-world* benchmark datasets, shown in Table 1, are considered: (**1**) *Reddit* (Kumar et al., 2018), (**2**) *Twitter* (Ordozgoiti et al., 2020), (**3**) *wikiRfA* (West et al., 2014), and (**4**) *wikiElec* (Leskovec et al., 2010a). Furhtermore, we evaluate the performance of the proposed graph autoencoder in several tasks with comparisons against benchmark signed graph representation methods, including: (**i**) POLE (Huang et al., 2022), (**ii**) SLF (Xu et al., 2019), (**iii**) SIGAT (Huang et al., 2019), (**iv**) SIDE (Kim et al., 2018), (**v**) SIGNET (Islam et al., 2018), (**v**) the Spectral-SGCN (S-GCN) Singh and Chen (2022), and finally, (**vi**) SLIM (Nakis et al., 2023). (For Additional details see subsection 2.2 in the supplementary).

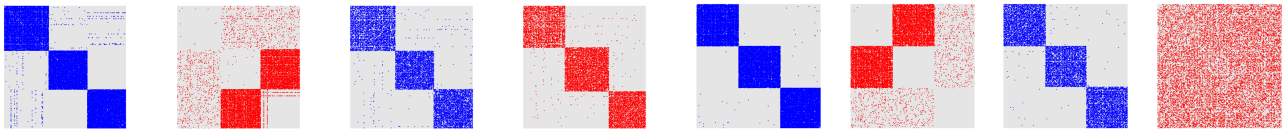
Characterizing both polarization levels. We consider the two polarized networks of Figure 2 and compare our SGAAE with SLIM in terms of expressing the two levels of network polarization. We use this baseline as it is the only direct competitor defining polarization memberships. Results are provided in Figure 4, where we present the re-ordered adjacency matrices based on the structures uncovered by the two models. We witness how SLIM successfully characterizes the

Table 1: Network statistics; #Nodes, #Positive, #Negative links.

	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{Y}^+\rvert$	$\lvert\mathcal{Y}^-\rvert$	Density
Reddit	35,776	128,182	9,639	0.0001
Twitter	10,885	238,612	12,794	0.0021
wiki-Elec	7,117	81,277	21,909	0.0020
wiki-RfA	11,332	117,982	66,839	0.0014

Table 2: Area Under Curve (AUC-PR) scores for representation size of $K = 8$. (OOM: memory or high runtime error.) The standard error of the mean for all the cases is approximately 0.005.

	WikiElec			*WikiRfa*			*Twitter*			*Reddit*		
Task	$p@n$	$p@z$	$n@z$	$p@n$	$p@z$	$n@z$	$p@n$	$p@z$	$n@z$	$p@n$	$p@z$	$n@z$
POLE	.929	.922	.544	.927	.937	.779	.998	.932	.668	OOM	OOM	OOM
SLF	**.964**	.926	.787	**.983**	.922	.881	.994	.870	.740	**.966**	.956	.850
SiGAT	.960	.724	.439	.969	.646	.497	**.999**	.861	.582	.965	.692	.232
SIDE	.907	.779	.608	.920	.806	.739	.974	.831	.469	.957	.820	.614
SiGNet	.944	.670	.298	.950	.572	.417	.998	.647	.248	.956	.510	.083
S-SGCN	.952	.920	.759	.9630	.911	.853	.989	.917	.601	.960	.935	.806
SLIM	.953	.956	.785	.973	.969	.907	**.999**	**.962**	.813	.958	**.960**	.850
SGAAE	.960	**.972**	**.895**	.980	**.976**	**.951**	**.999**	.960	**.816**	.964	.961	**.866**



(a) SGAAE:1-LEVEL (b) SGAAE:1-LEVEL (c) SGAAE:2-LEVEL (d) SGAAE:2-LEVEL (e) SLIM:1-LEVEL (f) SLIM:1-LEVEL (g) SLIM:2-LEVEL (h) SLIM:2-LEVEL

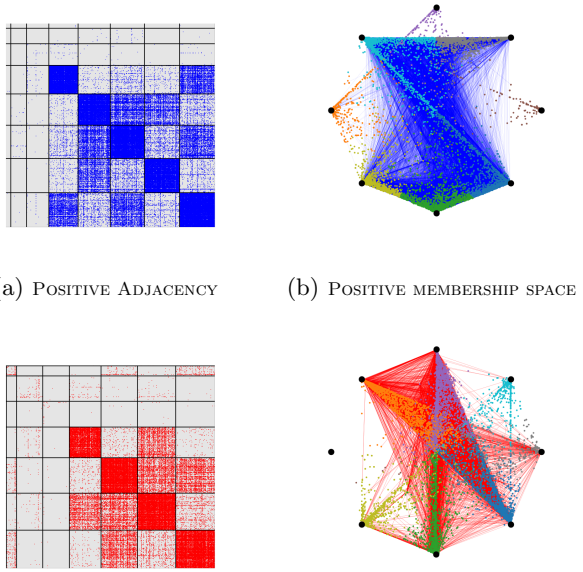
Figure 4: INFERRED COMMUNITY MEMBERSHIPS: (a)-(d) visualization of the re-ordered adjacency matrices based on the inferred community memberships of SGAAE, proving the expressive capabilities to provide characterization for both levels of polarization. (e)-(h) the same visualization for SLIM failing to explain or detect the structure over the negative ties and thus unable to account for the different polarization levels.

1-LEVEL polarization but fails in the more advanced task of the 2-LEVEL polarization. On the other hand, SGAAE successfully infers all latent structures, as powered by disentangling community memberships in terms of both positive and negative structures.

