T-PAIR: Temporal Node-pair Embedding for Automatic Biomedical Hypothesis Generation

Uchenna Akujuobi, *Member, IEEE,* Michael Spranger, Sucheendra K. Palaniappan,
and Xiangliang Zhang✉, *Senior Member, IEEE*

Abstract—In this paper, we study an automatic hypothesis generation (HG) problem, which refers to the discovery of meaningful implicit connections between scientific terms, including but not limited to diseases, chemicals, drugs, and genes extracted from databases of biomedical publications. Most prior studies of this problem focused on the use of static information of terms and largely ignored the temporal dynamics of scientific term relations. Even when the dynamics were considered in a few recent studies, they learned the representations for the scientific terms, rather than focusing on the term-pair relations. Since the HG problem is to predict term-pair connections, it is not enough to know with whom the terms are connected, it is more important to know how the connections have been formed (in a dynamic process). We formulate this HG problem as a future connectivity prediction in a dynamic attributed graph. The key is to capture the temporal evolution of node-pair (term-pair) relations. We propose an inductive edge (node-pair) embedding method named T-PAIR, utilizing both the graphical structure and node attribute to encode the temporal node-pair relationship. We demonstrate the efficiency of the proposed model on three real-world datasets, which are three graphs constructed from Pubmed papers published until 2019 in *Neurology*, *Immunotherapy*, and *Virology*, respectively. Evaluations were conducted on predicting future term-pair relations between millions of seen terms (in the transductive setting), as well as on the relations involving unseen terms (in the inductive setting). Experiment results and case study analyses show the effectiveness of the proposed model.

Index Terms—Hypothesis generation, Dynamic edge embedding, Graph neural networks, Biomedical term relation prediction

1 Introduction

In the sciences, we are now uniquely privileged to sit side by side with the giants on whose shoulders we stand – Gerald Holton. The road to scientific discoveries has traditionally involved scientific expertise and ideas coalescing to form hypotheses, which are then checked for validity (e.g., as shown in [1]). The foundation of both expertise and ideas is rooted in literature, as literature provides the background for new knowledge and information. Thus, new hypotheses with minimum uncertainty about undiscovered knowledge can be made from already published scholarly literature [2], [3]. In this paper, we focus on new hypotheses about whether two scientific terms/concepts are relevant to each other, given that there is no direct link between them at the current knowledge scope. For instance, in 1998 [4], *Schizophrenia* and *Calcium-Independent Phospholipase A2* independently studied in [5] (in 1997) and in [6] (in 1995) were connected because they had *oxidative stress* as a common factor. Recently, the number of scientific publications is growing at an exponential rate, with over a million articles published yearly [7], [8]. Reading this great number of publications is far beyond the capability of individual researchers. It thereby makes the process of reading to extract undiscovered knowledge a tedious and time-consuming task.

Towards advanced scientific knowledge discovery, computers have been introduced to play an ever-greater role in the scientific process with automatic hypothesis generation (HG) based on machine learning. The study of automated HG has attracted considerable attention in recent years [9], [10], [11], [12], [13], [14]. The existing HG methods form three main groups. The first group [9], [10], [15] are based on the ABC model for discovery. The ABC model postulates that if entities $\{a, b\}$ are related, and $\{b, c\}$ are related, there should exist a relationship between $\{a, c\}$. For instance, if diseases X and Y are highly similar in characteristics and causation, and drug Z is a known cure of disease X. Then, drug Z has a high chance of being a remedy for disease Z.

The second group covers methods that use a combination of advanced machine learning strategies to extract and analyze hidden connections from scientific publications. These methods include but not limited to association rules [10], [12], [16], text mining [13], [17], clustering and topic modeling [2], [18], [19], and others [14], [20], [21]. However, most of the previous studies fail to capture and utilize the dynamic evolution of the entity meaning, which can provide crucial information on inferring the future connectivity of the entities (e.g., medical terms).

To address this limitation, the final group of studies [20], [21] focuses on incorporating the temporal information of the terms in the hypothesis generation process. Specifically, they model the dynamic meaning of the terms and then infer the term-term relationship by the combination of the term semantic features. Therefore, the inference performance highly depends on the quality of the learned term features. More importantly, it is not enough to know with whom the terms are connected (focusing on learning the term

- *U. Akujuobi and X. Zhang are with Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology (KAUST), Saudi Arabia.*
  E-mail: {Uchenna.Akujuobi, Xiangliang.Zhang}@kaust.edu.sa
  Corresponding author: Xiangliang Zhang
- *M. Spranger is with Sony Computer Science Laboratories Inc.*
  E-mail: michael.spranger@gmail.com
- *S. Palaniappan is with Systems Biology Institute, Japan.*
  E-mail: Sucheendra@sbi.jp

representation), it is essential to know *how the connections have been formed (in a dynamic process)*. For example, *Fish Oils* (FO) was discovered to help treat *Raynaud's Disease* (RD) in 1985. In our study in Section 4.4.1, the predicted relevance score between FO and RD in the evaluation period (*1980 - 1989*) corresponding the true discovery year (1985) was much lower by modeling the static term relations (0.12) than by modeling the temporal relations (0.61), due to the implicit differentiation on the recent and former relations.

Therefore, we target on learning the *temporal dynamics of scientific term relations*, rather than just learning features for the terms themselves. Furthermore, previous methods have focused on predicting future connections of known terms; thus, they are *transductive*. Hence, they are not able or too computationally expensive to deal with the situations where new terms (e.g., new diseases, drugs) are observed continuously. In practice, new terms continuously appear in the scientific publication flooding.

Our study is to design a new *inductive* model, but also working for the transductive setting, utilizing both the graphical structure and node features to *encode the temporal node-pair relationship*. To model the historical evolution of term-pair relations, we construct a term relationship graph $G = \{V, E\}$. We then decompose this graph into temporal graphlet with different node features per graphlet, denoting the temporal nature of the node attributes.

Definition 1. **Temporal graphlet**: *A temporal graphlet* $G^t = \{V^t, E^t, x_v^t\}$ *is a temporal subgraph at time step t, which consists of nodes* $V^t \subset V$ *and* $E^t \subset E$*. The variable* x^t *is the node attribute of* V^t.

We construct a temporal sequence[1] of graphlets, where the t-th graphlet $G^t = \{V^t, E^t, x_v^t\}$ is the part of the graph $G = \{V, E\}$ observed at an incremental time t. We then formulate our HG problem as a temporal node-pair connectivity prediction task, defined as:

Definition 2. **Temporal node-pair connectivity prediction task**: *At each time step t, we are given an attributed graph* G^t*, and a set of node pairs A, each of which pair is composed of two nodes in* V*. At time t, part of the node pairs in A have labels* y_l^t*, denoted as* A_l^t*, for which we know if one node-pair is connected at time* $t+1$ *(positive, in* y_l^t*), or not connected at time* $t+1$ *(negative, in* y_l^t*). The remaining pairs* $A_u^t = A^t / A_l^t$ *are unlabeled. The goal is to infer the change in labels of the pairs in* A^T *at the last time step T by using the temporal graphlets until* $T-1$, $G = \{G^1, G^2, ..., G^{T\ 1}\}$*, and the corresponding node pairs labels,* $y_l = \{y_l^1, y_l^2, ..., y_l^{T\ 1}\}$.

About the above-defined problem, we would like to clarify the following points:

1) The positive pairs in A_l^t is a subset[2] of the node pairs linked by edges in E^{t+1}. Labeled pairs in A^t can be considered as supervising pairs obtained from the future step $t+1$, and is used to guide the node-pair relation learning in G^t. Formulating the problem in this way is novel and enables to model *how the pairwise relations have been formed in a dynamic process*. We present in section 3.1 with more details about the construction of graph G, and the label set y over t=1, ..., T.
2) Positive pairs in A_l^t stays as positive in A_l^{t+1} (i.e., $A_{l+}^t \to A_{l+}^{t+1}$). A negative pair in A_l^t can become positive at $t+1$, or stays negative (i.e., $A_l^t\ \ \to A_l^{t+1}$). An unlabeled pair can become positive, or negative at $t+1$ (i.e., $A_u^t \to A_l^{t+1}$). Therefore, at the final time step T, we infer labels for pairs in A_u^T and changes in A_l^T .

To address this defined problem, we target on building an automatic HG model that has the following advanced unique features:

The model can capture how the relations of term-term (node-node) pairs evolve over time and take advantage of this relation dynamics to infer new future connections (hypotheses). In other words, the model is built on a set of temporal graphlets $G = \{G^1, G^2, ..., G^T\}$, and the labeled set of node pairs, A_l^t; $1 \le t \le T-1$, unlike most existing solutions that only use G^T [11], [14].

The model should work in both transductive and inductive setting. The transductive inference is to predict future connections between known terms (node pairs in A_u^T and A_l^T where nodes already exist in $V^{T\ 1}$). The inductive inference is to predict future connections with previously unknown terms (nodes out of $V^{T\ 1}$).

We propose a model named **T-PAIR**, for addressing the **temporal node-pair connectivity prediction** problem. To capture the time-evolving relation between terms (nodes), T-PAIR incrementally learns the node-pair embedding at each time step by using a recurrent neural network structure. The node-pair connectivity prediction (i.e., HG problem) can be addressed by classifying the learned node-pair embeddings. T-PAIR takes comprehensive usage of the set of temporal graphlets $G = \{G^1, G^2, ..., G^T\}$ including all its node attributes, with the sequential supervising information in $A_l = \{A_l^1, A_l^2, ..., A_l^{T\ 1}\}$, for inferring the connectivity of pairs in A_u^T and A_l^T .

We evaluated T-PAIR on three real-world datasets of scholarly publications in the PubMed database until 2019 (with detailed graph information and formulation process in section 3.1.1 and 4.1). The model is trained end-to-end and shows superior performance on both transductive and inductive hypotheses generation tasks. Case studies demonstrate that T-PAIR predicts the positive relationship between medical terms before they were actually discovered.

The remaining parts of this paper include relate work discussion in Section 2, the introduction of T-PAIR in Section 3, the experimental evaluation in Section 4 and conclusion and future work in Section 5.

2 Related Works

2.1 Hypothesis Generation

Hypothesis generation aims to extract implicit information in scholarly documents to infer future connections between scholarly terms. In recent years, researchers have proposed

[1] We focus on insertion-only temporal graphlets since the scientific concepts (terms/nodes) never disappear over timeline, but continuously appear with new connections made. See Figure 1.

