Topo2vec: A novel node embedding generation based on the topology of network for the link prediction *(supplementary file)*

Koushik Mallick, Sanghamitra Bandyopadhyay, *Fellow, IEEE,* Subhasis Chakraborty, Rounaq Choudhuri and Sayan Bose

1 Algorithms

In *Proposed Method (ref: Section 3 of the main manuscript)*, we have described two algorithms that are used in our proposed method, Topo2vec. Here, for detailed references to the used algorithms, we have a more elaborate explanations of the same.

The Algorithm 1 *(ref: Algorithm 1 of the main manuscript)* performs the initial exploration of a sub-graph by mining its neighbours from the adjacency list of the concerned source node. Algorithm 2 *(ref: Algorithm 2 of the main manuscript)* extends the context sub-graph formation as built in Algorithm 1 by employing more extensive but stringent methodology.

Algorithm 1 To explore the initial context subgraph of each individual nodes in Graph G, $G = \{V, E\}$,

The input: Adjlist (Adjacency list data structure of the Graph.) Initialize $context\_nodes \leftarrow NULL$, $node\_number = \|V\|$.
for $CurrNode \leftarrow (0 : node\_number - 1)$ **do**
  $visited[0 : node\_number - 1] \leftarrow 0$
  $visited[CurrNode] = 1$
  $templist = Adjlist[CurrNode]$;
  if templist==NULL **then**
    continue;
  else
    $size \leftarrow length(templist)$,Score[0:size-1]$\leftarrow 0$
    for $i \leftarrow$ (0 :size-1) **do**
      $Score[i] = NA_{CurrNode}^{templist(i)}$
    end for
    $[Score, ind] = Sort(Score)$;
    $templist \leftarrow templist[ind]$;
    $templist(i) = [\exists i, Score[i] \geq \tau]$;
  end if
  for $k \leftarrow 0 : Average\_degree$ **do**
    $templist =$ SEARCH$(templist, visited, CurrNode)$
  end for
  context_nodes=[context_nodes; templist]
end for=0

Algorithm 2 To expand the context sub-graph of a source node in a network.

procedure SEARCH($templist$, $visited$, $CurrNode$)
$Tempauglist \leftarrow NULL$
$Utemplist \leftarrow Uniquelist(templist)$
for $i \leftarrow (0 : length(Utemplist) - 1)$ **do**
  if ($visited[Utemplist[i]]$==0) **then**
    $visited[Utemplist[i]] \leftarrow 1$
    $neighborl = Adjlist$[Utemplist[i]]
    for $j \leftarrow (0 : |neighborl| - 1)$ **do**
      $Score[i] \leftarrow NA_{CurrNode}^{neighborl(j)}$
    end for
    $[Score, indx] = Sort(Score)$
    $neighborl = neighborl[indx]$
    $neighborl(i) = [\exists i, Score[i] \geq \tau]$
    $Auglist \leftarrow neighborl$
    for $k \leftarrow (0 : |(Auglist)| - 1)$ **do**
      $Score[k] \leftarrow SA_{CurrNode}^{Auglist(k)}$
    end for
    $[Score, indx] = Sort(Score)$
    $Auglist = Auglist[indx]$
    $Auglist(k) = [\exists k, Score[k] \geq \tau]$
    $Tempauglist \leftarrow [Tempauglist, Auglist]$
  else
    Continue;
  end if
end for
$templist \leftarrow [templist, Tempauglist]$
return $templist$
end procedure=0

2 Analytical Analysis

The proposed algorithm Topo2vec for generation of node embeddings of a graph, has been evaluated using various operators as mentioned in *Evaluation Methods (ref: Section 5 of the main manuscript)*. In *Discussion (ref: Section 7 of the original manuscript)*, we have elaborated on the testing of the new feature projection method using different operators, *viz.*, *Hadamard Product*, *SimKron_Hadamard* and *SimKron_Avgerage* as described in *(Table 1 of the original manuscript)*. Here, we have shared the analytical analysis in

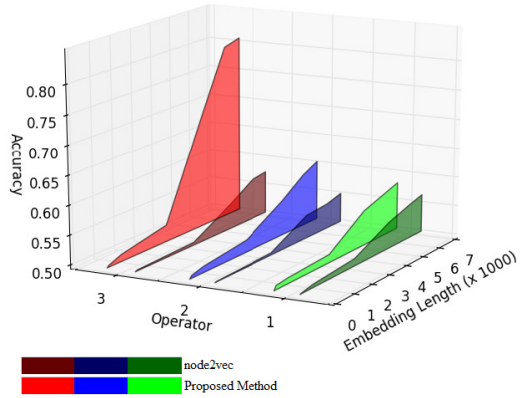

Fig. 1: Performance comparison of different pairwise feature representation operators defined in Table 1 in the manuscript, through clustering. Results are shown using PPI data (4 fold linked information) with the proposed method and node2vec. Index in the operator-axis represents (1) Hadamard, (2) *SimKron_Average* and (3) *SimKron_Hadamard* operators respectively.

the form of a 3-Dimensional plot which helps us visualize the difference in performance among these operators.