Learning Stance Embeddings from Signed Social Graphs

John Pougué-Biyong[1], Akshay Gupta[2], Aria Haghighi[2], Ahmed El-Kishky[2]
[1]University of Oxford, [2]Twitter Cortex
john.pougue-biyong@maths.ox.ac.uk,{akshayg,ahaghighi,aelkishky}@twitter.com

ABSTRACT

A key challenge in social network analysis is understanding the position, or stance, of people in the graph on a large set of topics. While past work has modeled (dis)agreement in social networks using signed graphs, these approaches have not modeled agreement patterns across a range of correlated topics. For instance, disagreement on one topic may make disagreement (or agreement) more likely for related topics. We propose the Stance Embeddings Model (SEM), which jointly learns embeddings for each user and topic in signed social graphs with distinct edge types for each topic. By jointly learning user and topic embeddings, SEM is able to perform cold-start topic stance detection, predicting the stance of a user on topics for which we have not observed their engagement. We demonstrate the effectiveness of SEM using two large-scale Twitter signed graph datasets we open-source. One dataset, TwitterSG , labels (dis)agreements using engagements between users via tweets to derive topic-informed, signed edges. The other, BirdwatchSG , leverages community reports on misinformation and misleading content. On TwitterSG and BirdwatchSG , SEM shows a 39% and 26% error reduction respectively against strong baselines.

KEYWORDS

stance embeddings, signed social graphs, signed graph datasets, edge-attributed graphs, stance detection

1 INTRODUCTION

Signed graphs (or networks) have been used to model support and opposition between members of a group of people, or community, in settings ranging from understanding political discourse in congress [22] to identifying polarization in social networks [11]. In such graphs, each node represents an individual in the community, a positive (+) edge indicates agreement between two community members and a negative (−) one denotes disagreement. For instance, Epinions [11] is a *who-trust-whom graph* extracted from the now-defunct online review site, where each edge represents whether one member has rated another as trustworthy (+) or not (−). The 108th US Senate signed graph [16] represents political alliances (+) or oppositions (−) between congressional members across 7,804 bills in the 108th U.S. Congress. Past work have leveraged signed graphs and insights from social psychology [5] in order to better understand and predict patterns of community interaction [11, 16].

Recent research in text-based stance detection has proven the benefits of capturing implicit relationships between topics, especially in cases where there are many topics at stake, and most with little training data [2, 3, 12]. One shortcoming of traditional signed graph analysis is that it reduces the interaction between any two individuals to a binary value of agreement (+) or disagreement (−). Interactions in communities may be much more complex and change depending on underlying context. In the U.S. senate, two senators may agree on bills related to climate change, but differ on taxation policy bills. In a sports community, two French football fans may support rival clubs, but will generally both support the national team at the World Cup. Most communities will have several different aspects, or *topics*, of discourse that have rich structure and dynamics within a community. For instance, in the French football fan example, it is very likely for someone to support the national team if we have observed support for a local club. This example and others highlight the value of modeling community stance across a range of topics [15].

In this work, we use *signed topic graphs* to represent (dis)agreement across topics of discourse with a community. Each edge represents a binary agreement value ($\{+, -\}$) with respect to a single topic t; the inventory of topics is assumed to be fixed and finite, but varies across applications.

Our proposed method, the Stance Embeddings Model (SEM), detailed in Section 3, leverages an extension of the node2vec algorithm [6] to signed topic graphs to learn embeddings for nodes as well as for topics. Learning member (node) and topic embeddings jointly enables us to represent topic-informed stance embeddings for each member, which can accurately predict member agreement across community topics (Section 5.4). This allows us to do zero-shot topic-stance prediction for a member, even when we haven't observed past engagement from the member on a topic (Section 5.5). As importantly, it allows us to capture implicit relationships between topics (Section 5.7).

We apply and evaluate our approach on two Twitter-based signed social graphs that we open-source alongside this work (see Section 4). For both of these datasets, we represent online interactions as a signed topic graph, where each node is a Twitter user[1] and each edge represents an interaction between users on a given topic. The TwitterSG dataset (Section 4.1) consists of ∼13M interactions (edges) between ∼750k Twitter users (nodes), spanning 200 sports-related topics; each edge represents one user replying to another user's Tweet or explicitly using the 'favorite' UI action (AKA, a *like*). This graph is ∼6x larger than the Epinions graph, which to the best of the authors' knowledge, is the largest publicly available signed social graph. The BirdwatchSG dataset instead leverages Birdwatch[2] annotations to indicate whether a user finds information on a Tweet to be misinformation or misleading or are rated helpful in clarifying facts (see Section 4.2 for details).

The core contributions of this paper are:

- **Stance Embeddings Model (SEM)**: Generalisation of node2vec to signed topic graphs. The model enables us to

[1]Since we focus on a social network application, we use the term *user* to refer to a member of the Twitter community.
[2]https://blog.twitter.com/en_us/topics/product/2021/introducing-birdwatch-a-community-based-approach-to-misinformation

consider both topic and (dis)agreement for each edge during training, allowing to understand how topics relate to each other, how users engage with topics, and how users relate to each other across topics, even for topics a user may not have engaged with.
- **Datasets** Two signed topic graph datasets built with Twitter data, suitable for future research on understanding topical stance in large-scale communities and valuable resource for the graph mining community.

2 RELATED WORK

2.1 Graph embeddings

Our work falls at the intersection of the literature on shallow graph (or network) embeddings and signed graph embeddings.