[2] It is too expensive to include all $|G^{t+1}| * |G^{t+1}|$ pairs.

the use of machine learning for hypothesis generation. *ARROWSMITH* is a system proposed based the ABC model [9]. To curtail the limitation of the ABC model whose computational complexity significantly increases w.r.t. the number of common entities, other methods [10], [12], [16] proposed the use of semantic relations to augment ABC processing. Weeber et al. proposed the DAD-system [22], which applies Natural Language Processing model as a guide in generating a new hypothesis. Lindsay and Gordon [23] used lexical statistics, such as word frequency counts, to discover hidden connections in the medical literature. Srinivasan et al. [24], [25] used text mining techniques based on topic profiles, where a topic profile is a set of terms (single words and phrases) extracted from the documents relevant to the given topic.

More recently, Spangler et al. [13], [17] proposed to use text mining to identify entity relationships from medical texts. Shi et al. [14] used logistic regression to model the probability that two entities will be connected based on a given time window. They applied random walk to capture the structural information of the scholarly graph. Some other methods [2], [18], [19] proposed the incorporation of machine learning techniques such as Latent Dirichlet Allocation (LDA), clustering, and topical phrase mining. The previously mentioned methods all focused mostly on static scholarly graphs. However, as noted in [14], scholarly graphs exhibit a temporal structure. To capture the temporal information in such graphs, Jha et al. [20] proposed the use of a temporal matrix factorization framework to model the co-evolution of terms across knowledge-bases. Xun et al. [21] modeled the evolving schematics of term embeddings based on their indexing (MeSH) terms.

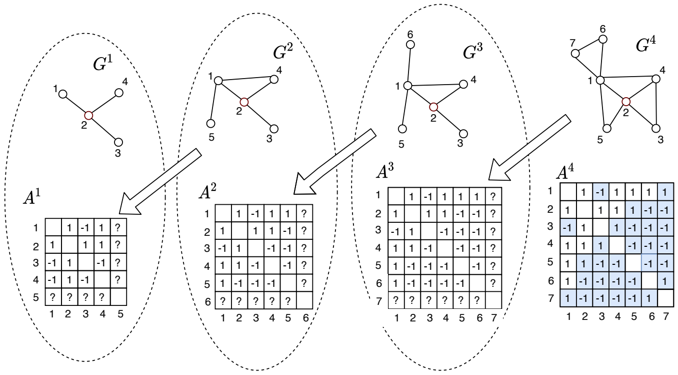
We formulate the HG problem in a novel way, as introduced in Definition 2. We focus on modeling the dynamic relationship between terms, presented in set of temporal graphlets $G = \{G^1, G^2, ..., G^T\}$ with the sequential supervision in $A_l = \{A_l^1, A_l^2, ..., A_l^T\}$. Also, our model can be used in both transductive and inductive settings. Hence, we can model the relationship between newly introduced terms and other existing terms in the networks.

2.2 Network Embedding

Network embedding has gained considerable momentum in recent years. We concentrate our discussion on two groups of related work, edge embedding, and dynamic network embedding.

We focus on the study of node-pair relation, specifically representing node-pair relationships as low dimensional vectors. This problem can be considered as edge embedding, which has a simple solution: joining the embedding of two nodes linked by one edge [26], [27]. Other sophisticate models embed node-pairs and their context in the same space [28], or learn node and edge embedding vectors jointly [29]. Several knowledge graph embedding methods [30], [31], [32], [33], [34] proposed to learn embedding for both (head and tail) node and (relation) edge jointly. These above-discussed methods all focused on static graphs and hence, and cannot model the temporal dynamics of node-pair relationships.

In the study of temporal/dynamic networks, Goyal et al. [35] used multiple non-linear layers to learn structural patterns of networks at different time steps and then learn the temporal transition using RNN. Singer et al. [36] proposed the use of orthogonal transformation to align node embeddings at time t_k and time t_o; $o = k + \epsilon$, which is then passed to RNN to capture the final node representation. Zhou et al. [37] modeled how a triad (a group of three vertices) develops from an open triad (no connection between two of the vertices) to a closed triad (all vertices are connected).



	1	2	3	4	5
1		1	-1	1	?
2	1		1	1	?
3	-1	1		-1	?
4	-1	1	-1		?
5	?	?	?	?	

	1	2	3	4	5	6
1		1	-1	1	1	?
2	1		1	1	-1	?
3	-1	1		-1	-1	?
4	1	1	-1		-1	?
5	1	-1	-1	-1		?
6	?	?	?	?	?	

	1	2	3	4	5	6	7
1		1	-1	1	1	1	?
2	1		1	1	-1	-1	?
3	-1	1		-1	-1	-1	?
4	1	1	-1		-1	-1	?
5	1	-1	-1	-1		-1	?
6	1	-1	-1	-1	-1		?
7	?	?	?	?	?	?	

	1	2	3	4	5	6	7
1		1	-1	1	1	1	1
2	1		1	1	1	-1	-1
3	-1	1		1	-1	-1	-1
4	1	1	1		-1	-1	-1
5	1	1	-1	-1		-1	-1
6	1	-1	-1	-1	-1		1
7	1	-1	-1	-1	-1	1	

Fig. 1: An illustration of the set of temporal graphlets $G = \{G^1, G^2, ..., G^T\}$ with the sequential supervision in $A = \{A^1, A^2, ..., A^T\}$ (T=4 here). As the graph grows with more nodes and more connections, A^t is constructed from the graph G^{t+1}. One unit at i, j of A^t is labeled by $y_t^{i,j} = 1$ (a positive relation) if node i and j are connected in G^{t+1}. The unit is labeled by $y_t^{i,j} = -1$ (a negative relation) if node i and j are NOT connected in G^{t+1}. During training, since we do not observe the future, the node pairs in A^T are equivalent to those in $A^{T\ \ 1}$. The difference between them is the changes in the colored units are only observed in testing. In the training process, G^t and A^t ($t = 1...T$) form training samples of node pairs with positive or negative labels. When testing, we aim to predict the unobserved pair relationships (colored tiles), in A^T.

However, none of these methods directly model the temporal dynamics of node-pair relationships as they aim to capture the dynamic semantics of the nodes. Rahman et al. [38] considered the node-pair features as an optimal coding problem and used temporary functions to learn the temporal patterns. However, the proposed method is non-scalable for graphs with high density. Our study, in contrast to previous methods, focuses on directly capturing the historical evolution of node-pairs (edges) in an attributed network, while preserving an inductive property. In addition, our proposed model comprehensively integrates the different sources of information in the network.

3 Methodology: T-PAIR

Since our problem is formulated on a temporal attributed graph, in this section, we first describe the graph construction. We then present our proposed T-PAIR model.

3.1 Data Preparation

3.1.1 Graph Construction

Given a dataset of scholarly publications (e.g., from PubMed), we extract and categorize the terms in the documents based on predicates extracted from MEDLINE defined on a set of UMLS terms [39]. Each term belongs to one of three categories, namely: 1) *Genes*, 2) *Chemicals*, and 3) *Diseases*. We then construct a network $G = \{V, E\}$, where V is the set of nodes corresponding to the biomedical terms. The relationship E represents the mention of two terms in the same literature. To be specific, an edge in E connects two nodes if the two corresponding terms are mentioned together in the same paper[3]. Although a co-occurrence of two terms in a paper can have a negative connotation, it is interesting to find them out. We leave the study of the type of relevancy between the co-occurrence connections for future work.

Next, we split the obtained network using a year window, thereby, obtaining a sequence of temporal graphlets $G = \{G^1, G^2, ..., G^T\}$. As defined in the Introduction Section 1, this graphlet sequence encapsulate the temporal evolution of node pair relationships. Since the node terms belong to three different categories, the graph $G^t = \{V^t, E^t, x^t\}$ is, in fact, a dynamic heterogeneous attributed graph, with incremental set of nodes V^t, where $V^1 \subseteq V^2 ... \subseteq V^t ... \subseteq V^T$ and edges $E^1 \subseteq E^2 ... \subseteq E^t ... \subseteq E^T$. The node attribute x^t is composed of the term (e.g., fish oil, lung cancer), and the term contexts, which are the aggregation of sentences encompassing the mention of the terms in the documents. We use the texts from the publication titles and abstracts. The node attributes vary per time period due to the increase in the number of publications.

3.1.2 Supervised Pair Construction

For each time step t, we construct pairs of nodes. The node pairs are then labeled based on the graph of the next observed time step. The constructed node pairs are of the format $a^{i,j} =< v_i, v_j >$, consisting of nodes v_i and v_j. As shown in Figure 1, the pairs per time step t are labeled based on the observation made on the graph $G^{t+1} = \{V^{t+1}, E^{t+1}\}$ of the next time window $t+1$. A node pair $a^{i,j}$ is assigned a positive class $+1$ if the connection is observed in graph G^{t+1} (i.e., $y^t_{a^{i,j}} = +1 \iff e(v_i, v_j) \in E^{t+1}$) or -1 otherwise.

Since we consider insertion only graphlets sequence, the graph size of the graphlets grows proportionally with the increase in time step. Therefore, the use of all possible pairs for training becomes more computationally expensive and less feasible in application. In this study for large graphs, the notion of node pair set is defined as a sampled subset of all possible node pairs. This sample is drawn uniformly for each time step t.

3.2 T-PAIR Overview

The architecture of our T-PAIR model is shown in Figure 2. As specified in Problem 1.1, T-PAIR is designed to learn from the set of temporal graphlets $G = \{G^1, G^2, ..., G^T\}$, and the labeled set $A_l = \{A^1_l, A^2_l, ..., A^T_l\}$. Once trained end-to-end, T-PAIR can be used to infer the labels of pairs in A^T_u.

One pair of nodes $a^{i,j}$ in A^t_l is used in the training process of T-PAIR by evaluating its connectivity prediction score $p^t_{a^{i,j}}$. The main steps of calculating the connectivity prediction score are given in Algorithm 1. The testing process also uses the same Algorithm 1 (with t=T), calculating $p^T_{a^{i,j}}$ for one pair of nodes in A^T_u. We next explain these steps with the help of Figure 2.

The connectivity prediction score is calculated in line 6 by $p^t_{a^{i,j}} = f_C(h^t_{a^{i,j}}; \theta_C)$, where θ_C is the classification network parameter, and the embedding vector $h^t_{a^{i,j}}$ for the pair $a^{i,j}$ is iteratively updated in lines 1-5. These iterations of updating $h^t_{a^{i,j}}$ are shown as the recurrent structure shown in Figure 2 (a), followed by the classifier $f_C(.; \theta_C)$.