Link prediction. We assess the performance of the proposed method in the task of sign prediction and signed link prediction, to quantify its capacity to infer meaningful links between nodes along with the sign, i. e., positive or negative, of these links. In this experiment, we eliminate 20% of network links and learn node embeddings based on the residual network. To create zero instances in the test set, the eliminated edges are coupled with a sample of the same number of node pairs, not showing as edges in the original network. Link signs in signed graphs introduce additional complexity to various tasks, leading to different types of prediction challenges. We address the tasks of (1) *Link Sign Prediction* (p@n), and (2) *Signed Link Prediction* (p@z, n@z), following the experimental setup as in Xu et al. (2019); Huang et al. (2022); Nakis et al. (2023). Specifically, the p@n task involves predicting the sign of a removed link, assessing the model's ability to differentiate between positive and negative links. The p@z task involves positive versus zero link prediction, assessing the model's ability to predict the presence of positive links. The n@z task involves negative versus zero link prediction, focusing on predicting the presence of negative links. We use robust assessment measures such as the precision-recall (AUC-PR) curve (AUC-ROC scores are provided in Table 2 in the supplementary), as a result of the sparsity and class imbalances found in signed networks.

For the *link sign prediction* task, denoted as $p@n$, the AUC-PR values for the undirected case are shown in Table 2. We observe that the proposed SGAAE model lies among the best-performing methods, while in a few cases is slightly outperformed by SLF. In the more challenging *signed link prediction* task, denoted as $p@z$ for positive and as $n@z$ for negative samples, we predict deleted links against disconnected network pairs and accurately determine each link's sign. The test set is divided into the positive and negative disconnected subsets, and models performance is assessed on those subgroups. AUC-PR scores are shown in Table 2. The proposed SGAAE method outcompetes all baselines for almost all datasets. We can observe that the performance improvements are, in several cases, significant, e. g., scores on the *WikiElec* dataset. Here, we highlight the superiority in the performance, especially for the arduous task $n@z$, where we observe the benefits of decoupling the archetypal positive and negative memberships. (For the effect of dimensionality K on model performance see Figure 4 in the supplementary.)

Network visualization. We show how our model can successfully infer archetypal structures that can characterize the negative and positive link patterns in the network. In Figure 5, we provide the re-ordered adjacency matrices focused on the positive and negative



(a) Positive Adjacency (b) Positive membership space

(c) Negative Adjacency (d) Negative membership space

Figure 5: Network Visualizations: for the *WikiRfa* where we show the re-ordered adjacency matrix based on the maximum positive/negative memberships to the archetypes—the positive/negative membership space where essentially we visualize the soft-memberships over the archetypes in a circular plot where each archetype is positioned every $\frac{2\pi}{K}$ rads.

ties that validate the successful characterization over latent communities in the WikiRfa network, showcasing 2-level polarization. Furthermore, we provide the positive/negative membership spaces and visualize the representation of each node in terms of the network archetypes. We observe that some archetypes are solely populated under one of the membership matrices $\mathbf{Z}$ or $\mathbf{W}$, yielding archetypes that describe groups formed uniquely under the positive or negative link structure. Consequently, this verifies the importance of accounting for different levels of polarization.

The novelty and performance gain of our model lie in its 2-level polarization characterization, which disentangles the latent spaces into positive and negative components. Our results confirm that the most challenging task in signed link prediction is distinguishing negative links from non-links, and our model significantly outperforms baselines on the *WikiElec* and *WikiRfa* datasets – achieving an average improvement of 30% (11% over the second-best) on *WikiElec* and 24% (5% over the most competitive) on *WikiRfa* in the n@z task. Figure 5 illustrates how these networks exhibit 2-level polarization by uncovering distinct communities based on positive and negative structures, leading to a more precise negative link characterization. In contrast, the Twitter network (Figure 6 in the supplementary) reflects a one-level polarization scenario, where our model matches the performance of the one-level polarization model SLIM – a finding that corroborates our artificial network experiments in Figure 4. For the *Reddit* dataset, characterized by a strong "us-versus-them" dynamic, our model outperforms SLIM by 1.6% in the n@z task. Overall, our model effectively captures two-level polarization where it exists and performs comparably to one-level models when it does not, highlighting its robust performance.

5 Conclusion, Limitations & Impact

In this paper, we presented SGAAE, a signed graph AE that extracts node-level representations that express node memberships over distinct extreme profiles. This is achieved by projecting the graph onto a learned polytope, which allows for polarization characterization. We showcased how our model can account for different levels of polarization, coupled with state-of-the-art performance in link prediction for signed networks. Our concept of 2-level polarization is in agreement with recent works providing real-world evidence that 1-level polarization fails to capture the complexity of actual networks (Ghasemian and Christakis, 2024). Specifically, the authors. observed that in multiple village social networks, the majority of negative ties occur within communities rather than between them. This finding suggests that the structure of negative links can develop independently from the positive network structure – a departure from traditional approaches in signed network analysis that have predominantly focused on inter-community dynamics. In terms of limitations, SGAAE defines a high number of model parameters making the optimization highly non-convex and prone to local minima. Understanding polarization in social networks holds significant broader and societal impacts. Our proposed SGAAE framework can offer insights into the mechanisms driving division and echo chambers within online communities, as enabled by the archetypal social space.

Acknowledgements

We gratefully acknowledge the reviewers for their constructive feedback and insightful comments. This work was supported by the French National research agency via the AML-HELAS (ANR-19-CHIA-0020) project. M. C. is also supported by the EUR BERTIP (ANR-18-EURE-0002), Plan France 2030. C. K. is supported by the IdAML Chair hosted at ENS Paris-Saclay, Université Paris-Saclay.

References

F. Bonchi, E. Galimberti, A. Gionis, B. Ordozgoiti, and G. Ruffo. Discovering Polarized Communities in Signed Networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, page 961–970, 2019.

C. Chen, O. Li, D. Tao, A. Barnett, C. Rudin, and J. K. Su. This looks like that: Deep learning for interpretable image recognition. In *Advances in Neural Information Processing Systems*, 2019.