Shallow graph embeddings. Shallow graph embedding methods learn node embeddings when node features are unavailable [1, 6, 17, 18, 21]. They leverage the structure, i.e. the adjacency matrix, of the graph only. The two most popular variants are node2vec [6] and its specific case DeepWalk [18]. Node2vec and DeepWalk build on top of word2vec [13, 14], a word embedding technique in natural language processing. Node2vec generates second-order random walks on unsigned graphs, and learns node embeddings by training a skip-gram with negative sampling (SGNS) [14] to predict the surroundings of the input node. The learnt embeddings are such that nodes close in the graph are close in the embedding space. However, node2vec is not adapted for signed graphs because it is based on the homophily assumption (connected nodes should lie close in the embedding space) whereby in signed graphs agreeing nodes should be closer while disagreeing nodes farther apart.

Signed graph embeddings. To overcome the homophily limitation, SNE [27] generates uniform random walks on signed graphs (by ignoring the weights on the edges) and replaces the skip-gram model by a log-bilinear model. The model predicts the representation of a target node given its predecessors along a path. To capture the signed relationships between nodes, two signed-type vectors are incorporated into the log-bilinear model. SIDE [8] generates first-order random walks and defines a likelihood function composed of a signed proximity term to model the social balance theory, and two bias terms to mimic the preferential attachment theory. SiNE [24] is a deep neural network-based model guided for social theories. SiNE maximises the margin between the embedding similarity of friends and the embedding similarity of foes. StEM [20] is a deep learning method aiming at learning not only representations of nodes of different classes (e.g. friends and foes) but also decision boundaries between opposing groups . Hence, unlike other methods, e.g. SiNE, which are distance-based (thus, use only local information), StEM attempts to incorporate global information.

Overall, to learn node embeddings, SNE and SIDE generate random walks but do not use the skip-gram in node2vec to learn parameters, while SiNE and StEM use a margin loss function and decision boundary method respectively.

Our model. Our approach extends the traditional skip-gram objective of word2vec and node2vec to signed graphs. We facilitate this by ensuring that each training example is constructed via a sign-informed random walk. Leveraging the skip-gram architecture not only provides scalability advantages, but also the flexibility to extend the embedding process to effectively utilize edge attributes. While all mentioned related works are edge-attribute agnostic, our proposed method can leverage edge attributes such as topics in the form of contextual topic embeddings. We argue that incorporating edge attributes, such as topics, in the embedding process can benefit understanding stance in signed social-network interactions.

2.2 Signed graphs in social media

In many real-world social systems, relations between two nodes can be represented as signed graphs with positive and negative links. Early signed graphs are derived from observations in the physical world such as the relationships among Allied and Axis powers during World War II [4].

The development of social media has enabled the mining of larger, signed social graphs. Typically, signed graphs in social media represent relations among online users where positive links indicate friendships, trust, and like, whereas negative links indicate foes, distrust, and dislike. Signed graphs in social media often have thousands of users and millions of links, and they are usually sparser than physical world signed graphs.

Existing datasets. Epinions, Slashdot [11], Wiki-Rfa [25], BitcoinOtc, BitcoinAlpha [9, 10] are the largest and most widely used signed graphs used for benchmarking signed graph embeddings methods. Epinions.com[3] was a product review site where users can write reviews for various products with rating scores from 1 to 5. Other users could rate the helpfulness of reviews. Slashdot[4] is a technology news platform on which users can create friend and foe links with other users. For a Wikipedia editor to become an administrator, a request for adminship (RfA) must be submitted, and any Wikipedia member may cast a supporting, neutral, or opposing vote. This induces a directed, signed graph Wiki-Rfa [25] in which nodes represent Wikipedia members and edges represent votes. BitcoinOtc and BitcoinAlpha [9, 10] are who-trusts-whom graphs of users who trade using Bitcoin on online platforms. Since Bitcoin users are anonymous, there is a need to maintain a record of users' reputation to prevent transactions with fraudulent and risky users. Platforms' members can rate each other members positively or negatively.

Our datasets. We curate and open-source two real-world signed social graphs with attributed (topics) edges. TwitterSG is a signed edge-attributed, multi-edge, directed graph with 12,848,093 edges between 753,944 Twitter users (nodes), spanning 200 sports-related topics: teams, sports, players, managers, and events (e.g. Los Angeles Lakers, Basketball, Cristiano Ronaldo, Zinedine Zidane, Olympics). TwitterSG contains ~6x more nodes than Epinions, the largest publicly available signed graph. BirdwatchSG is a signed edge-attributed, multi-edge, directed graph with 441,896 edges between 2,987 Birdwatch participants (nodes) based in the USA, and spanning 1,020 diverse topics prone to misleading content and/or partisanship (e.g. COVID-19, US Presidential Elections). Table 1 provides statistics on the datasets.

[3] https://en.wikipedia.org/wiki/Epinions
[4] https://slashdot.org/

Table 1: Statistics of signed graph datasets. The bottom two denote datasets released as part of this work.

Dataset	$\|V\|$	$\|E\|$	$\%_{\|E_-\|}$
BitcoinAlpha	3,783	24,186	7%
BitcoinOtc	5,881	35,592	9%
Epinions	131,828	841,372	15%
Slashdot	77,357	516,575	23%
Wiki-Rfa	10,835	159,388	22%
BirdwatchSG	2,987	441,986	37%
TwitterSG	753,944	12,848,093	10%

3 SEM: LEARNING STANCE EMBEDDINGS ON SIGNED TOPIC GRAPHS

3.1 Preliminaries

Let $G = (V, E)$ be a signed (un)directed topic graph: each edge has a topic t, and a sign of $-$ or $+$. We use T to denote the finite set of topics t. Note that there can be multiple edges between users corresponding to different topic interactions. We define $G_t = (V_t, E_t)$ the subgraph of G which contain all the edges with topic t. We aim to learn a node mapping function $f_V : V \rightarrow \mathbb{R}^d$, and a topic embedding function $f_T : T \rightarrow \mathbb{R}^d$.