The recurrent update function $h^\tau_{a^{i,j}} = f_A(h^{\tau\ 1}_{a^{i,j}}, z^\tau_{v_i}, z^\tau_{v_j}; \theta_A)$ in line 4 is shown in Figure 2 (b). It takes input of the embedding vector in previous step $h^{\tau\ 1}_{a^{i,j}}$, the node feature x^τ_v, and the neighboring node feature $x^\tau_{Nr(v)}$. The latter two are processed by an aggregation block shown in Figure 2(c), producing $z^\tau_v = f_G(x^\tau_v, x^\tau_{Nr(v)}; \theta_G)$, given in line 3. The aggregation network $f_G(; \theta_G)$ is implemented following GraphSAGE, which is one of the most popular graph neural networks for aggregating node and its neighbors [40]. We next introduce each of the blocks in details.

3.3 Neighborhood Aggregation, $f_G(.; \theta_G)$

The aggregator network $f_G(.; \theta_G)$, which has GraphSAGE at its core, aggregates the information of each node in a given node pair $a^{i,j} =< v_i, v_j >$ to obtain a terse representation for them. For each node v in the pair, the aggregation network takes as input the current node feature x^t_v as well as the neighborhood information, which includes the node features of the sampled node neighbors $x^t_{Nr(v)}$. At the initial step of aggregation $m = 0$, node vector $\beta^0_v = x^t_v$. Then, the representation vectors of neighbors $\{\beta^{m\ 1}_u, \forall u \in Nr(v)\}$ at iteration $m - 1$ are aggregated into a single vector β^m_v at iteration m, by considering the following aggregation options[4]:

LSTM Aggregator. This aggregator is based on the LSTM architecture. LSTM is adapted to aggregate information from a random permutation of the node's neighbors.

Pool Aggregator Each neighbor representation is passed through a fully connected layer. An element-wise pooling technique is then used to aggregate the information from its neighboring set

$$\beta^m_v \leftarrow \lambda(\{\sigma(W \cdot \beta^m_{u_i} + b) \forall u_i \in Nr(v)\}), \tag{1}$$

where λ denotes a symmetric vector operator (e.g., max, mean, and others), and σ is a nonlinear activation function.

Mean Aggregator The mean aggregator aggregates the neighborhood representation by taking the element-wise mean of the neighbor representations:

$$\beta^m_v \leftarrow \sigma(W \cdot MEAN(\{\beta^{m\ 1}_u, \forall u \in Nr(v)\})), \tag{2}$$

Inductive GCN Aggregator This is a modified mean based aggregation:

$$\beta^m_v \leftarrow \sigma(W \cdot MEAN(\{\beta^{m\ 1}_v\} \cup \{\beta^{m\ 1}_u, \forall u \in Nr(v)\})), \tag{3}$$

where σ is a nonlinear activation function. This method can be viewed as an inductive variant to the GCN approach [41].

[3] A mention can have a positive or negative connotation. We consider any kind of mention as a relationship, regardless of positive or negative.

[4] Different techniques exist for the neighborhood aggregation, as discussed in [40].

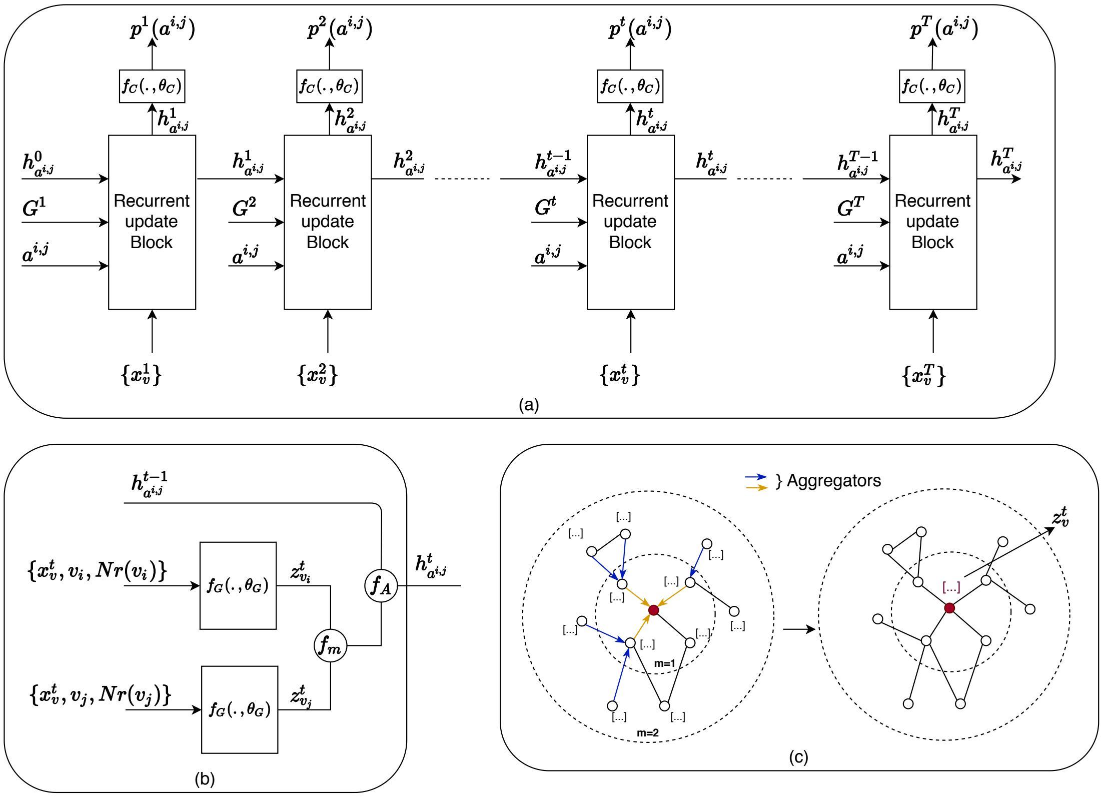


Fig. 2: The proposed T-PAIR model. Block (a) shows the outer view of the model framework. The inner structure of the recurrent update block and aggregator network are shown in block (b) and (c) respectively.

Algorithm 1: Calculate the future connection score for term pairs $a^{i,j} =< v_i, v_j >$

```
Input: $G = \{G^1, G^2, \ldots, G^T\}$ with node feature $x^t_v$,
    a node pair $a^{i,j} =< v_i, v_j >$ in $A^t_l$, and an
    initialized pair embedding vector $h^0_{a^{i,j}}$ (e.g.,
    by zeros) for the target node pair
Result: $p^t_{a^{i,j}}$, the connectivity prediction score for
    the node pair $a^{i,j}$
for $\tau \leftarrow 1 \cdots t$ do
    Obtain the current node feature $x^\tau_v$ ($v = v_i, v_j$) of
      both nodes (terms) $v_i, v_j$; as well as $x^\tau_{Nr(v)}$
      ($v = v_i, v_j$) for the node feature of sampled
      neighboring nodes for $v_i, v_j$;
    Aggregate the neighborhood information of node
      $v = v_i, v_j$, $z^\tau_v = f_G(x^\tau_v, x^\tau_{Nr(v)}; \theta_G)$;
    Update the embedding vector for the node pair
      $h^\tau_a = f_A(h^{\tau\ 1}_a, z^\tau_{v_i}, z^\tau_{v_j}; \theta_A)$ ;
end
Return $p^t_{a^{i,j}} = f_C(h^t_{a^{i,j}}; \theta_C)$
```

After the neighborhood aggregation by several layers (e.g., M layers), the final representation $z^t_v = \beta^M_v$. The performance of aggregators often depends on the property of the applied graph [40]. We evaluate different aggregators and report the best, which is maxpool in our application problem.

Neighborhood Definition. Following the principle of [40], to keep the computational footprint to a minimum, we work on a fix-size sample set of node neighbors instead of the full neighborhood nodes. Hence the notion of node neighbors $Nr(v)$ is defined as a fix-size sample of the full node neighborhood $\{u \in V : (u, v) \in E\}$. This sample is drawn uniformly at each iteration, thereby reducing the time and memory complexity. With the sampling strategy, the memory and time complexity per node aggregation step is fixed at $\mathcal{O}(\prod_{m=1}^{M} S_m)$, where S_m is the neighborhood sample size at layer m, and M is the maximum layer considered (i.e., up to M-hop neighbors).

3.4 Pair Embedding Generation, $f_A(.;\theta_A)$

For a given pair $a^{i,j}$ in the recurrent update block, after the neighborhood aggregation process at time step t, we obtain the final neighborhood representation of each node $\{z^t_{v_i}, z^t_{v_j}\}$. The previous pair vector representation $h^{t\ 1}_{a^{i,j}}$ is updated by the recurrent network $f_A(.;\theta_A)$ to obtain $h^t_{a^{i,j}}$ by:

$$h^t_{a^{i,j}} \leftarrow \sigma(R \cdot h^{t\ 1}_{a^{i,j}} + U \cdot (z^t_{v_i} + z^t_{v_j})), \qquad (4)$$

where σ is a nonlinear activation function, and the variables $\{R, U\}$ are the weights. The variable $h^t_{a^{i,j}}$ is the pair vector representation at time step t containing information on the temporal relationship between nodes v_i and v_j, $h^{t\ 1}_{a^{i,j}}$ is the output of the previous time step $t - 1$ (i.e., the historical information), and $\{z^t_{v_i}, z^t_{v_j}\}$ are the aggregated node neighborhood representation at time t for nodes v_i and v_j, respectively.

3.5 Parameter Learning

In order to train three networks $f_A(.;\theta_A)$, $f_G(.;\theta_G)$ and $f_C(.;\theta_C)$ in T-PAIR, we use all the labeled training data in $A_l = \{A_l^1, A_l^2, ..., A_l^T\}$. We have discussed for one pair $a^{i,j}$ in A_l^t, how to use Algorithm 1 to update its embedding vector $h^t_{a^{i,j}}$ and its connectivity prediction score $p^t_{a^{i,j}}$. Suppose the label of $a^{i,j} =< v_i, v_j >$ is $y^t_{a^{i,j}}$, then $p^t_{a^{i,j}}$ is expected to be close to $y^t_{a^{i,j}}$. From the current t to the end T, the embedding vector $h^t_{a^{i,j}}$ should evolve to produce $p^t_{a^{i,j}}$ similar to $y^t_{a^{i,j}}$. Therefore, we define our loss function as

$$-\sum_{t=1}^{T}\sum_{k=1}^{K} y^t_{a^{i,j}} \cdot \log(p^t_{a^{i,j}}) + (1 - y^t_{a^{i,j}}) \cdot \log(1 - p^t_{a^{i,j}}), \quad (5)$$

where $K = |A_l^t|$ is the cardinality of the labeled node pairs, $y^t_{a^{i,j}}$ is the true label of pair $a^{i,j} =< v_i, v_j >$. In our setting, $y^t_{a^{i,j}}$=1 if v_i and v_j are not connected at the current step t but connected at the next time step $t+1$. Otherwise $y^t_{a^{i,j}} = 0$. Note that once one pair is labeled as positive, in all subsequent time steps, this pair always has a positive label: $y^\tau_{a^{i,j}} = 1; t \leq \tau \leq T$.