A. Cutler and L. Breiman. Archetypal analysis. *Technometrics*, 36(4):338–347, 1994.

T. Derr, Y. Ma, and J. Tang. Signed Graph Convolutional Network. In *Proceedings of the 2018 IEEE International Conference on Data Mining*, pages 929–934, 2018.

H. Fan, F. Zhang, Y. Wei, Z. Li, C. Zou, Y. Gao, and Q. Dai. Heterogeneous Hypergraph Variational Autoencoder for Link Prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8): 4125–4138, 2021.

S. Fiorini, S. Coniglio, M. Ciavotta, and E. Messina. SigMaNet: One Laplacian to Rule Them All. In *Proceedings of the 37th AAAI Conference on Artificial Intelligence*, pages 7568–7576, 2023.

M. Gao, E.-P. Lim, D. Lo, and P. K. Prasetyo. On detecting maximal quasi antagonistic communities in signed graphs. *Data Mining and Knowledge Discovery*, 30, 04 2015. doi: 10.1007/s10618-015-0405-2.

S. Gautam, A. Boubekki, S. Hansen, S. Salahuddin, R. Jenssen, M. Höhne, and M. Kampffmeyer. ProtoVAE: A Trustworthy Self-Explainable Prototypical Variational Model. In *Advances in Neural Information Processing Systems*, pages 17940–17952, 2022.

A. Ghasemian and N. A. Christakis. The structure and function of antagonistic ties in village social networks. *Proceedings of the National Academy of Sciences*, 121 (26):e2401257121, 2024.

J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl. Neural Message Passing for Quantum Chemistry. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1263–1272, 2017.

A. Grover, A. Zweig, and S. Ermon. Graphite: Iterative Generative Modeling of Graphs. In *Proceedings of the 36th International Conference on Machine Learning*, pages 2434–2444, 2019.

W. L. Hamilton. *Graph representation learning*. Morgan & Claypool Publishers, 2020.

Y. He, M. Perlmutter, G. Reinert, and M. Cucuringu. MSGNN: A Spectral Graph Neural Network Based on a Novel Magnetic Signed Laplacian. In *Proceedings of the 1st Learning on Graphs Conference*, pages 40–1, 2022.

P. Hoff. Modeling homophily and stochastic equivalence in symmetric relational data. *Advances in Neural Information Processing Systems 20 - Proceedings of the 2007 Conference*, 20, 12 2007.

J. Huang, H. Shen, L. Hou, and X. Cheng. Signed Graph Attention Networks. In *Proceedings of the 28th International Conference on Artificial Neural Networks: Workshop and Special Sessions*, pages 566–577, 2019.

Z. Huang, A. Silva, and A. Singh. POLE: Polarized Embedding for Signed Networks. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining*, pages 390–400, 2022.

M. R. Islam, B. Aditya Prakash, and N. Ramakrishnan. SIGNet: Scalable Embeddings for Signed Networks. In *Proceedings of the 22nd Pacific-Asia Conference in Knowledge Discovery and Data Mining*, pages 157–169, 2018.

W. Jin, R. Barzilay, and T. Jaakkola. Junction Tree Variational Autoencoder for Molecular Graph Generation. In *Proceedings of the 35th International Conference on Machine Learning*, pages 2323–2332, 2018.

J. Kim, H. Park, J.-E. Lee, and U. Kang. SIDE: Representation Learning in Signed Directed Networks. In *Proceedings of the 2018 World Wide Web Conference*, pages 509–518, 2018.

T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016a.

T. N. Kipf and M. Welling. Variational Graph Auto-Encoders. *arXiv preprint arXiv:1611.07308*, 2016b.

T. N. Kipf and M. Welling. Semi-Supervised Classification with Graph Convolutional Networks. In *The 5th International Conference on Learning Representations*, 2017.

R. Kjærsgaard, A. Boubekki, and L. Clemmensen. Pantypes: diverse representatives for self-explainable models. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, pages 13230–13237, 2024.

G. Kollias, V. Kalantzis, T. Idé, A. Lozano, and N. Abe. Directed Graph Auto-Encoder. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, pages 7211–7219, 2022.

S. Kumar, W. L. Hamilton, J. Leskovec, and D. Jurafsky. Community interaction and conflict on the web. In *Proceedings of the 2018 World Wide Web*

Conference on World Wide Web, pages 933–943. International World Wide Web Conferences Steering Committee, 2018.

J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. In *WWW*, page 641–650, 2010a.

J. Leskovec, D. Huttenlocher, and J. Kleinberg. Signed Networks in Social Media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1361–1370, 2010b.

J. Li, J. Yu, J. Li, H. Zhang, K. Zhao, Y. Rong, H. Cheng, and J. Huang. Dirichlet Graph Variational Autoencoder. In *Advances in Neural Information Processing Systems*, pages 5274–5283, 2020a.

Y. Li, Y. Tian, J. Zhang, and Y. Chang. Learning Signed Network Embedding via Graph Attention. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pages 4772–4779, 2020b.

Y. Li, M. Qu, J. Tang, and Y. Chang. Signed Laplacian Graph Neural Networks. In *Proceedings of the 37th AAAI Conference on Artificial Intelligence*, pages 4444–4452, 2023.

D. Lo, D. Surian, K. Zhang, and E.-P. Lim. Mining direct antagonistic communities in explicit trust networks. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, page 1013–1018, 2011.

D. Lo, D. Surian, P. K. Prasetyo, K. Zhang, and E.-P. Lim. Mining direct antagonistic communities in signed social networks. *Information Processing & Management*, 49(4):773–791, 2013.

C. Lu, P. Jiao, H. Liu, Y. Wang, H. Xu, and W. Wang. SSNE: Status Signed Network Embedding. In *Proceedings of the 23rd Pacific-Asia Conference in Knowledge Discovery and Data Mining*, pages 81–93, 2019.

T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013.