Our approach will implicitly define embeddings for each edge using learned node and topic embeddings. For an edge (u, v) with topic t, we combine the source embedding and topic embedding using $\sigma(f_V(u), f_T(t))$; see Section 5.1 for choices of σ considered. This transformed source node embedding is combined with the target node embedding using an operator $\Phi(\cdot, \cdot)$ from Table 2. We evaluate these edge embeddings compared to other signed graph edge embeddings in Section 5, but for the remainder of this section, we will detail how we learn the node and topic embedding functions f_V and f_T.

3.2 Training data creation

As we apply the skip-gram objective to graph data via random walks, our work can be considered an extension to node2vec [6]. However, while node2vec only operates on unsigned homogeneous graphs, our embedding approach naturally incorporates signed edges as well as edge attributes such as topics.

Given an input signed topic graph, we outline how we create training examples to learn node and topic embeddings using the skip-gram objective.

Random walks on edge-attributed graphs. We first iterate through each topic-specific subgraph G_t, and mask the edge weights yielding a topic-graph $G'_t = (V_t, E'_t)$ where all edges are unsigned and unweighted. We follow the sampling procedure of [6], and define a second-order random walk with two parameters p and q that guide the walker on G'_t. Let us consider a walker that just traversed edge (s, u) and now resides at node u. The walker next decides to walk to edge (u, v) with the unnormalised transition probability π_{uv}:

$$\pi_{uv} = \begin{cases} \frac{1}{p} & \text{if } v = s \\ 1 & \text{if } d_{sv} = 1 \\ \frac{1}{q} & \text{if } d_{sv} = 2 \end{cases} \tag{1}$$

where d_{sv} is the shortest path distance between nodes s and v. p and q are return and in-out parameters respectively, and control how fast the walk explores and leaves the neighborhood of starting node s. For example, $q < 1$, means the walker is more inclined to visit nodes which are further away from node s.

For each node n in G'_t, we simulate r random walks of fixed length l starting at n. At every step of the walk, sampling is done based on transition probabilities defined in Eq. 1.

Creating signed contexts. In node2vec, the contexts of a source node u are the nodes surrounding it in the walks. The context vocabulary C is thus identical to the set of nodes V. This effectively embeds connected node close to each other in the embedding space. However, in signed graphs, agreeing nodes (linked with positive edges) should be embedded in close proximity while disagreeing nodes (linked with negative edges) should be farther away. We incorporate these insights into our skip-gram objective.

Unlike with node2vec, whereby a source node predicts context node, we propose to predict *sign and node* as contexts. In other words, we predict not only the context node, but also whether the source node agrees or disagrees with them on a given topic. While the context node is determined by the random walk, there may not be a signed edge between a source node and context node for that topic. To infer whether or not a source and context node agree on some topic, we apply Heider's social balance theory [5].

Let t be an arbitrary topic, and consider the graph G_t depicted in Figure 1. Assuming a random walk sampled via the procedure described above, we have a sequence of nodes. Using a window of size k around a source node u_0, $2k$ context nodes are produced from the walk: k before u_0 and k after: $(u_{-k} \ldots u_0 \ldots u_{+k})$. In addition we compute the inferred sign, $S(u_0, u_i)$, between our source node and the i_{th} context node as follows:

$$S(u_0, u_i) = \begin{cases} \prod_{m=i+1}^{0} w_{u_{m-1}u_m} & i < 0 \\ \prod_{m=1}^{i} w_{u_{m-1}u_m} & i > 0 \end{cases} \tag{2}$$

where w_{uv} is the weight, $+1$ or -1, between nodes u and v.

As seen in Equation 2, we can leverage Heider's social balance theory to assign each context node a sign with respect to the source node. In simple terms we have three rules:

(1) the friend (+) of my friend (+) is my friend (+)
(2) the friend (+) of my enemy (−) is my enemy (−)
(3) the enemy (−) of my enemy (−) is my friend (+)

Equation 2 applies this to (dis)agreements over topics and as such, we can compute the (dis)agreement sign between the source node and a context node simply by multiplying the edge signs between the source and context as defined by the random walk between them.[5]

[5] Although random walk generations and social balance theory are applied on each topical graph independently, we show in section 5.7 that our model is able to learn associations between topics during training.

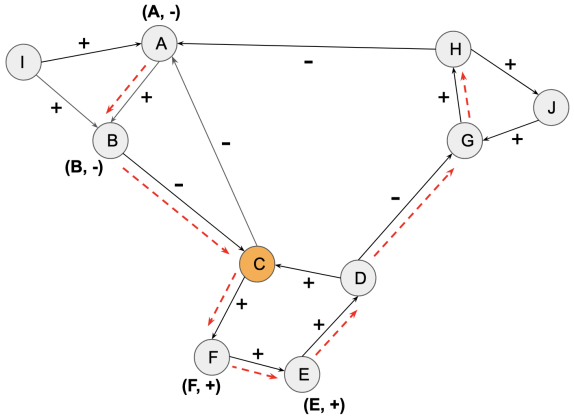

Figure 1: A sample random walk (in red) on a signed graph. The corresponding sign-informed contexts for source node $u_o = C$ are shown in bold (assuming a window of size 2).

By incorporating these signed (dis)agreements with the source node alongside each context node, our skip-gram objectives need to not only predict the context node, but also whether or not the context node agrees with the source node on a topic. As such, node proximity and stance both influence a node's embedding.

3.3 Learning node & topic embeddings

The training examples are composed of a source node u, a topic t, and a set of contexts $C_t(u)$ where contexts consist of *(node, sign)* pairs. We associate embedding vectors W_u, W_c, and W_t for the source, context (node-sign pair), and topics respectively; these vectors are parameters to be learned. In Fig. 2, we visualize this topic-aware skip-gram architecture as a generalisation of the original skip-gram neural network architecture.