In implementation, during training, $A^{T \ 1}$ and A^T are equivalent, since $A^{T \ 1}$ is constructed by using G^T, and there is no G^{T+1} to construct A^T. This setup enhances the model stability in the training process, forcing/guaranteeing that the positive pairs at $T-1$ remain positive at T. Note that in Figure 2, $A^{T \ 1}$ and A^T differ on the colored tiles that are not observed in training, and to make label inference in testing.

4 Experimental Evaluation

We first present the datasets for training and testing, the experimental setting, and then the quantitative evaluation results with comparison to a number of baseline methods, and also the parameter sensitivity analysis. Last, we present the qualitative study results on several real-world cases predicted by T-PAIR.

4.1 Dataset

In this project, we study the hypothesis generation problem on the PubMed data dump (end early 2019). This dataset contains the title and abstract of $\sim$ 22 million papers published from 1944 - 2019. Among the 200,000 keywords in the dataset, we select those belonging to: *Genes*, *Chemicals*, or *Diseases*. The selected keywords are the medical terms that we are interested in, and will be treated as nodes for graph construction.

To evaluate the model's adaptivity in different scientific domains, we construct three graphs from papers in *Neurology*, *Immunotherapy*, and *Virology*, following the instruction in section 3.1.1. The graph statistics are shown in Table 1. To set up the training and testing data, we split the graph by a 10-year interval starting from 1949 (i.e., $\{\leq 1949\}, \{1950-1959\}, \ldots, \{2010-2019\}$). We use year splits of ≤ 2009 ($\{G^1, G^2, ..., G^7\}$) for training, and the final split $2010-2019$ for testing.

For each split G^t, we create a set of labeled term pairs A_l^t. As introduced in section 3.1.1, positive pairs A^t_{l+} are those connected node pairs in graph G^{t+1} in the next year window $t+1$. The negative node pairs A_l^t are the samples of node pairs not connected in G^{t+1}, and the unlabeled node pairs are then the node pairs with nodes in G^{t+1} not observed in G^t.

For each node $v \in V^t$ in the training set, we generate 20 negative node pairs A_l^t , by pairing each node with the 20 randomly sampled non-node-neighbors in the next year split (i.e., such that $a_l^{i,j} \notin E^{t+1}$). For the testing set, we generate 40 negative node pairs to simulate the real-world scenario where the number of negative pairs is larger than that of positive pairs. The resulting data statistics for each dataset is given in Table 1.

At each t, for one node (a biomedical term), we extract its context, which is a concatenation of N sentences where the term appeared. The term and term contexts are each converted to a 300-dimensional feature vector by applying the latent semantic analysis (LSI) method on the document-term matrix features. The missing term and context attributes are completed with zero vectors. Then, a node has a feature vector defined as $x_v^t = \{x^t_{o_v}; x^t_{c_v}\}; \forall v \in V^t$, which is a concatenation of the term $x^t_{o_v}$ and term context $x^t_{c_v}$ feature vector. At each time t, the term and context features are updated with the new increase about them in discoveries, and publications.

In the inductive evaluation, we need to build feature vectors for new nodes. For the new node v at time t, its feature vector $x^t_{o_v}$ and its context feature vector $x^t_{c_v}$ can be defined as

$$\begin{aligned} x^t_{o_v} &= MEAN(x^t_{o_u} \forall u \in Samp(v)) \quad (6) \\ x^t_{c_v} &= MEAN(x^t_{c_u} \forall u \in Samp(v)), \end{aligned}$$

where $Samp(v)$ is the set of randomly sampled one-hop neighbors of the new node v at time t.

4.2 Experimental Setup

In all our experiments, we treat the graph to be undirected and set the hidden dimensions to $d = 128$. For each neural network based model, we performed a grid search over the learning rate $lr = \{1e^{\ 2}, 5e^{\ 3}, 1e^{\ 3}, 5e^{\ 4}, 1e^{\ 4}\}$, on the virology dataset from 1944 to 1969. The best parameters per model from the grid search are then used in all experiments. The T-PAIR models are trained for 3 epochs with a parameter set ($d = 128, lr = 5e^{\ 4}$, and $S = 20$), where S is the neighborhood sample size. The GraphSAGE, Dyntriad, DynAE, and DynAERNN models are trained for 20 epochs with $d = 28$ and $lr = \{1e^{\ 2}, 1e^{\ 2}, 1e^{\ 2}, 1e^{\ 2}\}$ respectively. The SGCN and nSNE models are trained for 100 and 200 epochs respectively with a parameter set of ($d = 128, lr = 1e^{\ 2}$). The tNodeEmbed models are trained for 5 epochs with a parameter set of ($d = 128, lr = 1e^{\ 2}$). The experiments were conducted on a Linux system using Python. We used the Scikit-Learn implementation of Logistic Regression with default settings for evaluating the embedding based methods. We implement T-PAIR using the Tensorflow library. Each GPU based experiment was conducted on an Nvidia 1080TI GPU.

TABLE 1: Three graph dataset statistics, with their number of nodes and edges. In testing, we use the term pairs existing in the split of 2010-2019. The pairs with *only* terms that have already been seen before 2010 are used in transductive setting. The pairs including at least one unseen term are used in inductive setting. The pairs including at least one already seen term are used in the comprehensive setting. The positive pairs are those with confirmed true connections, while the negative pairs are those without confirmed true connections.

	#nodes	#edges	#Test Pairs (Positive)			#Test Pairs (Negative)		
			Comprehensive	Transductive	Inductive	Comprehensive	Transductive	Inductive
Neurology	78,594	5,321,668	1,707,748	1,457,845	266,075	2,912,568	1,729,694	1,384,992
Immunotherapy	28,823	919,004	303,516	240,766	67,675	1,075,659	671,529	464,833
Virology	38,956	1,117,118	446,574	342,607	115,179	1,382,856	711,264	829,758

4.3 Quantitative Study

4.3.1 Comparison Methods

To evaluate the performance of our model, we compare T-PAIR with several state-of-the-art graph-based methods. We selected the most competitive baselines that are also publicly available online to avoid unfair evaluations due to potential faulty implementation. For plain network embedding baselines, we concatenate the output embeddings with the text (term and context) attributes to obtain the final node representation. The final embeddings of the nodes in each pair are then concatenated and fed through a logistic regression layer (a single layer perceptron) to obtain the probability of the two nodes getting connected in the next time window. These baseline methods include:

GraphSAGE. Since our method is based on GraphSAGE [40], we evaluate the different aggregators in unsupervised GraphSAGE for comparison.

Features. We use just the original term and context attributes obtained from LSI.

DynamicTriad [37]. It utilizes the triadic closure process to generate a graph embedding that preserves structural and evolution patterns of the graph.

Dynamic AE [35]. This method extends static autoencoders for dynamic graphs. It models the interconnection of nodes within and across time using multiple fully connected layers.

Dynamic AERNN [35]. It uses a fully connected encoder to initially acquire low dimensional hidden representation and feeds this representation into LSTMs to capture network dynamics.

SGCN [42]. It extends GCNs to signed networks. For this method, we mark the observed pairs as positive edges and the non-observed pairs as negative.

tNodeEmbed [36]. It leverages the temporal information in graphs to create rich node representations. We modified the original tNodeEmbed for handling unseen nodes in inductive learning. Given a new node, we aggregate information from its neighbors per time window with node embeddings initialized by the node features.

Node2Vec [27]. It is one of the most popular plain network embedding models.

Edge2Vec [29]. It utilizes edge semantics in its random walk node representation method by using an edge-type transition matrix.

nSNE [43]. It learns non-linear relationship between nodes in an edge embedding in signed networks.

TransR [30], TransH [33], TransE [34]. These are knowledge graph completion methods, which are modified for this task by conducting triplet $\{$head,tail,relationship$\}$ classification.

4.3.2 Results in Transductive Setting

In the transductive evaluation, we train the T-PAIR model on the training split (i.e., dataset corpus $\leq$ 2009) and evaluate its performance on the test pairs in the test time window (in our case $2010-2019$). These testing nodes have been seen in the training data. However, their relations are not clear until 2009. We thus predict their relations in $2010-2019$ based on their dynamic history before 2009 (being *negative* or *unlabeled*). The performance of our model compared to other state-of-the-art dynamic and static methods is shown in Table 2, for three evaluation datasets, respectively.

The results show that T-PAIR outperformed the baseline methods on all the three datasets. tNodeEmbed, also learns to capture and utilizes the temporal information in the data, indicating the importance of the utilization of the temporal information. However, T-PAIR outperforms tNodeEmbed, because T-PAIR learns the temporal dynamics of node-pair relationships while tNodeEmbed captures the dynamic semantics of the nodes. In addition, The improved results obtained from the combination of the graph topology-based embedding (Node2vec) and the text-based embedding (Features) highlight the enriched information property of multi-source data. The nSNE and SGCN incorporate the signs given to the edges to create the node representations. However, the node representation generated by nSNE is dependent on only the graph node topology, while SGCN directly incooperates the text-based attribute embedding of the nodes in learning a node representation. SGCN performance, in comparison to nSNE, additionally shows that the incorporation of text-based attributes in the network embedding process enriches the learned representations.

4.3.3 Results in Inductive Setting

The inductive evaluation setting allows the testing pairs with at least one new unseen node (not available in the training graphs). The obtained results reported in Table 2 show that T-PAIR can generate embedding for new nodes to capture the relationship between the new nodes and the

$*$. Second best performance

†. For the GraphSAGE and T-PAIR models, we report only the aggregation variant that gave the best result - maxpool aggregation.