M. Mørup and L. Kai Hansen. Archetypal analysis for machine learning. In *2010 IEEE International Workshop on Machine Learning for Signal Processing*, pages 172–177, 2010.

N. Nakis, A. Celikkanat, L. Boucherie, C. Djurhuus, F. Burmester, D. M. Holmelund, M. Frolcová, and M. Mørup. Characterizing Polarization in Social Networks using the Signed Relational Latent Distance Model. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, pages 11489–11505, 2023.

B. Ordozgoiti, A. Matakos, and A. Gionis. Finding large balanced subgraphs in signed networks. In *Proceedings of The Web Conference 2020*, page 1378–1388, 2020.

G. Salha, S. Limnios, R. Hennequin, V.-A. Tran, and M. Vazirgiannis. Gravity-Inspired Graph Autoencoders for Directed Link Prediction. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 589–598, 2019.

G. Salha-Galvan, J. F. Lutzeyer, G. Dasoulas, R. Hennequin, and M. Vazirgiannis. Modularity-aware graph autoencoders for joint community detection and link prediction. *Neural Networks*, 153:474–495, 2022.

A. Sarkar, N. Mehta, and P. Rai. Graph Representation Learning via Ladder Gamma Variational Autoencoders. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pages 5604–5611, 2020.

M. Simonovsky and N. Komodakis. GraphVAE: Towards Generation of Small Graphs Using Variational Autoencoders. In *Proceedings of the 27th International Conference on Artificial Neural Networks*, pages 412–422, 2018.

R. Singh and Y. Chen. Signed Graph Neural Networks: A Frequency Perspective. *arXiv preprint arXiv:2208.07323*, 2022.

J. G. Skellam. The frequency distribution of the difference between two poisson variates belonging to different populations. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 109(3): 296–296, 1946.

R.-C. Tzeng, B. Ordozgoiti, and A. Gionis. Discovering conflicting groups in signed networks. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 10974–10985. Curran Associates, Inc., 2020.

S. Wang, J. Tang, C. Aggarwal, Y. Chang, and H. Liu. Signed Network Embedding in Social Media. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pages 327–335, 2017.

R. West, H. S. Paskov, J. Leskovec, and C. Potts. Exploiting social network structure for person-to-person sentiment analysis. *TACL*, 2:297–310, 2014.

R. Winter, F. Noé, and D.-A. Clevert. Permutation-Invariant Variational Autoencoder for Graph-Level Representation Learning. In *Advances in Neural Information Processing Systems*, pages 9559–9573, 2021.

Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip. A comprehensive survey on graph neural

networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24, 2020.

P. Xu, J. Wu, W. Hu, and B. Du. Link prediction with signed latent factors in signed social networks. *Proceedings of the Acm Sigkdd International Conference on Knowledge Discovery and Data Mining*, pages 1046–1054, 2019.

L. Yang, N.-M. Cheung, J. Li, and J. Fang. Deep Clustering by Gaussian Mixture Variational Autoencoders with Graph Embedding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6440–6449, 2019.

S. Yuan, X. Wu, and Y. Xiang. SNE: Signed Network Embedding. In *Proceedings of the 21st Pacific-Asia Conference in Knowledge Discovery and Data Mining*, pages 183–195, 2017.

M. Zhang, S. Jiang, Z. Cui, R. Garnett, and Y. Chen. D-VAE: A Variational Autoencoder for Directed Acyclic Graphs. In *Advances in Neural Information Processing Systems*, pages 1588–1600, 2019.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Supplementary Material

1 THE PROPOSED SGAAE MODEL: ADITIONAL DETAILS

We next provide additional important information and experimental results that further elucidate the key concepts presented in the main paper. In this supplementary material, we offer detailed explanations of our methodology, including derivations, proofs, and additional experimental settings. These extended details are designed to reinforce the validity of our approach, ensure transparency in our methods, and guarantee the reproducibility of our results across various datasets and scenarios.

1.1 Notation

Table 1: Table of symbols

Symbol	Description
$\mathcal{G}$	A signed graph where edges can have both positive and negative signs
$\mathcal{V}$	Vertex set
$\mathcal{E}$	Edge set
$\mathcal{E}^+$	Positive edge set
$\mathcal{E}^-$	Negative edge set
$\mathcal{E}^-$	Negative edge set
$\mathbf{Y}$	Total adjacency matrix $\in \mathbb{Z}^{N\times N}$ where $\mathbf{Y}_{ij} = \mathcal{Y}(i,j)$ if $(i,j) \in \mathcal{E}$ and 0 otherwise
$\mathbf{Y}^+$	Positive links adjacency matrix $\in \mathbb{Z}^{N\times N}$ where $\mathbf{Y}^+_{ij} = \max(\mathbf{Y}_{ij}, 0)$
$\mathbf{Y}^-$	Negative links adjacency matrix $\in \mathbb{Z}^{N\times N}$ where $\mathbf{Y}^-_{ij} = \min(\mathbf{Y}_{ij}, 0)$
N	Number of nodes
K	Dimension size/Number of Archetypes
$\mathbf{P}^{(+)}$	Alg. 1: Probability matrix of a positive edge between two communities $\in [0,1]^{K\times K}$
$\mathbf{P}^{(-)}_{1-level}$	Alg. 1: Probability matrix of a negative edge between two communities based on the positive community assignments $\in [0,1]^{K\times K}$
$\mathbf{P}^{(-)}_{2-level}$	Alg. 1: Probability matrix of a negative edge between two communities based on the negative community assignments $\in [0,1]^{K\times K}$
$\boldsymbol{\sigma}^{(-)}$	Negative community assignment vector $\in \{0,1,2,\ldots,K\}^N$
$\boldsymbol{\sigma}^{(+)}$	Positive community assignment vector $\in \{0,1,2,\ldots,K\}^N$
$\mathbf{C}$	Extreme points construction matrix for the convex hull enclosing the data matrix $\mathbf{X}$ $\mathbf{C} \in \Delta^{N-1}$ such that $\mathbf{C} = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_N\}$
$\mathbf{X}$	Example data matrix, such that $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ with $\mathbf{X} \in \mathbb{R}^{P\times N}$
$\mathbf{u}_i$	the vector representing the convex combinations over the K archetypes for datapoint $\mathbf{x}_i \in \Delta^{K-1}$
γ_i, δ_i	Positive and negative space bias terms of node i, respectively
$\tilde{\mathbf{z}}_i$	Positive space latent embedding for node $i \in \mathbb{R}^K$
$\tilde{\mathbf{w}}_i$	Negative space latent embedding for node $i \in \mathbb{R}^K$
$\mathbf{z}_i$	Positive space latent membership vector over the K archetypes for node $i \in \Delta^{K-1}$
$\mathbf{w}_i$	Negative space latent membership vector over the K archetypes for node $i \in \Delta^{K-1}$
λ_{ij}	Poisson rate (intensity) of node pair (i,j)
λ^+_{ij}	Positive interaction Skellam rate (intensity) of node pair (i,j) of the Skellam distribution
λ^-_{ij}	Negative interaction Skellam rate (intensity) of node pair (i,j) of the Skellam distribution
$\mathcal{I}_{\lvert y\rvert}$	Modified Bessel function of the first kind and order $\lvert y\rvert$
Δ^{K-1}	The standard $K-$simplex
$\mathbf{A}$	The matrix containing the archetypes (extreme points of the convex hull) with $\mathbf{A} \in \mathbb{R}^{(K)\times(K)}$