To learn these vectors, we generalise the SkipGram objective to incorporate topic information t as follows:

$$\max_{W} \sum_{t \in T} \sum_{u \in V} \left[-\log Z_{u,t} + \sum_{c \in C_t(u)} W_c \cdot \sigma(W_t, W_u) \right] \quad (3)$$

where $Z_{u,t} = \sum_{c' \in C_t} \exp(W_{c'} \cdot \sigma(W_t, W_u))$, with $\sigma(\cdot,\cdot)$ an operation over topic and node embedding vectors (e.g. addition of both vectors). As the partition function $Z_{u,t}$ is expensive to compute, we approximate it using negative sampling [14]. Moreover, the sign in any context c of Equation 3 is derived from Equation 2.

4 DATASETS

In this section, we describe two new social-network signed topic graphs that we curate and open-source alongside our work. Both datasets are fully anonymized without personally identifiable information.

4.1 TwitterSG

Twitter Signed Graph, or TwitterSG , is a signed, directed, edge-attributed graph of users, drawn from Twitter interactions. A positive-signed edge exists from user A to user B if user A liked a tweet

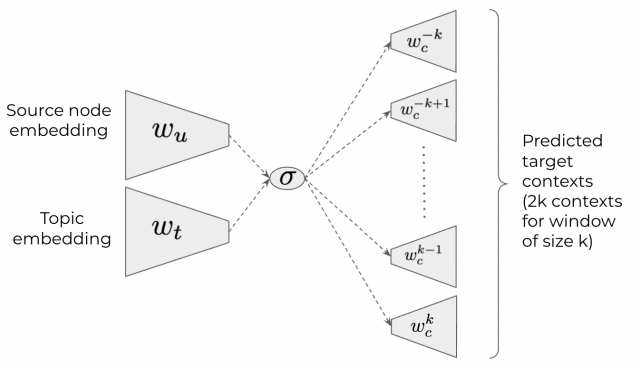

Figure 2: Skip-gram architecture for our model (SEM). Takes source node and topic as input and predicts sign-aware contexts.

posted by user B. A negative-signed edge exists from user A to user B if user A expressed opposition towards user B's tweet, e.g., by replying *I disagree with you.* The topic of an edge from user A to user B is determined by the topic of user B's tweet, also called the *target tweet.* Tweet topics were inferred with a topic classifier provided and used in production by Twitter; we restrict interactions in TwitterSG to sports-related topics (e.g., sports teams, players, managers, or events). The tweets related to these interactions were published between 20th May (Ice Hockey World Championships) and 8th August 2021 (closing date of the 2020 Tokyo Olympic Games), and collected via Twitter API.

Several challenges arise when attempting to build a large signed graph with interactions on Twitter. First, the graph may be extremely sparse due to the number of active users and the skewed distribution of tweets per user. Second, opposition mostly goes silent (the user may keep scrolling if they do not agree with a statement) or is expressed via reply to a tweet, which requires more effort than clicking a *like* button to express support. For this reason, there is a substantial unbalance between the amount of support and opposition signals. And lastly, opposition in a tweet may be implicit.

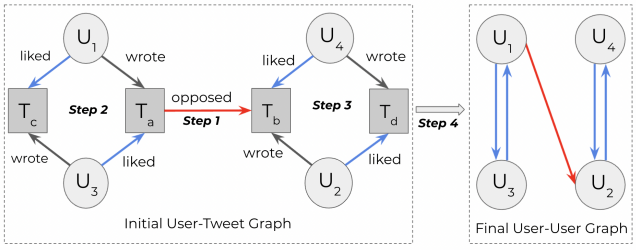

Figure 3: Steps involved in building TwitterSG . The final user-user graph is obtained following step 4 of Section 4.1.

To overcome these challenges, we adopt a multi-step strategy to create a user-tweet graph (Fig. 3), that we project onto a user-user graph:

(1) We curated a list of high-precision English and French expressions which express clear opposition (e.g. "I disagree" and "you're wrong")[6]. We retained all sports-related tweets T_a containing at least one of these expressions, and the tweets T_b they replied to. For the sake of clarity, tweet T_a (T_b) is posted by user U_1 (U_2).
(2) To control the graph sparsity, we retained all users $U3$ who both (i) wrote a tweet T_c liked by user $U1$, and (ii) liked the tweet T_a (opposition tweet) written by user $U1$.
(3) Similarly, we retained all users $U4$ who both (i) wrote a tweet T_d liked by user $U2$, and (ii) liked the tweet T_b written by user $U2$. At this stage, the positive interactions largely outnumbered the negative ones (300k negative interactions for more than 100M positive ones). Filtering out a large portion of positive edges would increase the share of negative edges but would decrease the number of users. Conversely, filtering in a large portion of positive edges would push the share of negative edges close to 0 but increase the number of users. We found a trade-off cut by selecting a share of likes (retrieved in steps (2) and (3)) so that the share of negative edges in our graph is close to 10%. We ranked the topics by decreasing frequency and filtered out all the tweets not related to the top 200 topics.
(4) We project the resulting user-tweet graph onto a user-user graph. We anonymise all the nodes (users) and edges (tweets).

Eventually, the edge data of the final graph is provided under the format depicted in Fig. 4. TwitterSG contains 753,944 nodes (users),

source_node	*target_node*	*topic*	*weight*
1	6	Copa America	+1
1	6	NFL	-1
4	5	Kylian Mbappe	-1

Figure 4: TwitterSG . An edge represents that the source node (user) has a positive (+) or negative (−) stance towards the target node (user) for the given topic.

200 topics and 12,848,093 edges. Among these edges, 9.6% are negative (opposition) and 90.4% are positive. Most frequent topics are depicted in Figure 6. There may be several edges between two nodes (several interactions, several topics).