‡. For graph topology based methods, we append the node features to the learned embedding for classification

TABLE 2: Evaluation results on the Virology, Immuniotherapy and Neurology datasets respectively; showing the Macro-F1 score (F1-Macro), F1 Score of observed connections (F1-Positive), and the AUC score. The evaluation is performed in transductive settings (only nodes/terms observed in training), inductive settings (pairs with at least one new node/term unobserved in training), and comprehensive settings (with both observed and unobserved nodes/terms in training)

	Virology								
	Transductive			Inductive			Comprehensive		
	F1-Positive	F1-Macro	AUC	F1-Positive	F1-Macro	AUC	F1-Positive	F1-Macro	AUC
Feature	0.56	0.71	0.80	0.47	0.71	0.79	0.55	0.72	0.83
Edge2vec^z	0.18	0.48	0.69	0.18	0.56	0.74	0.18	0.52	0.73
Node2vec^z	0.76	0.84	0.93	0.15	0.50	0.54	0.58	0.72	0.82
GraphSAGE^y	0.31	0.55	0.70	0.15	0.54	0.80	0.28	0.57	0.78
nSNE^z	0.27	0.53	0.71	0.20	0.57	0.66	0.26	0.56	0.72
TransH	0.17	0.25	0.27	0.29	0.49	0.57	0.28	0.53	0.53
TransR	0.34	0.31	0.38	0.02	0.19	0.16	0.23	0.38	0.40
TransE	0.28	0.43	0.43	0.27	0.55	0.58	0.26	0.41	0.43
SGCN	0.67	0.74	0.84	0.50	0.72	0.87	0.64	0.76	0.86
Dyntriad^z	0.59	0.72	0.84	0.50	0.70	0.85	0.57	0.72	0.84
Dynaernn^z	0.49	0.65	0.80	0.42	0.68	0.76	0.48	0.68	0.82
Dynae^z	0.57	0.71	0.80	0.45	0.70	0.70	0.55	0.72	0.78
tNodeEmbed	0.73	0.82	0.91	0.49	0.72	0.84	0.69	0.81	0.89
T-PAIR^y	**0.83**	**0.88**	**0.95**	**0.63**	**0.80**	**0.88**	**0.80**	**0.87**	**0.94**

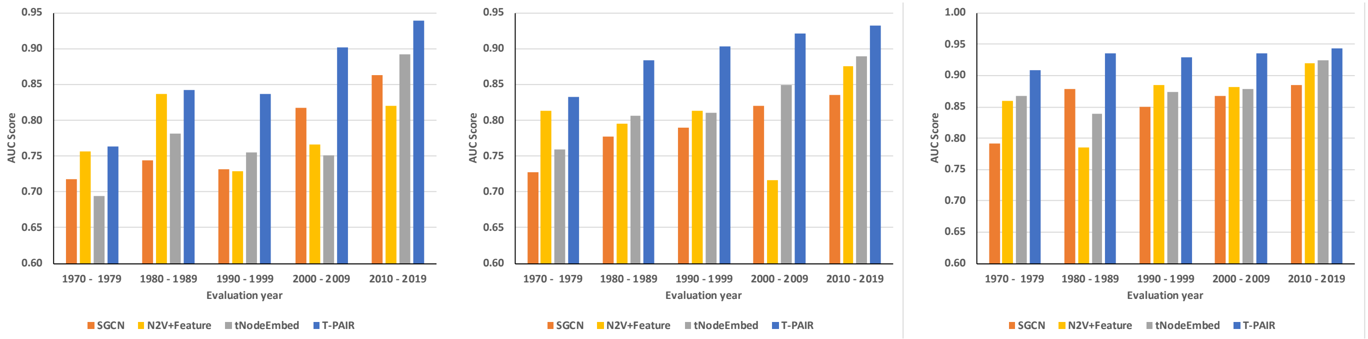
	Immunotherapy								
	Transductive			Inductive			Comprehensive		
	F1-Positive	F1-Macro	AUC	F1-Positive	F1-Macro	AUC	F1-Positive	F1-Macro	AUC
Feature	0.53	0.70	0.80	0.42	0.68	0.77	0.51	0.70	0.82
Edge2vec^z	0.14	0.48	0.70	0.04	0.48	0.73	0.12	0.49	0.73
Node2vec^z	0.75	0.84	0.93	0.23	0.58	0.65	0.67	0.80	0.88
GraphSAGE^y	0.26	0.55	0.71	0.19	0.56	0.55	0.25	0.55	0.67
nSNE^z	0.29	0.56	0.75	0.24	0.58	0.67	0.28	0.58	0.74
TransH	0.16	0.24	0.25	0.36	0.57	0.64	0.37	0.61	0.60
TransR	0.31	0.44	0.45	0.45	0.59	0.65	0.56	0.66	0.66
TransE	0.22	0.38	0.39	0.34	0.58	0.62	0.44	0.65	0.64
SGCN	0.59	0.70	0.81	**0.48**	**0.71**	0.86	0.58	0.72	0.84
Dyntriad^z	0.57	0.72	0.84	0.23	0.32	0.39	0.42	0.59	0.65
Dynaernn^z	0.41	0.63	0.77	0.36	0.65	0.76	0.40	0.64	0.79
Dynae^z	0.40	0.63	0.76	0.29	0.61	0.65	0.39	0.64	0.76
tNodeEmbed	0.71	0.82	0.90	0.31	0.63	0.85	0.65	0.80	0.89
T-PAIR^y	**0.81**	**0.87**	**0.95**	0.46	0.70	**0.88**	**0.76**	**0.85**	**0.93**

	Neurology								
	Transductive			Inductive			Comprehensive		
	F1-Positive	F1-Macro	AUC	F1-Positive	F1-Macro	AUC	F1-Positive	F1-Macro	AUC
Feature	0.67	0.72	0.82	0.53	0.73	0.81	0.66	0.75	0.85
Edge2vec^z	0.64	0.67	0.72	0.37	0.64	0.75	0.60	0.69	0.76
Node2vec^z	0.86	0.87	0.94	0.00	0.46	0.67	0.78	0.84	0.90
GraphSAGE^y	0.63	0.66	0.71	0.37	0.64	0.76	0.60	0.69	0.77
nSNE^z	0.66	0.70	0.75	0.28	0.23	0.34	0.54	0.56	0.54
TransH	0.33	0.30	0.31	0.44	0.58	0.64	0.44	0.47	0.49
TransR	0.23	0.41	0.41	0.24	0.49	0.53	0.42	0.62	0.63
TransE	0.45	0.57	0.58	0.08	0.33	0.31	0.70	0.76	0.76
SGCN	0.74	0.78	0.87	0.51	0.72	**0.87**	0.71	0.79	0.89
Dyntriad^z	0.77	0.80	0.88	0.22	0.40	0.49	0.64	0.70	0.78
Dynaernn^z	0.70	0.73	0.83	0.23	0.37	0.53	0.58	0.64	0.74
Dynae^z	0.69	0.74	0.82	0.24	0.37	0.37	0.56	0.64	0.65
tNodeEmbed	0.83	0.86	0.93	0.62	0.78	**0.87**	0.80	0.86	0.92
T-PAIR^y	**0.88**	**0.89**	**0.95**	**0.64**	**0.79**	**0.87**	**0.86**	**0.89**	**0.94**

already existing nodes in the graph. This capability is made possible due to the temporal neighborhood aggregation strategy of T-PAIR. In aggregating the node neighborhood of the new nodes, the temporal properties of the new node neighbors are disseminated to the new nodes, and thus leading to a better node representation for the new nodes, and then a better relation prediction.

Another observation is that using just the text-based feature outperformed the methods using a combination of graph topology embedding and text-based features in an inductive setting. We postulate that with lack of additional information on new nodes, methods trained to generate node representation from graph topology fails to capture the topological characteristics of the new nodes. This poor performance is because the new nodes were not available in the training graph set. Hence, the obtained representation becomes detrimental to the overall performance in combination with the text-based features. On the other hand, the text-based features capture the textual information of the nodes and is thus a useful information source for node relationship evaluation.

To better understand the performance of the model in



(a) Virology

(b) Immunotherapy

(c) Neurology

Fig. 3: AUC score of incremental prediction (per year) made by T-PAIR and three other baselines. The models are incrementally trained with data before the evaluation time period.

a real-world setting, we also evaluate and report the performance of the models in a comprehensive setting, which includes pairs with unobserved or observed nodes/terms. The obtained results show that T-PAIR outperformed the baseline methods in the three datasets.

4.3.4 Incremental Prediction

The above-presented evaluation results show that T-PAIR well captures how the connections between terms have been formed in a dynamic process. Therefore, T-PAIR predicts more accurately the relations at last time step T in the testing period than other existing approaches, in both transductive and inductive setting. A thorough evaluation of T-PAIR's capability of handling the dynamically evolving relations is to evaluate T-PAIR in an incremental training manner. That is to say, we train T-PAIR on data until $t-1$ and evaluate its performance on predicting the testing pairs in t.

As shown in Figure 3, we evaluate T-PAIR's prediction performance on testing pairs in $1970-1979$ by using all training data until 1969, performance in time interval $1980-1989$ by training until 1979, and so forth. We compare with several models trained in a similar way, such as Node2Vec+Feature, SGCN, and tNodeEmbed. This evaluation is conducted in comprehensive setting.

We observe that T-PAIR outperformed the other methods in all periods. This observation shows that T-PAIR incrementally learns from the temporal network information, improving the representations learned with each time step. The static methods gave a different performance at each year range. This performance shows that these methods learn on each single graph shots. Hence, the performance is dependent on the information extracted at each year range. The tNodeEmbed method overall displayed improved performance at each consecutive year range also. This again highlights the importance of the consideration of the temporal dynamic relation evolution process.

4.3.5 Pair Embedding Visualization

We further analyze the pair embedding learned by T-PAIR. We visualize the sampled pairs in the neurology data using the *t-SNE* method [44]. For clear visibility, we randomly sample 800 pairs and visualize the learned embeddings in Figure 4. We denote with colors the true label in comparison to the predicted labels. In the visualization, we observe that

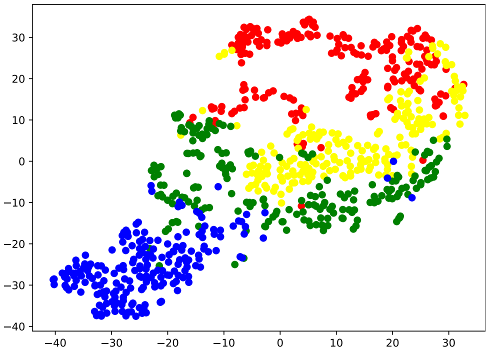
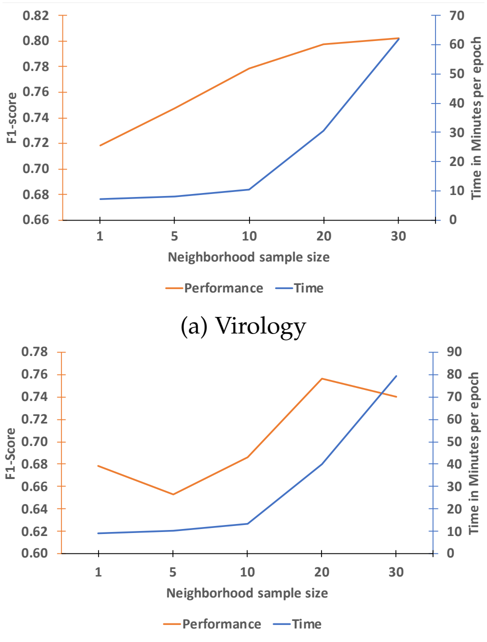


Fig. 4: Pair embedding visualization. The blue color denotes the *true positive samples*, the red points are *true negative*, the green points are *false positive*, and the yellow points are *false negative*.

the true positives (blue) and true negatives (red) are further apart. In contrast, the false positives (green) and false negatives (yellow) are both closer to the true positives (blue). We postulate that this is because we can say with higher certainty based on some research that a term pair should or should not be connected in the next time window. However, sometimes the term pair might not be discovered in the next time window because researchers did not manually discover the relationship in order to study it, or they discovered the relationship late in the coming years after the time window. On the other hand, a relationship can be weak based on the literature information but was discovered manually by wet-lab experiments. Hence our aim in this paper is not to discourage the traditional lab experimental studies but to mutually connect it with automatic HG for a more enhanced research experience.