1.2 The concept of 2-LEVEL polarization

Typically, polarization is understood as a simple "for or against" dynamic, which we characterize in terms of our work as a traditional 1-LEVEL view.

In this perspective, people are grouped into two opposing sides (e.g., left vs. right in a political debate), and polarization is defined by strong agreement within each group. For instance, members of team A generally like and agree with each other while dislike and disagree with members of team B. However, reality is much more nuanced. Imagine characterizing dynamics in a political debate: while we can easily identify left and right clusters, focusing only on this first level of polarization misses an important layer of complexity.

The second level of polarization highlights how negative connections or disagreements, and animosities form their own distinct structures. This is not just about team A disliking team B. Within team A, for example, there could be subgroups that bond over a mutual dislike of something else entirely, even if they broadly agree on the main issue. In other words, a hidden network within the network emerges, fueled by negativity. This second level reveals a deeper community structure, formed not by agreement, but by shared animosity. These hidden layers of negativity play a critical role in shaping the overall dynamics of polarization.

1.3 Link prediction details

For the proposed method, on the concatenated respective Skellam rates and log-rates, we fit a logistic regression classifier, as $\chi_{ij} = \left[\lambda_{ij}^{+}, \lambda_{ij}^{-}, \log \lambda_{ij}^{+}, \log \lambda_{ij}^{-}\right]$. A concatenation can benefit from a linear function of the rates as well as their ratio or product as permitted by the log transformation, as our Skellam probability formulation depends on both the rates ratio and products. For the baselines, we create feature vectors using the average, weighted L_1, weighted L_2, concatenation and the Hadamard product as operators. With the exception of the Hadamard product, which is utilized directly for predictions, we fit a logistic regression classifier for each of these feature vectors and select the operator that yields the highest performance for each baseline.

1.4 Complexity analysis

The time and space complexity of SGAAE can be broken down into two parts:

Skellam log-likelihood calculation. The model scales prohibitively as $\mathcal{O}(N^2)$ due to the need to compute the node pairwise Gram matrix, which restricts the analysis of large-scale networks. To address this limitation, we adopt an unbiased estimation of the log-likelihood through random sampling. Specifically, gradient updates are performed based on the log-likelihood of a block formed by a randomly sampled set S of network nodes (sampled with replacement per iteration). This approach improves scalability by reducing the space and time complexity to $\mathcal{O}(S^2)$.

GCN layers message passing. The time complexity of each GCN layer is: $\mathcal{O}(E \times H_{input} + N \times H_{input} \times H_{output})$ while the space complexity can be approximated as: $\mathcal{O}(E + N \times H_{input} + N + H_{input} \times H_{output})$, where E is the number of network edges, H_{input} the number of features per node, H_{output} the number of hidden features, and N the number of network nodes.

1.5 Gram Matrices Weakly Generalize Distance Matrices

We next provide an analysis to justify the choice of the inner product over the Euclidean distance proposed in Nakis et al. (2023), as a proximity metric for the Skellam rates. We achieve this by showing that the inner product can provide a more expressive metric.

Consider a set of vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ in $\mathbb{R}^d$. Let X be the $d \times n$ matrix whose columns are the vectors $\mathbf{x}_i$.

The Gram matrix G is defined as:

$$G = X^T X \tag{1}$$

Then, the inner product between two vectors is calculated as:

$$G_{ij} = \mathbf{x}_i^T \mathbf{x}_j \tag{2}$$

The Euclidean distance matrix D is also defined as:

$$D_{ij} = ||\mathbf{x}_i - \mathbf{x}_j|| \tag{3}$$

The squared Euclidean distance formula can be expressed in terms of inner products, as:

$$D_{ij}^2 = (\mathbf{x}_i - \mathbf{x}_j)^T(\mathbf{x}_i - \mathbf{x}_j) \tag{4}$$

By extending the product, the squared distance between any two vectors can be computed directly from the Gram matrix as:

$$D_{ij}^2 = G_{ii} + G_{jj} - 2G_{ij} \tag{5}$$

which justifies that the Gram matrix contains all the necessary information to compute the distance matrix.