4.2 BirdwatchSG

Birdwatch Signed Graph, or BirdwatchSG , is a signed, directed, edge-attributed graph of users, drawn from note ratings on Birdwatch[7]. Birdwatch is a pilot launched by Twitter in January 2021 in the USA to address misleading information on the platform, in a community-driven fashion: the Birdwatch participants can identify information in tweets they believe is misleading and write notes that provide informative context. They can also rate the helpfulness (either *helpful*, *somewhat helpful*, or *not helpful*) of notes added by other contributors. All Birdwatch contributions are publicly available on the Download Data page of the Birdwatch site[8] so that anyone in the USA has free access to analyse the data.

Starting with Birdwatch data from January to July 2021, we create a positive (negative) edge from participant U_1 to U_2 if participant U_1 rated a note written by participant U_2 as *helpful* (*not helpful*). We filter out the *somewhat helpful* ratings. The topic associated with an edge is the topic of the tweet the note refers to. We anonymise all the nodes and edges. Eventually, the edge data of the final graph is provided under the format depicted in Fig. 5.

source_node	*target_node*	*topic*	*weight*
1	6	US Politics	+1
1	6	Ted Cruz	-1
4	5	Twitter	-1

Figure 5: BirdwatchSG . An edge represents that the source node (user) has a positive (+) or negative (−) stance towards the target node (user) for the given topic.

The graph contains 2,987 nodes (users), 1,200 topics and 441,986 edges. Among these edges, 36.9% are negative (opposition) and 63.1% are positive. Most frequent topics are depicted in Figure 6. There may be several edges between two nodes (several interactions, several topics).

5 EXPERIMENTS

In this section, we evaluate the embeddings produced by our SEM method (Section 3) and compare its performance to three state-of-the art signed graph embedding models on our TwitterSG and BirdwatchSG datasets (Section 4).

5.1 Embedding Models

SEM variants. We evaluate three variants of SEM, each of which corresponds to a different choice of σ function to combine node and topic embeddings (Section 3):

- SEM-*mask*: The topic information is ignored. This corresponds to $\sigma(W_t, W_u) = W_u$ in the first layer of the topic-aware skip-gram architecture, Fig. 2.
- SEM-*addition*: The topic and node embeddings are added in the first layer of the topic-aware skip-gram architecture (Fig. 2), i.e., $\sigma(W_t, W_u) = W_t + W_u$.
- SEM-*hadamard*: The topic and node embeddings are combined via element-wise multiplication (hadamard) in the first layer of the topic-aware skip-gram architecture, i.e., $\sigma(W_t, W_u) = W_t \times W_u$.

Note that the SEM variants only change how the user and topic embedding are combined during node2vec training (Section 3.3).

[6]Key expressions provided in appendix A and open-sourced dataset.
[7]https://blog.twitter.com/en_us/topics/product/2021/introducing-birdwatch-a-community-based-approach-to-misinformation
[8]https://twitter.github.io/birdwatch/

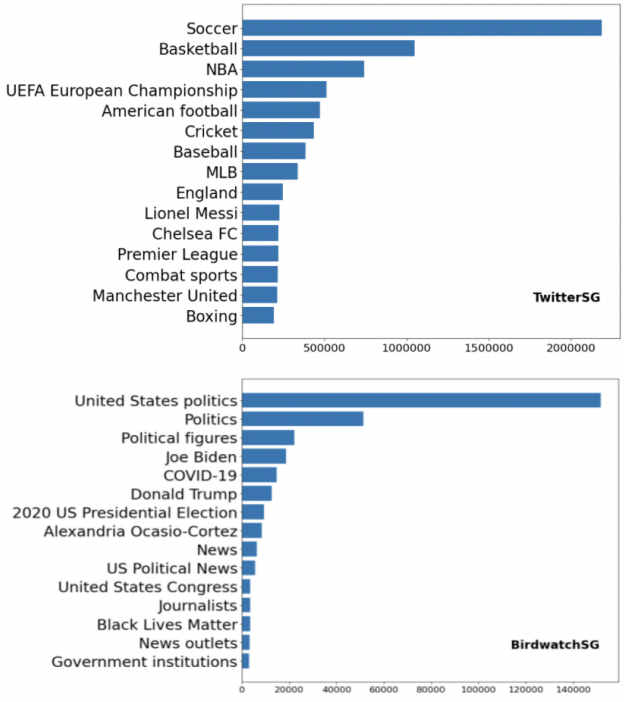

Figure 6: Top-15 topics in TwitterSG (top) and BirdwatchSG (bottom) respectively, ordered by decreasing frequency.

Baselines. We compare SEM to three state-of-the-art signed graph embedding methods described in Section 2.1: StEM [20], SIDE [8], SiNE [24]. Like SEM-*mask*, these three methods are topic agnostic and were only tested on signed graphs lacking topics, or other attributes, on edges.

5.2 Training setup

We set the node embedding dimension (d) to 64 for all methods and experiments. For SEM variants, we set walks per node $r \in \{5, 10, 20, 80\}$, walk length $l = 40$, context size $k = 5$, return parameter $p = 1.5$, in-out parameter $q = 0.5$, negative sample size to 20, subsampling threshold to 1e-5, and the optimisation is run for 1 to 5 epochs. For two given users and a given topic, edge weights are summed and the overall topical edge weight is set to +1 if the sum is positive, and -1 otherwise. For baseline methods, we use the same parameter settings as those suggested in their respective papers. The edge topic information is masked for baselines and SEM-*mask*.

5.3 Evaluation setup

We follow previous work by evaluating our method, SEM, and baselines on a signed link prediction task [8, 20, 24]. In signed link prediction, we are given a signed graph where the sign, or agreement value, on several edges is missing and we predict each edge's sign value using the observed edges. In particular, we formulate link sign prediction as a binary classification task using embedding learned from each method as follows. For each dataset, we perform 5-fold cross-validation (80/20% training/test set) and evaluate with mean AUC over the 5 folds. For all approaches, we create edge embeddings by combining node embeddings using $\Phi(u_1, u_2)$ using operations from Table 2. Note that this means for topic-aware SEM variants we do not explicitly use the topic embedding for evaluation.