4.3.6 Parameter Sensitivity Study

We analyzed the sensitivity of our method to the model depth M (M-hop neighborhood). We found that $M=2$ was the optimal setting for a consistent performance boost in comparison to $M=1$ and $M\geq 3$. This observation matches the finding in [40]. Thus we omit such obvious results and presents the sensitivity of T-PAIR on the neighborhood sample size. Figure 5a and 5b show results evaluated in the virology and immunotherapy datasets, respectively. We



(a) Virology

(b) Immunotherapy

Fig. 5: Parameter sensitivity on the neighborhood sample size, analyzed on the Neurology and Immunotherapy datasets.

observe in Figure 5 that despite the increase in the runtime, there is no significant performance improvement with an increase of the neighborhood sample size after 20 for both datasets. Specifically, the model exhibited a diminishing return property. This analysis was performed using the *maxpool* neighborhood aggregation. Other neighborhood aggregation methods exhibited similar properties, albeit with different run times - the LSTM aggregation has the most extended runtime.

4.4 Qualitative Study

4.4.1 Five Golden Test Cases

To further analyze the performance of our method, we follow the experimental setup used in previous HG literature [20], [24], [45]. In these works of literature, the evaluation is based on replicating the five golden test-cases reported by the pioneers in this area of study. The gold standard test cases we evaluate with are :

1) Raynaud's Disease (**RD**) and Fish Oils (**FO**) (discovered in 1985)
2) Migraine Disorder (**MIG**) and Magnesium (**MG**) (discovered in 1988)
3) Arginine (**ARG**) and Somatomedin C (**IGF1**) (discovered in 1994)
4) Alzheimer Disease (**AD**) and Indomethacin (**INN**) (discovered in 1989)
5) Schizophrenia (**SZ**) and Calcium - Independent Phospholipase A2 (**PA2**) (discovered in 1997)

We use the neurology data for this analysis. Our evaluation setup is as follows: We train the model using the same setting as described in sections 4.1 and 4.2. We train the model for each year window and record the predicted probability of the golden test term pairs to be connected in the next time window. For instance, in the case of "Raynaud's Disease (RD) and Fish Oils (FO) (1985)", we give as input the term pair <*Raynaud's Disease, Fish Oils* >. The predicted probability can be seen as the confidence score of two terms to be connected in the next time window. The reproduction of the gold test-cases on a subset of the PubMed dataset shows that our method performs well even with limited data. In Table 4, we evaluate the prediction quality of our T-PAIR model against the performance of a static model **(static)** using the complete graph formed from $t = 1$ until $t^{discovered} - 1$ (before the year interval when they were discovered), and a contemporary model **(recent)** using only the graph formed in $t^{discovered} - 1$. The results confirm the usefulness of modeling the temporal link formulation process and that using only the recently formed links is often insufficient. The static model predicts *No* for the relevance of FO and RD, as the predicted relevance score is 0.12, less than 0.5. Also, using only the graph formed from 1970 to 1979, the relevance score is predicted to be 0.04, which is much lower than the value predicted by T-PAIR of 0.61. The same advantage of T-PAIR is found on prediction the link between SZ and PA2. For the pairs MIG - MG, ARG - IGF1, and AD - INN, all the three models predicted *Yes* due to their less sensitivity to the temporal link formulation process. The predicted probability of our proposed model for the pairs at different years can be seen in Table 3.

TABLE 3: Evaluation results on the golden test cases

	Evaluation test year			Discovery year
	1970 - 1979	1980 - 1989	1990 - 1999	
RD - FO	0.04	0.61	0.75	1985
MIG - MG	0.58	0.98	0.98	1988
ARG - IGF1	0.00	0.02	0.95	1994
AD - INN	0.74	0.97	0.98	1989
SZ - PA2	0.58	0.85	0.93	1997

TABLE 4: Prediction of the golden test cases, whether the term pairs in these cases are correctly predicted to be linked (**Y**) in their discovered year interval $t^{discovered}$ (with the predicted relevance scores in parentheses).

	Prediction for term pairs in the golden test cases				
	RD - FO	MIG - MG	ARG - IGF1	AD - INN	SZ - PA2
Static	N (0.12)	**Y (0.99)**	**Y (0.98)**	**Y (0.96)**	N (0.32)
Recent	N (0.04)	**Y (0.81)**	**Y (0.97)**	**Y (0.91)**	N (0.47)
T-PAIR	**Y (0.61)**	**Y (0.98)**	**Y (0.95)**	**Y (0.97)**	**Y (0.93)**

Raynaud's Disease (RD) and Fish Oils (FO). *Raynaud disease* is a disorder of the blood vessels, which causes blood vessels to narrow in response to cold or stress. Swanson [46] studied and discovered the use of dietary *fish oils* in treating patients with *Raynaud's syndrome*. In our evaluation dataset, we note that *Raynaud's disease* was first observed in the 1969 subgraph, while *Fish oil* was first observed in the 1949 subgraph. The result shows the incremental relationship learning per year window.

Migraine Disorder (MIG) and Magnesium (MG). *Migraine* is a common primary headache disorder with multiple potential triggers. In our records, *migraine* and *magnesium* were first observed in the 1949 subgraph without a link. Until in [47], their relationship was studied in 1988. The result shows that right from the start, our model had high confidence in the relationship between the two terms.

Arginine (ARG) and Somatomedin C (IGF1). The relationship between a growth-regulating peptide (*Somatomedin C*) and an essential amino acid (*Arginine*) was first analyzed in [15]. However, *Somatomedin C* was first observed in 1979, while *Arginine* was observed in 1949. The obtained result shows that with each time window, our model increasingly learned the relationship between the two terms.

Alzheimer Disease (AD) and Indomethacin (INN). *Indomethacin* and *Alzheimer* were first observed in the 1969 and 1949 publications, respectively. *Alzheimer* is a progressive disease that destroys neurological and other body functions. *Indomethacin* is a non-steroid anti-inflammatory agent. Our model predicts their high relevance early in 1970s (with a score 0.74). In 1989, *Indomethacin* was studied for the treatment of *Alzheimer*.

Schizophrenia (SZ) and Calcium - Independent Phospholipase A2 (PA2). Swanson and Smalheiser [15] postulated the link between a gene (*Calcium-Independent Phospholipase A2*) and *Schizophrenia*-a chronic mental disorder that severely affects a person ability to thinks, feel, and behave. In our dataset, we observe *Schizophrenia* and *PA2* in the 1949 and 1979 subgraphs, respectively. Our model could predict the relationship with increasing confidence.

4.4.2 Predicted Cases Analysis

To demonstrate and analyze the predicted cases, we use the neurology graph ≤ 2009 as the current graph and the graph of $\{2010 - 2019\}$ as the future graph. First, we sent a list of top 100 predicted hypotheses to a team of domain experts to evaluate the interestingness and validity. With respect to the top 100 generated hypothesis, 74% was confirmed to be of high quality, 14% was said to be plausible, and 12% was evaluated to be of low quality. Next, we analyze the prediction of the T-PAIR model trained on the term-term connections made before 2010 (i.e., $\leq$ 2009). We present the prediction analysis result in Tables 5, 6, and 7.

Our investigation targets on answering the following questions about the term pairs that are ranked on the top of the prediction list according to their predicted relevant scores:

> Do the top ranked term pairs have a connection indeed in the future graph $\{2010 - 2019\}$? (Are the predictions correct?)
>
> Do the top ranked term pairs have a short path in the current graph ($\leq$ 2009)? (Was their relevance obvious?)

We first extract the top ranked term pairs predicted by T-PAIR to be connected in the future graph, and then divide these top term pairs into three groups:

- **Group 1** including term pairs whose shortest path distances in the current graph is **more than 3**;
- **Group 2** including term pairs whose shortest path distances in the current graph is **3 (three-hop away)**;
- **Group 3** including term pairs whose shortest path distances in the current graph is **2 (two-hop away)**.

The results of **Group 1** is presented in Table 5. We observed that the top-6 predicted connections truly exist in the future graph, although the nodes associated with each of these connections were far away from each other in the current graph. This result shows that the T-PAIR model can learn not just topology-based relationships on graphs but also the semantic meaning and relationship between the term textual attributes.

TABLE 5: Top ranked term pairs in Group 1. They are predicted to be connected in future graph, but were more than 3 hops away from each other in the current graph (in the neurology dataset until 2009), i.e., shortest path distances ≥ 4 from the dataset as of 2009.

Term 1	Term 2	Connected in future graph
Apnea	BRAT1	Yes
Diabetes	Sel1L	Yes
Autoimmune diseases	DIPDHAQ^x	Yes
Multiple sclerosis	miR-572	Yes
Trauma	PACAP	Yes
Mental retardation	DOCK8	Yes

The results of **Group 2** is presented in Table 6. We observe that four of the top-six predicted connections were not observed in the future graph. We then delve deeper to probe these unobserved pairs. We were able to find co-occurrence of the <*Macroglobulinemia*, *HNF-1* > and <*Sezary syndrome*, *Antibiotic* > pairs in the full PubMed dataset occurring in [52] and [53] respectively; although the pairs were not connected in the publications used in the construction of the neurology dataset. However, we could not find any co-occurrence of the < *Tenuifoliside*, *WAGR syndrome*> and < *Interleukin-15*, *Antero-superior* | *antero-posterior*> pairs in the full PubMed dataset. On further analysis, we found that some psychiatric/behavioral and physical problems, including *depression* and *inflammations*, are some of the features of *WAGR syndrome* [48], [49]. Studies in [57] show *Tenuifoliside* has *anti-apoptotic*, *neuroprotective*, and *anti-inflammatory* effects. We also find *Osteoarthritis* is one linking factor between *Interleukin-15* and *Antero* regions [50], [51].