We also aim to answer whether we can uniquely reconstruct the Gram matrix (or the original vectors) from the distance matrix alone. Given the distance matrix D, we have $G_{ij} = \frac{1}{2}(G_{ii} + G_{jj} - D_{ij}^2)$. However, without additional information, the Gram matrix cannot be uniquely determined from the distance matrix alone. For instance, different sets of vectors that are related by an isometric transformation (rotation, reflection, or translation) will have the same distance matrix but different Gram matrices.

Conclusion on the choice of the inner product. The Gram matrix weakly generalizes the distance matrix because the distance matrix can be uniquely derived from the Gram matrix using $D_{ij}^2 = G_{ii} + G_{jj} - 2G_{ij}$. The distance matrix alone is insufficient to uniquely determine the Gram matrix, as it does not contain information about the absolute positioning and orientation of the vectors, but rather their relative distances.

1.6 Generating networks with multi-level polarization

We here continue by generating networks via Algorithm 1 of the main paper, for different levels of polarization by controlling the scalar value α. For all networks we use $N = 100$ number of nodes while we set the group probability matrices as:

The positive edges probability matrix $P^{(+)}$ is:

$$P^{(+)} = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.1 & 0.8 & 0.1 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$

The negative edges probability matrix for 1-level polarization $P^{(-)}_{1-level}$ is:

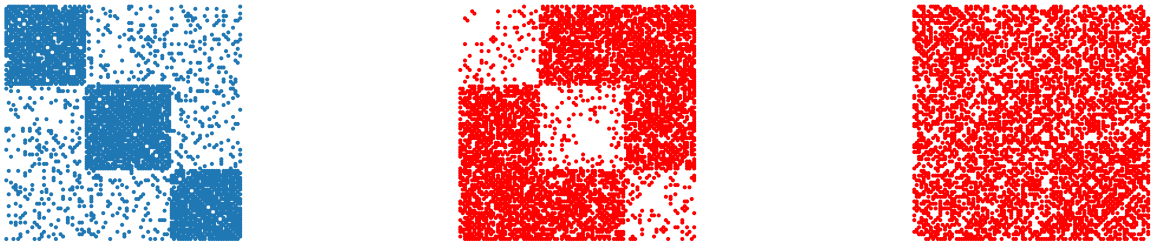
$$P^{(-)}_{1-level} = \begin{bmatrix} 0.05 & 0.6 & 0.6 \\ 0.6 & 0.05 & 0.6 \\ 0.6 & 0.6 & 0.05 \end{bmatrix}$$

The negative edges probability matrix for 2-level polarization $P^{(-)}_{2-level}$ is:

$$P^{(-)}_{2-level} = \begin{bmatrix} 0.9 & 0.1 & 0.1 \\ 0.1 & 0.9 & 0.1 \\ 0.1 & 0.1 & 0.9 \end{bmatrix}$$

- We start with the case of $\alpha = \mathbf{0.95}$, where we have 95% of the negative structure being a consequence of the participation in the positive community structure (i.e. members of a team generally like and agree with each other while dislike and disagree with members of a different team). We here observe the classical case of 1-LEVEL polarization. Results are provided in Figure 1.

- We continue with the case of $\alpha = \mathbf{0.05}$, where we have 5% of the negative structure being a consequence of the participation on the positive community structure while 95% of the negative structure comes from the community structure, formed by shared animosity. We here witness the more nuanced case of 2-LEVEL polarization. Results are provided in Figure 2.
- We finally set $\alpha = \mathbf{0.50}$, where we have an equal presence of negative structure based both on 1-LEVEL and 2-LEVEL polarization. Results are provided in Figure 3.

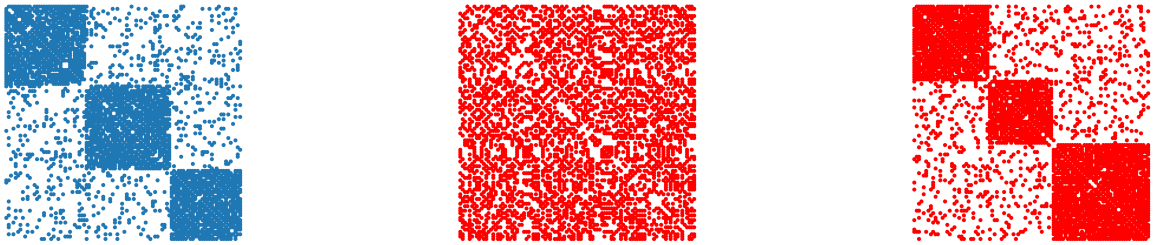


(a) Positive links adjacency matrix re-ordered based on the positive community memberships $\mathbf{c}^{(+)}$

(b) Negative links adjacency matrix re-ordered based on the positive community memberships $\mathbf{c}^{(+)}$ (1-LEVEL)

(c) Negative links adjacency matrix re-ordered based on the negative community memberships $\mathbf{c}^{(-)}$ (2-LEVEL)

Figure 1: INFERRED COMMUNITY MEMBERSHIPS: (a)-(c) visualization the re-ordered adjacency matrices based on the ground truth community memberships for $\alpha = 0.95$



(a) Positive links adjacency matrix re-ordered based on the positive community memberships $\mathbf{c}^{(+)}$

(b) Negative links adjacency matrix re-ordered based on the positive community memberships $\mathbf{c}^{(+)}$ (1-LEVEL)

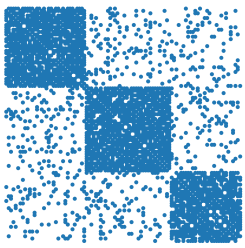
(c) Negative links adjacency matrix re-ordered based on the negative community memberships $\mathbf{c}^{(-)}$ (2-LEVEL)

Figure 2: INFERRED COMMUNITY MEMBERSHIPS: (a)-(c) visualization of the re-ordered adjacency matrices based on the ground truth community memberships for $\alpha = 0.05$.