Using the edge representations in the training set, we fit a binary classifier to predict edge signs on the test set. Due to the sign imbalance sign in the edge data, we downsample the positive signs when fitting the classifier.

Table 2: Operations (Φ) to produce edge embeddings from node embeddings for evaluation (Section 5.3)

Operation	Output
hadamard	$w[i] = u_1[i] \times u_2[i]$
ℓ_1	$w[i] = \lvert u_1[i] - u_2[i] \rvert$
ℓ_2	$w[i] = (u_1[i] - u_2[i])^2$
Average	$w[i] = \frac{1}{2}(u_1[i] + u_2[i])$
Concatenation	$w = u_1 \otimes u_2$

Table 3: Mean AUC from 5-fold CV on stance detection using nearest neighbors (kNN) and logistic regression (LR) on edge embbedings to predict stance (Section 5.3).

	TwitterSG			BirdwatchSG		
	kNN		LR	kNN		LR
	$k = 5$	$k = 10$		$k = 5$	$k = 10$	
SiNE	86.0	86.6	61.1	86.4	80.6	76.8
StEM	91.1	91.2	84.5	90.7	88.0	87.7
SIDE	91.0	87.5	82.1	92.6	90.0	82.7
SEM-mask	90.5	92.3	84.4	92.4	90.4	86.6
SEM-addition	**94.0**	**95.3**	**88.1**	**94.6**	**92.9**	**91.5**
SEM-hadamard	91.4	92.7	83.8	94.1	92.3	91.3

5.4 Stance detection: predicting link sign

In Table 3, we report results for SEM variants and baselines using both nearest neighbors (kNN) and logistic regression (LR) classification on edge embeddings. For each approach, we report the best value over choices of translation operator $\Phi(\cdot, \cdot)$ from Table 2.

On TwitterSG and BirdwatchSG , SEM-*mask*, the topic-agnostic version of SEM, shows better or competitive performance with the three baselines. The topic-aware SEM variants significantly outperform topic-agnostic baselines on both datasets and across both edge embedding classifiers. On TwitterSG , SEM-addition improves the AUC by 2.9% and 3.0% the AUC for the $k = 5$ and $k = 10$ kNN classifiers respectively, compared to the best performing topic-agnostic method. On BirdwatchSG , SEM-addition improves the AUC by 2.0% and 2.5% the AUC at $k = 5$ and $k = 10$ respectively, compared

to the best performing topic-agnostic method. These results demonstrate that SEM learns improved node embeddings for signed edge prediction.

5.5 Cold-start topic-stance detection

One important advantage of learning user and topic embeddings jointly is the potential for predicting the stance of a user on topics for which we have not observed their engagement. We investigate the performance of methods on this 'cold start' subset of test samples (u_1, u_2, w, t) such that the engagement of user u_1 or u_2 on topic t was not observed during training. In other words, there is no training sample $(u_1, ., ., t)$ or $(., u_2, ., t)$. This represents 28% and 17% of the test data for TwitterSG and BirdwatchSG respectively (average over 5 folds).

In Table 4, we present signed edge prediction AUC results limited to only 'cold start' using only nearest neighbors classification since this had better performance overall. Only SEM-addition is able to maintain performance across both datasets and edge embedding classifiers (compare to Table 3). This hints that, during training, SEM-addition learns topic relationships such that an observed disagreement on one topic affect the likelihood of disagreements (or agreements) for other topics.

Table 4: Mean AUC from 5-fold CV on cold-start stance detection using nearest neighbors (kNN). Results remain comparable to Table 3, demonstrating we can effectively still maintain high accuracy without prior data on a user's interactions with a topic. (Section 5.5).

	TwitterSG		BirdwatchSG	
	kNN		kNN	
	$k = 5$	$k = 10$	$k = 5$	$k = 10$
SiNE	83.0	84.3	84.2	80.1
StEM	92.8	90.2	89.9	88.4
SIDE	88.7	86.0	91.3	89.0
SEM-mask	87.9	90.0	90.5	90.1
SEM-addition	**95.1**	**96.1**	**95.7**	**93.9**
SEM-hadamard	90.4	90.1	93.4	92.4

5.6 Learning topic embeddings for topic-agnostic approaches

We investigate learning topic embeddings separately from user node embeddings in order to understand the value of jointly learning these as we propose. In order to explore this, we alter how we train a link prediction classifier for topic-agnostic approaches to also learn a topic embedding table. For topic-aware SEM-methods, we instead opt to freeze this topic embedding table to what was learned during graph embedding.

As depicted in Figure 7, for a given edge $e = (u_1, u_2, w, t)$, this classifier takes as input the pre-trained user embeddings u_1 and u_2 combined with a topic embedding t learned as part of this classifier

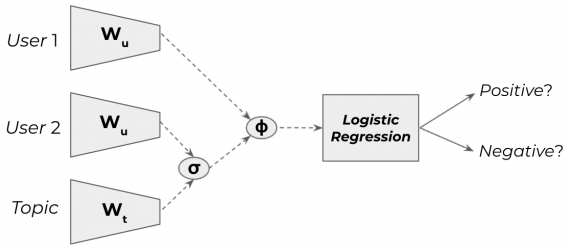

Figure 7: Logistic regression classifier for stance detection to investigate learning topic embeddings separately from user node embeddings (Section 5.6).

training process for topic-agnostic approaches. We combine these embeddings similarly to how we propose in Section 3.1 for training SEM: The user embedding u_2 and topic embeddings t are combined via functions $\sigma(\cdot, \cdot)$ matching the choices for σ that combine the graph-embedding learned user and topic embeddings defined in Section 5.1. The resulting vector is combined with u_1 user embedding vector via functions $\Phi(\cdot, \cdot)$ defined in Table 2. The resulting edge embedding is the input to the LR classifier. Note that we deliberately combine the topic embedding with the user embedding u_2 only. Indeed edge operations ℓ_1 and ℓ_2 in Table 2 involve the difference between source and target node embeddings. So combining the topic embedding into source and target embeddings would cancel each other out. Note also that when we set σ=mask, we effectively ignore this learned (or frozen topic embedding), reducing to the same setting for LR in Table 3. For other values of σ the topic embedding (learned or frozen from graph embedding) is used for edge prediction.