Finally, the results of **Group 3** is presented in Table 7. The term pairs of these top-ranked connections have shortest distances of two. Three of the term pairs were indeed connected in the future graph. Out of the neurology future graph dataset, we found the co-occurrence of the terms of the pair <*Transient ischemic attack*, *Phycoerythrin*> in [54]. This confirms the correctness of our prediction, although the paper [54] is not in the neurology term network construction. Making a simple search on the internet about *Amphiregulin* and *Triethyltin* brings up several mentions of the *Triethyltin* compounds with *Amphiregulin*. However, the term pair <*Amphiregulin*, *Triethyltin*> was not found in our downloaded version of the PubMed dataset. Several studies [55], [56] have shown that a slight change to the *Rtl1/Mart1* can result in several undesired effects on the expression of several genes. However, its relationship with the *BBS4* gene is currently not so glaring. A certain connection between *Rtl1/Mart1* and *BBS4* gene is *Protein*.

§. DIPDHAQ| O,O′-bis-(3′-iodopropyl)-1,4-dihydroxyanthraquinone

¶. Cognitive difficulties | Fatigue, cognitive/psychological or musculoskeletal symptoms

∗∗. Antero-superior insula|Antero-posterior

††. Sca-1 | Spinocerebellar ataxia type 1 | Ataxin-1

‡‡. SGBS | Simpson-golabi-behmel syndrome | developmental overgrowth | Developmental overgrowth of the gpc3-deficient

TABLE 6: Top ranked term pairs in Group 2. They are predicted to be connected in future graph, but were 3 hops away in the current graph (in the neurology dataset until 2009), i.e., shortest path distances is 3 in the dataset as of 2009.

Term 1	Term 2	Shortest Path	Connected in future graph
NOTCH1	Gcn2	- NOTCH1 - Encephalopathy - Sleep deprivation - Gcn2	Yes
Loss of consciousness	Ceftaroline	- Loss of consciousness - Infections - BAL9141 - Ceftaroline	Yes
Tenuifoliside	WAGR syndrome	- WAGR syndrome - Cognitive difficulties $^{\{}$ - Acetylcholine - Tenuifoliside	No, but clues found in [48], [49]
interleukin-15	Antero	- interleukin-15 - Cancer - Dyskinesias - Antero	No, but clues found in [50], [51]
Macroglobulinemia	HNF-1	- Macroglobulinemia - Injury to cranial nerves - C-Met - HNF-1	No, but found in [52]
Sezary syndrome	Antibiotic	- Sezary syndrome - Aermatoses - Amyloid beta peptide(1-42) - Antibiotic	No, but found in [53]

TABLE 7: Top ranked term pairs in Group 3. They are predicted to be connected in future graph, but were 2 hops away in the current graph (in the neurology dataset until 2009), i.e., shortest path distances is 2 in the dataset as of 2009.

Term 1	Term 2	Shortest Path	Connected in future graph
Spinocerebellar ataxia type 1^{yy}	c-KIT	- Spinocerebellar ataxia type 1 - corticobasal degeneration - c-KIT	Yes
Triethyltin	Amphiregulin	- Triethyltin - Calcium - Amphiregulin	No, but with internet evidence
Transient ischemic attack	Phycoerythrin	- Transient ischemic attack - P-selectin - phycoerythrin	No, but found in [54]
Kawasaki disease	Aortitis	- Kawasaki disease - Livedoid vasculitis - Aortitis	Yes
BBS4	Rtl1/Mart1	- BBS4 - Amino acid - RTL1	No, but clues found in [55], [56]
Simpson-golabi-behmel syndrome ^{zz}	Tetrasomy	- Simpson-golabi-behmel syndrome - Cancer - Tetrasomy	Yes

TABLE 8: Top ranked negative predicted term pairs. They were not connected in the connected graph (in the neurology dataset until 2009) and are not predicted to be connected in future graph.

Term 1	Term 2	Connection in next Graph
PPARalpha	Cementoma	No
COX 8-3 §§	Optochine	No
DMP695	YAP ¶¶	No
Essaven gel	Myocardon	No
PDM	HLH-m5	No
Osajin	Thiadiazine	No

4.4.3 Analysis of Predicted Negative Cases

We also analyze the predicted negative term pairs to further understand the model performance. The result is presented in Table 8. We observed that the top negatively predicted term pairs were neither connected in the current nor in the future graph. The pairs seem to not be related in the biomedical point of view. For example, in the pair < *PPARalpha, Cementoma*>, *PPARalpha* is a nuclear receptor protein that regulates the expression of a number of genes critical for lipid and lipoprotein metabolism. *Cementoma* is an odontogenic tumor of cementum (a specialized calcified substance covering the root of a tooth). The same way, in the pair <*Osajin, Thiadiazine*>, *Osajin* is the major bioactive isoflavone present in the fruit of Maclura pomifera. *Thiadiazine* on the other hand, is a synthetic six-membered heterocycle having carbon atoms, sulfur atom, nitrogen atoms and two double bonds.

4.4.4 COVID-19 case study

To evaluate the applicability of the proposed model on new diseases, we analyze the model's performance on COVID-19 dataset [5]. We train on five-year windows split till 2015, i.e., excluding the new terms in 2016-2020 in the COVID-19 graph, such as *covid-19*, *sars-cov-2*. The trained model

§§. COX 8-3| cytochrome c oxidase subunit VIII & Optochine

¶¶. YAP|Yes-associated protein

[5] obtained in march 2020 from https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge

TABLE 9: Top ranked terms predicted to be connected with term *covid-19*, trained by graphs until 2015. Verification of the existence was conducted in the graph in 2020.

Epithelial cells	Lung	Pressure
Patients	Proteinuria	Liver
Tobacco	Humans	Public Health
RNA	Test	Air
Time	Hospitalization	Cough
Clustering	Morbidity	Antibodies
Blood urea Nitrogen	Face	

then predicts the connectivity between *covid-19* as a new term and other terms, which can also be a new term or a term existing before 2015. The top predicted terms predicted to be connected with *covid-19* are shown in Table 9, with the verification in COVID-19 graph in 2020. We notice that the top terms are generally relevant to *covid-19*, and we do observe most of their connections in the graph 2016 - 2020. For instance, *Cough*, *Fever*, *SARS*, and *Hand* (washing of hands) were known to be relevant to *covid-19* when writing this paper, although it was not observed in the training graph.

5 Conclusion and Future Direction

In this paper, we pose the hypothesis generation problem as a node-pair relationship prediction task on attributed temporal graphs and propose T-PAIR - a novel node-pair representation learning method on temporal graphs. To utilize the temporal property, at each time step, the model considers the previous history and the current neighborhood information of both nodes in the pair. The aggregated information is then used to predict the probability of the nodes to be connected in the next time step. The quantitative experiments and analyses show that T-PAIR outperforms several state-of-the-art methods in both inductive and transductive settings. The qualitative analyses show also the effectiveness and usefulness of the proposed method.

An exciting direction for future work is the extension of the method to directed temporal graphs such that a directed path can be obtained linking two terms. Another idea would be to incorporate the relationship type into the model (e.g., Drug-Disease type).

Acknowledgments

This work was supported and funded by Computational Bioscience Research Center (CBRC), King Abdullah University of Science and Technology (KAUST), and National Science Foundation of China (NSFC No. 61828302). We would also like to thank Ayako Yachie from The Systems Biology institute for her help in validating and checking the list of hypotheses generated by our algorithm.

References

[1] R. D. King, K. E. Whelan, F. M. Jones, P. G. Reiser, C. H. Bryant, S. H. Muggleton, D. B. Kell, and S. G. Oliver, "Functional genomic hypothesis generation and experimentation by a robot scientist," *Nature*, vol. 427, no. 6971, p. 247, 2004.

[2] J. Sybrandt and I. Safro, "Validation and topic-driven ranking for biomedical hypothesis generation systems," *bioRxiv*, 2018.

[3] D. R. Swanson, "Undiscovered public knowledge," *The Library Quarterly*, vol. 56, no. 2, pp. 103–118, 1986.

[4] N. R. Smalheiser and D. R. Swanson, "Calcium-independent phospholipase a2 and schizophrenia," *Archives of General Psychiatry*, vol. 55, no. 8, pp. 752–753, 1998.

[5] B. M. Ross, C. Hudson, J. Erlich, J. J. Warsh, and S. J. Kish, "Increased phospholipid breakdown in schizophrenia: evidence for the involvement of a calcium-independent phospholipase a2," *Archives of general psychiatry*, vol. 54, no. 5, pp. 487–494, 1997.

[6] C.-F. Kuo, S. Cheng, and J. R. Burgess, "Deficiency of vitamin e and selenium enhances calcium-independent phospholipase a2 activity in rat lung and liver," *The journal of nutrition*, vol. 125, no. 6, pp. 1419–1429, 1995.

[7] B.-C. Björk, A. Roos, and M. Lauri, "Global annual volume of peer reviewed scholarly articles and the share available via different open access options." in *ELPUB*, 2008, pp. 178–186.

[8] P. Larsen and M. Von Ins, "The rate of growth in scientific publication and the decline in coverage provided by science citation index," *Scientometrics*, vol. 84, no. 3, pp. 575–603, 2010.

[9] N. R. Smalheiser and D. R. Swanson, "Using arrowsmith: a computer-assisted approach to formulating and assessing scientific hypotheses," *Computer methods and programs in biomedicine*, vol. 57, no. 3, pp. 149–153, 1998.

[10] D. Hristovski, C. Friedman, T. C. Rindflesch, and B. Peterlin, "Exploiting semantic relations for literature-based discovery," in *AMIA annual symposium proceedings*, vol. 2006, 2006, p. 349.

[11] J. Sybrandt, M. Shtutman, and I. Safro, "Moliere: Automatic biomedical hypothesis generation system," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 1633–1642.

[12] D. Weissenborn, M. Schroeder, and G. Tsatsaronis, "Discovering relations between indirectly connected biomedical concepts," *Journal of biomedical semantics*, vol. 6, no. 1, p. 28, 2015.