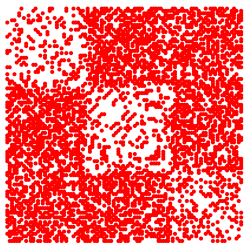
2 EXPERIMENTAL SETUP

2.1 Training details

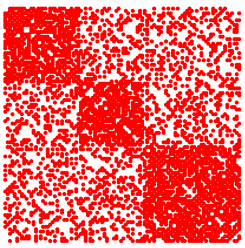
All experiments for SGAAE have been conducted on an 8 GB NVIDIA RTX 2070 Super GPU. In addition, we adopted the Adam optimizer (Kingma and Ba, 2014) with an initial learning rate of lr = 0.005 and for 3000 iterations and a learning rate scheduler. To scale the analysis to larger graphs, we use random sampling with a sample size of around 3000 nodes for all networks. To avoid the effect of local minimas we run our model five times and choose the performance of the model with the best log-likelihood on the training data. The reported results are averaged over three runs, while standard deviations were found in the scale of 10^{-3} and were omitted.



(a) Positive links adjacency matrix re-ordered based on the positive community memberships $\mathbf{c}^{(+)}$



(b) Negative links adjacency matrix re-ordered based on the positive community memberships $\mathbf{c}^{(+)}$ (1-LEVEL)



(c) Negative links adjacency matrix re-ordered based on the negative community memberships $\mathbf{c}^{(-)}$ (2-LEVEL)

Figure 3: INFERRED COMMUNITY MEMBERSHIPS: (a)-(c) visualization the re-ordered adjacency matrices based on the ground truth community memberships for $\alpha = 0.50$

2.2 Datasets and baselines

In our experimental set-up we are using four real-world signed networks. These are, *Reddit* which is extracted from hyperlinks capturing directed pairwise connections between communities in a social platform (Kumar et al., 2018), *Twitter* that is a social network of tweets focusing on the debate about the reform of the Italian Constitution (Ordozgoiti et al., 2020), *wikiRfA* and *wikiElec* that are election networks representing users as entities and their connections expressing different types of votes, i.e. supporting, neutral, opposing, about them being elected as administrators on Wikipedia (West et al., 2014; Leskovec et al., 2010). In addition we consider the following baselines: (**i**) POLE (Huang et al., 2022) which extracts embeddings by applying a decomposition on the auto-covariance similarity matrix of signed random walks, (**ii**) SLF (Xu et al., 2019) that learns representations based on two latent factors capturing positive and negative interactions respectively, (**iii**) SIGAT (Huang et al., 2019) that is a GNN-based method leveraging a graph attention mechanism for updating the node embeddings, (**iv**) SIDE (Kim et al., 2018) that extracts pair-wise signs based on balance theory and random walks, (**v**) SIGNET (Islam et al., 2018) which encodes proximity of signed pairwise interactions based on deep neural networks, (**v**) S-SGCN a spectral signed graph convolutional network using the signed Laplacian capable of retaining low-frequency information, or preserving high-frequency components (Singh and Chen, 2022) and finally, (**vi**) SLIM (Nakis et al., 2023) which is a latent distance-based model that extracts a unified embedding space based on the Skellam likelihood. Differently from our method, the latter distance-based model does not consider neural network-based representations, among other distinct technical aspects described in the manuscript.

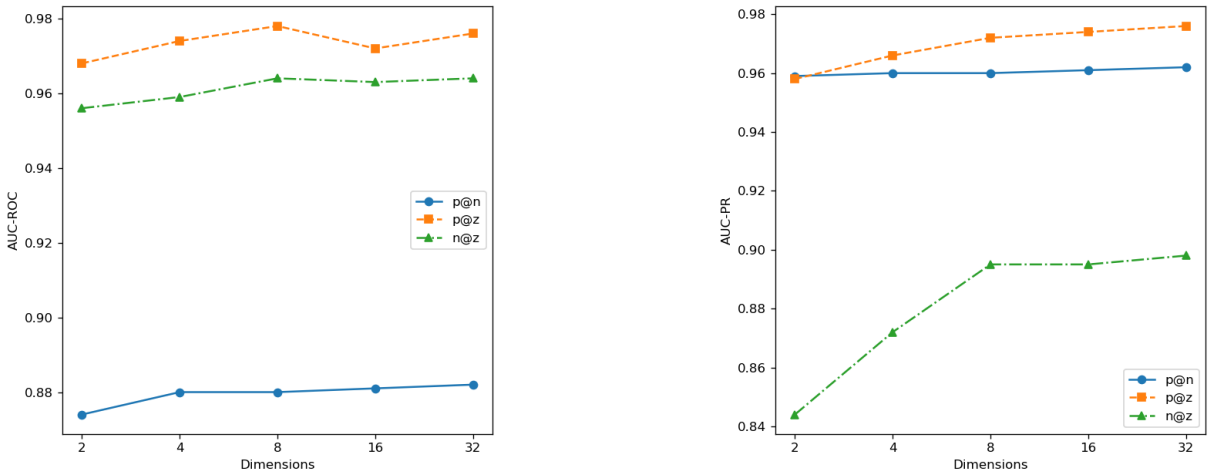
2.3 GNN details

For each encoder, we use two GCN layers with 64 input and output hidden dimensions. For each network node, we use the 32 first eigenvectors of the signed Normalized Laplacian of the network as features. For all MLPs, we use 64 hidden dimensions.

3 ADDITIONAL RESULTS

3.1 Effect of dimensionality in predictive performance

We next provide the effect of dimensionality (i.e. number of archetypes) in SGAAE for its predictive performance. We observe in Figure 4 that the model is not very sensitive in the number of dimensions, but it slightly benefits from a higher number of introduced archetypes.



(a) AUC-ROC (b) AUC-PR

Figure 4: Effect of Dimensionality: *WikiElec*.