In Table 5, for each σ and graph embedding approach, we report the best AUC found over functions Φ. The performance of SEM-*addition* remains unmatched by the topic-agnostic methods even when topic-agnostic approaches learn topic embeddings during classifier training. Performance is still significantly degraded compared to our best results in Table 3, demonstrating that training topic and node embeddings in tandem remains the most beneficial way to incorporate context (topic) into stance detection on signed graphs. We do note however that for SEM-variant performance decreases if we use the learned topic embedding at test time.

5.7 Visualising stance embeddings

In Figure 8, we depict the topic embeddings obtained with SEM-*addition* trained on TwitterSG , and projected with tSNE [23]. We can discern clear clusters of topics associated to a specific sport (e.g. NFL, hockey, baseball) or group of sports (e.g. fighting sports: WWE, Wrestling). Among these clusters, we observe finer-resolution groups. For instance, English football clubs lie close to the Premier League topic. Karim Benzema, Antoine Griezmann and Paul Pogba are the closest neighbours to France, while Zinedine Zidane and Raphael Varane are close to Real Madrid CF. Michael Jordan and Kobe Bryant are closest neighbours. We observe similar patterns on BirdwatchSG topics, and with SEM-*hadamard* (not depicted due to space constraints). The presence of meaningful

Table 5: Mean AUC from 5-fold CV on stance detection where we learn topic embeddings learned during link prediction, separately from graph embedding (Section 5.6).

	TwitterSG			BirdwatchSG		
	σ = mask	add.	had.	mask	add.	had.
SiNE	61.1	62.0	65.3	76.8	77.3	76.8
StEM	84.5	84.6	80.2	87.7	88.1	81.5
SIDE	82.1	82.2	81.1	82.7	82.6	79.5
SEM-mask	84.4	84.3	80.1	86.6	86.3	82.3
SEM-addition	**88.1**	**88.1**	81.2	**91.5**	89.8	86.9
SEM-hadamard	83.8	78.7	87.4	**91.3**	88.4	86.9

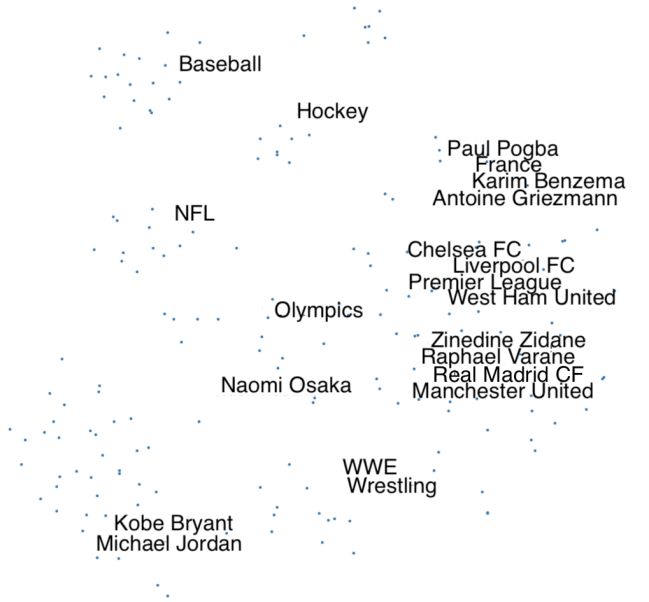

Figure 8: TwitterSG topic embeddings learnt by SEM. Related topics are embedded close to each other. Only certain representative topics are labelled for readability.

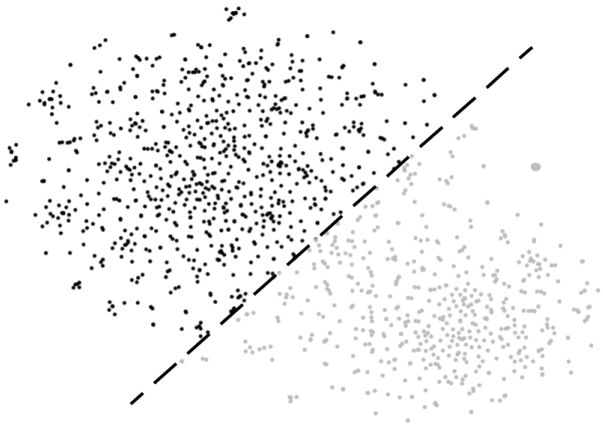

Figure 9: BirdwatchSG user embeddings learnt by SEM. Two opinion communities are observed in Birdwatch, in keeping with the polarised climate current present in the US political public debate.

Figure 10: BirdwatchSG edge embeddings derived from SEM's node embeddings. Negative edges are depicted in black, positive in grey. Edges with different signs form distinctive clusters.

topical clusters demonstrate the ability of our method to capture topic similarities when a diverse range of topics are discussed.

The US public debate is known to be politically polarised, and so are Birdwatch reports according to recent research [7, 19, 26]. Consequently, we expect to observe two major clusters of user embeddings. Figure 9 displays the user embeddings obtained with SEM-*addition* trained on BirdwatchSG , and projected with tSNE. The presence of two distinct opinion clusters prove the ability of our sign-informed context generation strategy to capture oppositions, and separate opposing views in the graph.