[13] S. Spangler, A. D. Wilkins, B. J. Bachman, M. Nagarajan, T. Dayaram, P. Haas, S. Regenbogen, C. R. Pickering, A. Comer, J. N. Myers *et al.*, "Automated hypothesis generation based on mining scientific literature," in *KDD*, 2014, pp. 1877–1886.

[14] F. Shi, J. G. Foster, and J. A. Evans, "Weaving the fabric of science: Dynamic network models of science's unfolding structure," *Social Networks*, vol. 43, pp. 73–85, 2015.

[15] D. R. Swanson and N. R. Smalheiser, "An interactive system for finding complementary literatures: a stimulus to scientific discovery," *Artificial intelligence*, vol. 91, no. 2, pp. 183–203, 1997.

[16] V. Gopalakrishnan, K. Jha, A. Zhang, and W. Jin, "Generating hypothesis: Using global and local features in graph to discover new knowledge from medical literature," in *BICOB*, 2016, pp. 23–30.

[17] S. Spangler, *Accelerating Discovery: Mining Unstructured Information for Hypothesis Generation*. Chapman and Hall/CRC, 2015.

[18] R. K. Srihari, L. Xu, and T. Saxena, "Use of ranked cross document evidence trails for hypothesis generation," in *KDD*, 2007, pp. 677–686.

[19] S. H. Baek, D. Lee, M. Kim, J. H. Lee, and M. Song, "Enriching plausible new hypothesis generation in pubmed," *PloS one*, vol. 12, no. 7, p. e0180539, 2017.

[20] K. Jha, G. Xun, Y. Wang, and A. Zhang, "Hypothesis generation from text based on co-evolution of biomedical concepts," in *KDD*, 2019, pp. 843–851.

[21] G. Xun, K. Jha, V. Gopalakrishnan, Y. Li, and A. Zhang, "Generating medical hypotheses based on evolutionary medical concepts," in *ICDM*, 2017, pp. 535–544.

[22] M. Weeber, H. Klein, A. R. Aronson, J. G. Mork, L. De Jong-van Den Berg, and R. Vos, "Text-based discovery in biomedicine: the architecture of the dad-system." in *Proceedings of the AMIA Symposium*, 2000, p. 903.

[23] R. K. Lindsay and M. D. Gordon, "Literature-based discovery by lexical statistics," *Journal of the American Society for Information Science*, vol. 50, no. 7, pp. 574–587, 1999.

[24] P. Srinivasan, "Text mining: generating hypotheses from medline," *Journal of the American Society for Information Science and Technology*, vol. 55, no. 5, pp. 396–413, 2004.

[25] P. Srinivasan and B. Libbus, "Mining medline for implicit links between dietary substances and diseases," *Bioinformatics*, vol. 20, no. suppl_1, pp. i290–i296, 2004.

[26] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *KDD*, 2014, pp. 701–710.

[27] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *KDD*, 2016, pp. 855–864.

[28] N. Zhao, H. Zhang, M. Wang, R. Hong, and T.-S. Chua, "Learning content–social influential features for influence analysis," *International Journal of Multimedia Information Retrieval*, vol. 5, no. 3, pp. 137–149, 2016.

[29] Z. Gao, G. Fu, C. Ouyang, S. Tsutsui, X. Liu, J. Yang, C. Gessner, B. Foote, D. Wild, Y. Ding *et al.*, "edge2vec: Representation learning using edge semantics for biomedical knowledge discovery," *BMC bioinformatics*, vol. 20, no. 1, p. 306, 2019.

[30] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *AAAI*, 2015.

[31] G. Ji, K. Liu, S. He, and J. Zhao, "Knowledge graph completion with adaptive sparse transfer matrix," in *AAAI*, 2016.

[32] B. Shi and T. Weninger, "Proje: Embedding projection for knowledge graph completion," in *AAAI*, 2017.

[33] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes." in *AAAI*, vol. 14, no. 2014, 2014, pp. 1112–1119.

[34] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *NeurIPS*, 2013, pp. 2787–2795.

[35] P. Goyal, S. R. Chhetri, and A. Canedo, "dyngraph2vec: Capturing network dynamics using dynamic graph representation learning," *arXiv preprint arXiv:1809.02657*, 2018.

[36] U. Singer, I. Guy, and K. Radinsky, "Node embedding over temporal graphs," *arXiv preprint arXiv:1903.08889*, 2019.

[37] L. Zhou, Y. Yang, X. Ren, F. Wu, and Y. Zhuang, "Dynamic network embedding by modeling triadic closure process," in *AAAI*, 2018.

[38] M. Rahman, T. K. Saha, M. A. Hasan, K. S. Xu, and C. K. Reddy, "Dylink2vec: Effective feature representation for link prediction in dynamic networks," *arXiv preprint arXiv:1804.05755*, 2018.

[39] NCBI, "Ncbi resource coordinators. pubmed." [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/

[40] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *NeurIPS*, 2017, pp. 1024–1034.

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

[42] T. Derr, Y. Ma, and J. Tang, "Signed graph convolutional networks," in *ICDM*, 2018, pp. 929–934.

[43] W. Song, S. Wang, B. Yang, Y. Lu, X. Zhao, and X. Liu, "Learning node and edge embeddings for signed networks," *Neurocomputing*, vol. 319, pp. 42–54, 2018.

[44] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[45] M. Yetisgen-Yildiz and W. Pratt, "A new evaluation methodology for literature-based discovery systems," *Journal of biomedical informatics*, vol. 42, no. 4, pp. 633–643, 2009.

[46] D. R. Swanson, "Fish oil, raynaud's syndrome, and undiscovered public knowledge," *Perspectives in biology and medicine*, vol. 30, no. 1, pp. 7–18, 1986.

[47] ——, "Migraine and magnesium: eleven neglected connections," *Perspectives in biology and medicine*, vol. 31, no. 4, pp. 526–557, 1988.

[48] K. Hashimoto, "Bdnf and obesity in the wagr syndrome," *Clinical Psychopharmacology and Neuroscience*, vol. 7, no. 2, p. 63, 2009.

[49] "Wagr syndrome - genetics home reference - nih." [Online]. Available: https://ghr.nlm.nih.gov/condition/wagr-syndrome

[50] C. Scanzello, E. Umoh, F. Pessler, C. Diaz-Torne, T. Miles, E. Dicarlo, H. Potter, L. Mandl, R. Marx, S. Rodeo *et al.*, "Local cytokine profiles in knee osteoarthritis: elevated synovial fluid interleukin-15 differentiates early from end-stage disease," *Osteoarthritis and Cartilage*, vol. 17, no. 8, pp. 1040–1048, 2009.

[51] G. A. Sheridan and J. P. Cashman, "Osteoarthritis of the hip joint," in *Hip Surgeries*. IntechOpen, 2018.

[52] C. Jiménez, S. Alonso-Álvarez, M. Alcoceba, G. Ordóñez, M. García-Álvarez, M. Prieto-Conde, M. Chillón, A. Balanzategui, R. Corral, L. Marín *et al.*, "From waldenström's macroglobulinemia to aggressive diffuse large b-cell lymphoma: a whole-exome analysis of abnormalities leading to transformation," *Blood cancer journal*, vol. 7, no. 8, pp. e591–e591, 2017.

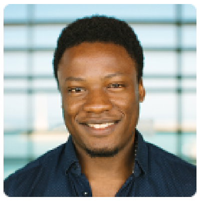
[53] D. Mermin, A. Védie, M. Jullie, A. Fauconneau, M. Beylot-Barry, and A. Pham-Ledard, "Cutaneous granular bacteriosis occurring in staphylococcus aureus septicaemia," in *Annales de dermatologie et de venereologie*, vol. 144, no. 4, 2017, pp. 275–278.

[54] R. Meamar, H. Nikyar, L. Dehghani, M. Talebi, M. Dehghani, M. Ghasemi, B. Ansari, and M. Saadatnia, "The role of endothelial progenitor cells in transient ischemic attack patients for future cerebrovascular events," *Journal of research in medical sciences: the official journal of Isfahan University of Medical Sciences*, vol. 21, 2016.

[55] C. A. Bidwell, L. N. Kramer, A. C. Perkins, T. S. Hadfield, D. E. Moody, and N. E. Cockett, "Expression of peg11 and peg11as transcripts in normal and callipyge sheep," *BMC biology*, vol. 2, no. 1, p. 17, 2004.

[56] Y. Sekita, H. Wagatsuma, K. Nakamura, R. Ono, M. Kagami, N. Wakisaka, T. Hino, R. Suzuki-Migishima, T. Kohda, A. Ogura *et al.*, "Role of retrotransposon-derived imprinted gene, rtl1, in the feto-maternal interface of mouse placenta," *Nature genetics*, vol. 40, no. 2, p. 243, 2008.

[57] "Tenuifoliside a: Cas:139726-35-5: Manufacturer chemfaces." [Online]. Available: http://www.chemfaces.com/natural/Tenuifoliside-A-CFN90770.html



Uchenna Akujuobi is working toward the PhD degree in the MINE Laboratory at the Division of Computer, Electrical, and Mathematical Sciences and Engineering, King Abdullah University of Science and Technology (KAUST), Saudi Arabia. This research was started during his time at the Sony Computer Science Laboratories Inc. His research interests include network embedding, graph mining, information retrieval, text mining, and deep neural networks.

Michael Spranger is currently a researcher and senior program manager at Sony Computer Science Laboratories Inc. and a Deputy General Manager at Sony AI based in Tokyo, Japan. He recieved the Ph.D. degree in computer science from the Vrije Universiteit in Brussels (Belgium) in 2011 (in Computer Science/Artificial Intelligence). He has published more than 60 peer-reviewed papers and holds patents on personal information processing using NLP. He is active in the AI community, developmental robotics and computational linguistics.

Sucheendra K. Palaniappan Sucheendra K. Palaniappan is currently a Senior Scientist at the Systems Biology Institute, Tokyo. He also leads the data science and engineering teams at SBX corporation, Tokyo as its Director of Data science and Engineering. Previously, he was a post-doctoral fellow at INRIA, Rennes with the SUMO Team and INRIA, Saclay with the Lifeware Team. He got his PhD in Computer science from the Department of computer science, School of computing, National University of Singapore. His core interests lies at the cross roads of computational and complex system

Xiangliang Zhang is currently an Associate Professor and directs the Machine Intelligence and Knowledge Engineering (MINE) Laboratory at the Division of Computer, Electrical, and Mathematical Sciences and Engineering, King Abdullah University of Science and Technology (KAUST), Saudi Arabia. She received the Ph.D. degree in computer science from INRIA-University Paris-Sud, France, in July 2010. She has authored or co-authored over 130 refereed papers in various journals and conferences. Her current research interests and experiences include machine learning, and data mining.