3.2 Additional performance metrics for link prediction

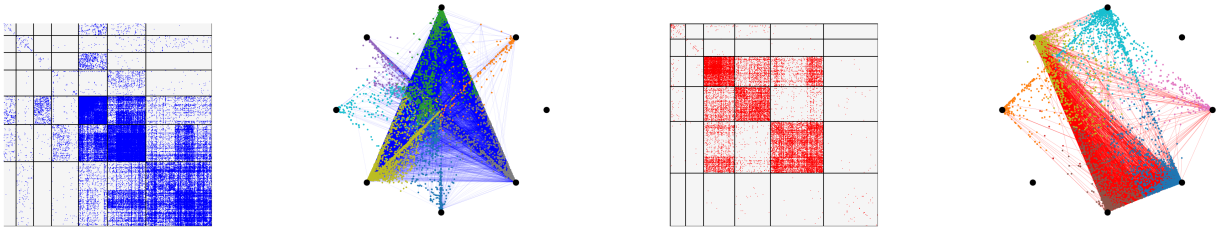
The AUC-ROC results are provided in Table 2, as a complementary metric of the AUC-PR score provided in the main paper. We once more observe favorable performance for the proposed SGAAE against all baselines.

Table 2: Area Under Curve (AUC-ROC) scores for representation size of $K = 8$.

	WikiElec			*WikiRfa*			*Twitter*			*Reddit*		
Task	$p@n$	$p@z$	$n@z$	$p@n$	$p@z$	$n@z$	$p@n$	$p@z$	$n@z$	$p@n$	$p@z$	$n@z$
POLE	.809	.896	.853	.904	.921	.767	.965	.902	.922	OOM	OOM	OOM
SLF	**.888**	.954	.952	**.971**	.963	.961	.914	.877	.968	**.729**	.955	.968
SiGAT	.874	.775	.754	.944	.766	.792	**.998**	.875	.963	.707	.682	.712
SIDE	.728	.866	.895	.869	.861	.908	.799	.843	.910	.653	.830	.892
SignNet	.841	.774	.635	.920	.736	.717	.968	.719	.891	.646	.547	.623
SpectralGCN	.864	.942	.932	.942	.949	.942	.861	.919	.907	.688	.929	.955
SLIM	.862	.965	.935	.956	**.980**	.960	.988	**.963**	.972	.667	.955	**.978**
SGAAE	.880	**.978**	**.964**	.968	.986	**.980**	.988	.961	**.977**	.711	**.957**	.975

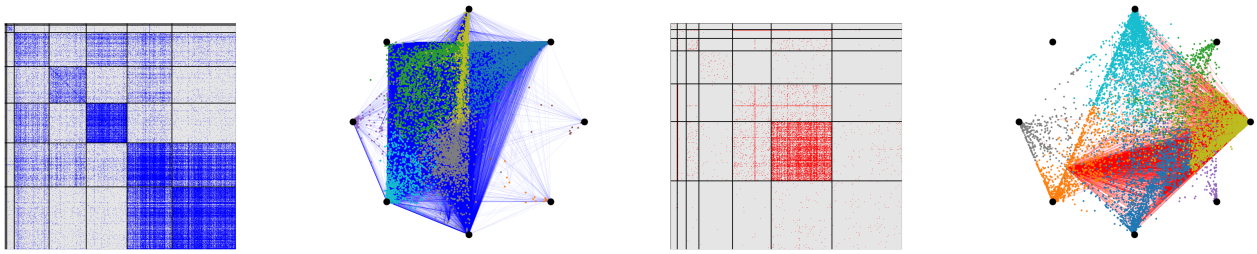
3.3 Additional network visualizations

We provide in Figure 5 and Figure 6 additional visualizations, focused on both the positive and negative ties, that validate the successful characterization over latent communities in the WikiElec, and Twitter networks respectively.



(a) Positive Adjacency (b) Positive membership space (c) Negative Adjacency (d) Negative membership space

Figure 5: Network Visualizations: for the *WikiElec* network where we show the re-ordered adjacency matrix based on the maximum positive/negative memberships to the archetypes—the positive/negative membership space where essentially we visualize the soft-memberships over the archetypes in a circular plot where each archetype is positioned every $\frac{2\pi}{K}$ rads.



(a) Positive Adjacency (b) Positive membership space (c) Negative Adjacency (d) Negative membership space

Figure 6: Network Visualizations: for the *Twitter* network where we show the re-ordered adjacency matrix based on the maximum positive/negative memberships to the archetypes—the positive/negative membership space where essentially we showcase the soft-memberships over the archetypes in a circular plot where each archetype is positioned every $\frac{2\pi}{K}$ rads.

References

J. Huang, H. Shen, L. Hou, and X. Cheng. Signed Graph Attention Networks. In *Proceedings of the 28th International Conference on Artificial Neural Networks: Workshop and Special Sessions*, pages 566–577, 2019.

Z. Huang, A. Silva, and A. Singh. POLE: Polarized Embedding for Signed Networks. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining*, pages 390–400, 2022.

M. R. Islam, B. Aditya Prakash, and N. Ramakrishnan. SIGNet: Scalable Embeddings for Signed Networks. In *Proceedings of the 22nd Pacific-Asia Conference in Knowledge Discovery and Data Mining*, pages 157–169, 2018.

J. Kim, H. Park, J.-E. Lee, and U. Kang. SIDE: Representation Learning in Signed Directed Networks. In *Proceedings of the 2018 World Wide Web Conference*, pages 509–518, 2018.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

S. Kumar, W. L. Hamilton, J. Leskovec, and D. Jurafsky. Community interaction and conflict on the web. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 933–943. International World Wide Web Conferences Steering Committee, 2018.

J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. In *WWW*, page 641–650, 2010.

N. Nakis, A. Celikkanat, L. Boucherie, C. Djurhuus, F. Burmester, D. M. Holmelund, M. Frolcová, and M. Mørup. Characterizing polarization in social networks using the signed relational latent distance model. In F. Ruiz, J. Dy,