Social Network Analysis using Knowledge-Graph Embeddings and Convolution Operations

Bonaventure C. Molokwu[1], Shaon Bhatta Shuvo[1], Ziad Kobti[1], and Narayan C. Kar[2]
[1]School of Computer Science, University of Windsor
[2]Electrical Engineering (ECE), University of Windsor
401 Sunset Avenue, Ontario N9B-3P4, Canada

Abstract

Link prediction and node classification in social networks remain open research problems with respect to Artificial Intelligence (AI). Innate representations about social network structures can be effectively harnessed for training AI models in a bid to predict ties; and detect clusters via classification of actors with regard to a given social network. In this paper, we have proposed a distinct hybrid model: Representation Learning via Knowledge-Graph Embeddings and Convolution Operations (RLVECO), which hybridizes the strengths of Knowledge-Graph Embeddings (VE) and Convolution Operations (CO) in extracting and learning meaningful features from social graphs via Representation Learning (RL). RLVECO utilizes an edge sampling approach for exploiting features of a social graph via learning the context of each actor with respect to its neighboring actors.

Introduction

Earth comprises several biosystems and these systems are affected by interactions between a range of biotic and abotic factors that control the dynamics of these biosystems. Interactions within and/or between biosystems is a strategy for survival, and these can be modelled via social networks. With regard to the recent advances in Artificial Intelligence (AI), we can effectively model and analyze real-world complex systems as social networks structures using appropriate AI techniques. Considering the impact of COVID-19 pandemic, Social Network Analysis (SNA) can serve as a handy technique for modelling, analyzing, and predicting the impact of this viral disease. However, social networks are complex and non-static structures which pose analytical challenges to Machine Learning (ML) and Deep Learning (DL) models. Hence, analyzing and learning underlying knowledge from communities, comprising social actors and their existent social ties/relationships, using given sets of standard still remain a crucial research problem in SNA. With the goal of solving prediction-based and classification-related problems in social network structures, we have introduced a distinct framework (RLVECO) possessing biform learning layers. RLVECO aims at learning the intrinsic patterns of relationship binding spatial social actors using a twofold Representation Learning (RL) layer as opposed to most state-of-the-art approaches based on a sole RL layer.

On one hand, the prediction of links brings about correlations and/or ties formation which increases the tendency for transitivity in social networks. On the other hand, the classification of nodes induces the formation of cluster(s), and clusters give rise to homophily in social networks. Our proposition is a unique clustering model based on an iterative learning approach; and it possesses the ability to learn the non-linear distributed representations [1] enmeshed in a social graph. Primarily, learning in RLVECO is achieved via semi-supervised training. The novelty of our work contains three (3) research contributions as stated below:

- Proposition of a DL-based and hybrid model, RLVECO, aimed at solving link prediction, node classification as well as community detection problems in SNA.
- Detailed benchmarking reports with respect to classic objective functions used for classification tasks.
- Comparative analyses, between RLVECO and state-of-the-art approaches, against standard real-world social networks.

Table 1: Benchmark datasets

Dataset	Classes → {label: 'description'}
CiteSeer	$G(V, E) = G(3312, 4732)$ {C1: 'Agents', C2: 'Artificial Intelligence', C3: 'Databases', C4: 'Information Retrieval', C5: 'Machine Learning', C6: 'Human-Computer Interaction'}
Cora	$G(V, E) = G(2708, 5429)$ {C1: 'Case_Based', C2: 'Genetic_Algorithms', C3: 'Neural_Networks', C4: 'Probabilistic_Methods', C5: 'Reinforcement_Learning', C6: 'Rule_Learning', C7: 'Theory'}
Facebook Page2Page	$G(V, E) = G(22470, 171002)$ {C1: 'Companies', C2: 'Governmental Organizations', C3: 'Politicians', C4: 'Television Shows'}
PubMed Diabetes	$G(V, E) = G(19717, 44338)$ {C1: 'Diabetes Mellitus - Experimental', C2: 'Diabetes Mellitus - Type 1', C3: 'Diabetes Mellitus - Type 2'}