Further, we visually inspect the ability of the model to distinguish positive and negative edges. Let $e = (u_1, u_2, w, t)$ be an edge of topic t going from user u_1 to u_2 with weight $w \in \{-1, 1\}$. For the sake of visualisation, we define the embedding of edge e as the hadamard product of the two user embeddings $u_1 \times u_2$. Figure 10 displays the projected BirdwatchSG edge embeddings obtained with SEM-*addition* and tSNE. The positive (negative) edges are colored in blue (red). We observe distinct clusters of positive or negative edges, which confirms the capability of the model to discriminate positive and negative edges.

6 CONCLUSIONS

In this work, we introduce SEM, a framework for learning stance embeddings in signed, edge-attributed, social networks. Utilizing sign-informed random walks to generate training examples, we demonstrate how the scalable skip-gram objective can be successfully applied to learn signed-graph embeddings. Our approach is

flexible and can incorporate arbitrary edge-attribute such as topics, to provide context embeddings in edge interactions. Experimental results show that SEM embeddings outperform state-of-the-art signed-graph embedding techniques on two new datasets: TwitterSG and BirdwatchSG . We open-source these two datasets to the network mining community to spur further research in social network analysis.

REFERENCES

[1] Amr Ahmed, Nino Shervashidze, Shravan Narayanamurthy, Vanja Josifovski, and Alexander J Smola. 2013. Distributed large-scale natural graph factorization. In *Proceedings of the 22nd international conference on World Wide Web*. 37–48.

[2] Emily Allaway and Kathleen McKeown. 2020. Zero-Shot Stance Detection: A Dataset and Model using Generalized Topic Representations. *arXiv preprint arXiv:2010.03640* (2020).

[3] Emily Allaway, Malavika Srikanth, and Kathleen McKeown. 2021. Adversarial Learning for Zero-Shot Stance Detection on Social Media. *arXiv preprint arXiv:2105.06603* (2021).

[4] Robert Axelrod and D Scott Bennett. 1993. A landscape theory of aggregation. *British journal of political science* 23, 2 (1993), 211–233.

[5] Dorwin Cartwright and Frank Harary. 1956. Structural balance: a generalization of Heider's theory. *Psychological review* 63, 5 (1956), 277.

[6] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 855–864.

[7] Zexi Huang, Arlei Silva, and Ambuj Singh. 2021. POLE: Polarized Embedding for Signed Networks. arXiv:cs.SI/2110.09899

[8] Junghwan Kim, Haekyu Park, Ji-Eun Lee, and U Kang. 2018. Side: representation learning in signed directed networks. In *Proceedings of the 2018 World Wide Web Conference*. 509–518.

[9] Srijan Kumar, Bryan Hooi, Disha Makhija, Mohit Kumar, Christos Faloutsos, and VS Subrahmanian. 2018. Rev2: Fraudulent user prediction in rating platforms. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. ACM, 333–341.

[10] Srijan Kumar, Francesca Spezzano, VS Subrahmanian, and Christos Faloutsos. 2016. Edge weight prediction in weighted signed networks. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*. IEEE, 221–230.

[11] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Signed networks in social media. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 1361–1370.

[12] Rui Liu, Zheng Lin, Yutong Tan, and Weiping Wang. 2021. Enhancing Zero-shot and Few-shot Stance Detection with Commonsense Knowledge Graph. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. 3152–3157.

[13] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[14] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[15] Corrado Monti, Giuseppe Manco, Cigdem Aslay, and Francesco Bonchi. 2021. Learning Ideological Embeddings from Information Cascades. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 1325–1334.

[16] Zachary Neal. 2014. The backbone of bipartite projections: Inferring relationships from co-authorship, co-sponsorship, co-attendance and other co-behaviors. *Social Networks* 39 (2014), 84–97.

[17] Mingdong Ou, Peng Cui, Jian Pei, Ziwei Zhang, and Wenwu Zhu. 2016. Asymmetric transitivity preserving graph embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 1105–1114.

[18] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 701–710.

[19] Nicolas Pröllochs. 2021. Community-Based Fact-Checking on Twitter's Birdwatch Platform. *arXiv preprint arXiv:2104.07175* (2021).

[20] Inzamam Rahaman and Patrick Hosein. 2018. A method for learning representations of signed networks. In *Proceedings of the 14th International Workshop on Mining and Learning with Graphs (MLG)*.

[21] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*. 1067–1077.

[22] Matt Thomas, Bo Pang, and Lillian Lee. 2006. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Sydney, Australia, 327–335. https://aclanthology.org/W06-1639

[23] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[24] Suhang Wang, Jiliang Tang, Charu Aggarwal, Yi Chang, and Huan Liu. 2017. Signed network embedding in social media. In *Proceedings of the 2017 SIAM international conference on data mining*. SIAM, 327–335.

[25] Robert West, Hristo S Paskov, Jure Leskovec, and Christopher Potts. 2014. Exploiting social network structure for person-to-person sentiment analysis. *Transactions of the Association for Computational Linguistics* 2 (2014), 297–310.

[26] Taha Yasseri and Filippo Menczer. 2021. Can the Wikipedia moderation model rescue the social marketplace of ideas? *arXiv preprint arXiv:2104.13754* (2021).

[27] Shuhan Yuan, Xintao Wu, and Yang Xiang. 2017. Sne: signed network embedding. In *Pacific-Asia conference on knowledge discovery and data mining*. Springer, 183–195.

A LIST OF OPPOSITION EXPRESSIONS FOR TWITTERSG

To build TwitterSG , we mined all sports-related tweets containing at least one of the following expressions which express opposition:

- English expressions
  - *I disagree%*
  - *I do not (don't) agree%*
  - *you (u) are ('re) wrong%*
  - *%you (u) idiot%*
  - *%you (u) stupid%*
  - *%f*ck you (u)%*
  - *%cry more%*
- French expressions
  - *%ta gueule (tg)%*
  - *pleure(.)*
  - *tais toi(.)*
  - *abruti(.)*
  - *ratio(.)*
  - *tu dis nimp(.)*