Proposed Framework

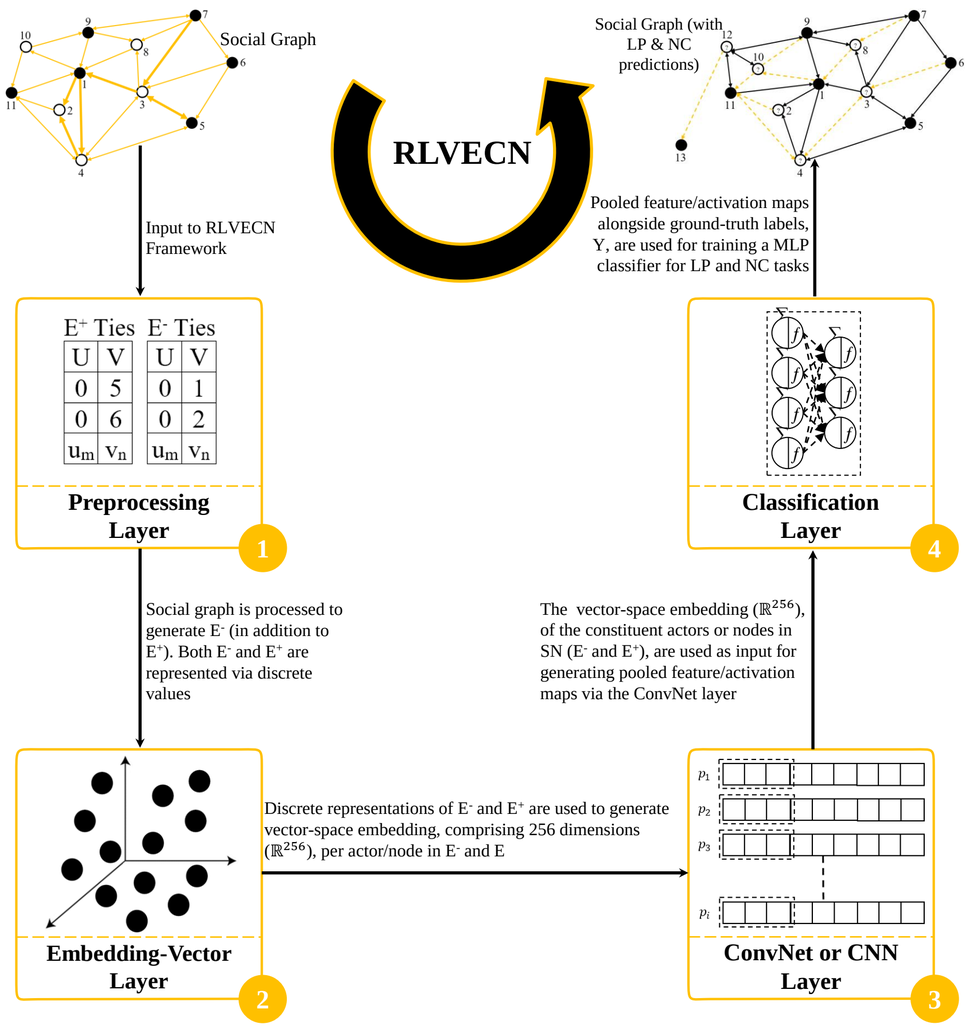

Fig. 1: Proposed System Architecture

Training Algorithms

Algorithm Proposed Procedure for Node Classification

Input: $\{V, E, Y_{lbl}\} \equiv$ {Actors, Ties, Ground-Truth Labels}
Output: $\{Y_{ulb}\} \equiv$ {Predicted Labels}

Initialization:
// V_{lbl} : Labelled actors // V_{ulb} : Unlabelled actors
$V_{lbl}, V_{ulb} \subset V = V_{lbl} \cup V_{ulb}$
$E : (u_i, v_j) \in \{U \times V\}$ // $(u_i, v_j) \equiv$ (source, target)
// $|E_{train}| = \sum indegree(V_{lbl}) + \sum outdegree(V_{lbl})$
$E_{train} = E_t : u_i, v_j \in V_{lbl}$
$E_{pred} = E_p : u_i, v_j \in V_{ulb}$

$f_c \leftarrow$ Initialize // Construct classifier model

Training:
for $t \leftarrow 0$ **to** $|E_{train}|$ **do**
  $f : E_t \rightarrow \{X_t \in \mathbb{R}^q\}$ // Embedding operation
  $f_t \in F = X_t \odot K_t$ // Convolution operation
  $r_t \in R = g(F) = max(0, f_t)$
  $p_t \in P = h(R) = maxPool(r_t)$
  $f_c|\Theta : p_t \rightarrow Y_{lbl}$ // MLP classification operation
end for

return $Y_{ulb} = f_c(E_{pred}, \Theta)$

Algorithm Proposed Procedure for Link Prediction

Input: $\{V, E, \mathbb{B}_{gTruth}\} \equiv$ {Actors, Ties, Ground-Truth Entities}
Output: $\{\mathbb{B}_{pred}\} \equiv$ {Predicted Entities}

Initialization:
$\mathbb{B}_{gTruth} : \{0, 1\} \equiv$ {C0 : -ve/False tie, C1 : +ve/True tie}
$E = E_{+ves} \cup E_{-ves} = (u_i, v_j) \in \{U \times V\} \subset \{V \times V\}$
// E_{train} : Ground-Truth edgelist
// $E_{pred} : E'_{train}$ = Complement of E_{train}
$E_{train} = E_t : E \rightarrow \mathbb{B}_{gTruth}$ // $|E_{train}| = E - E_{pred}$
$E_{pred} = E - E_{train}$

$f_c \leftarrow$ Initialize // Construct prediction model

Training:
while $E_{train} \neq NULL$ **do**
  $f : E_t \rightarrow \{X_t \in \mathbb{R}^q\}$ // Embedding operation
  $f_t \in F = X_t \odot K_t$ // Convolution operation
  $r_t \in R = g(F) = max(0, f_t)$
  $p_t \in P = h(R) = maxPool(r_t)$
  $f_c|\Theta : p_t \rightarrow \mathbb{B}_{gTruth}$ // MLP : $\Theta = similarity()$
end while

return $\mathbb{B}_{pred} = f_c(E_{pred}, \Theta)$

This research work was supported by International Business Machines (IBM). Research was conducted on high performance IBM Power System S822LC Linux Server.