EasyChair Preprint

№ 11464

MAT: Effective Link Prediction via Mutual Attention Transformer

Van Quan Nguyen, Quang Huy Pham, Quang Dan Tran,
Kien Bao Thang Nguyen and Hieu Nghia Nguyen

December 6, 2023

MAT: Effective Link Prediction via Mutual Attention Transformer

Quan Van Nguyen[1,2,3], Huy Quang Pham[1,2,4], Dan Quang Tran[1,2,5]
Thang Kien-Bao Nguyen[1,2,6], Nghia Hieu Nguyen[1,2,7]
[1]Faculty of Information Science and Engineering, University of Information Technology, Ho Chi Minh City, Vietnam
[2]Vietnam National University, Ho Chi Minh City, Vietnam
Email: {[3]21521333, [4]21522163, [5]21521917, [6]21521432, [7]19520178}@gm.uit.edu.vn

***Abstract*—The Data Science and Advanced Analytics (DSAA) 2023 competition [1] focuses on proposing link prediction methods to solve challenges about network-like data structure, such as network reconstruction, network development, etc., from articles on Wikipedia. In this challenge, our "UIT Dark Cow" team proposes the Mutual Attention Transformer (MAT) method to predict if there is a link between two Wikipedia pages. Our method achieved the 5th and 4th position on the leaderboard for the public and private tests, respectively. Our source code is publicly available for the ease of experimental re-implementation at the following link: *https://github.com/minhquan6203/source-code-dsaa-2023*.**

***Index Terms*—DSAA 2023, Link Prediction, Attention, Transformer**

I. Introduction

The DSAA (Data Science and Advanced Analytics) 2023 is a part of the DSAA 2023 conference. This challenge aims to discover effective approaches that can indicate whether or not a link exists between two Wikipedia pages. In particular, given a pair of nodes (u, v) along with information about each node, the main objective is to determine if there is a link between the two nodes, in which the label of 1 indicates "yes" and the label of 0 indicates "no". The final decision will be based on the effectiveness of the method (the score), performance (the required time for the training and testing process), and the novelty of the proposed method.

By successfully predicting the presence or absence of a link, we contribute to exploring the underlying connections and relationships available in the Wikipedia network. The outcome of this task is essential for various applications such as network reconstruction, recommendations, and understanding network evolution.

In this paper, we present the Vanilla Fully Connected (VFC) method and the Mutual Attention Transformer (MAT) method. The VFC method only uses id information. In contrast, the MAT method can use the id and textual information of both nodes. Based on the results of the competition, we can confidently confirm that the VFC and MAT are effective in solving the task proposed in DSAA Challenge 2023.

II. Related Works

A. Graph Neural Networks

Zhang et al. [2] have highlighted that heuristics often rely on strong assumptions regarding the likelihood of connections between nodes, leading to limited effectiveness when these assumptions do not hold in certain networks. Consequently, the authors have proposed a methodology for extracting localized subgraphs surrounding target links. The objective of their study was to develop a function that can map subgraph patterns to the existence of links, facilitating the automatic learning of a tailored "heuristic" suitable for the specific network under analysis. Through their findings, which indicated that local subgraphs possess valuable information pertaining to link existence, the authors introduced a novel approach for learning heuristics based on Graph Neural Networks (GNNs). The utilization of GNN-based techniques enables the capture and utilization of structural characteristics within the subgraphs, enabling accurate predictions regarding the presence or absence of links.

B. Graph Convolution Embedded LSTM

GC-LSTM [3] is a graph convolution embedded LSTM networks for dynamic link prediction. GC-LSTM has some merits over existing methods, such as being able to capture both the local and global structural changes of the graph over time by using graph convolutional networks (GCNs), being able to handle graphs with varying sizes and densities by using long short-term memory networks (LSTMs), having the capacity to learn from node and edge features through the utilization of feature matrices as inputs to GCNs and LSTMs, this approach becomes more practical compared to many existing methods that exclusively handle removed links. Moreover, it enables the prediction of both added and removed links.

C. Graph Transformer Networks

The Graph Transformer Networks (GTNs) was introduced in 2019 by Seongjun et al. [4] as an innovative approach aimed at addressing the limitations of previous methods based on Graph Neural Networks (GNNs) in tasks such as

Link Prediction and Node Classification [5]. Previous GNN-based methods [6] were primarily designed to learn node representations on fixed and homogeneous graphs, thereby encountering challenges when dealing with misspecified or heterogeneous graphs comprising diverse node and edge types. In order to overcome these limitations, GTN was proposed as a solution capable of generating novel graph structures. This involves identifying meaningful connections between initially unconnected nodes within the original graph while concurrently learning effective node representations on these newly constructed graphs in an end-to-end manner. To evaluate the performance of GTNs, comparisons were made against state-of-the-art methods that rely on predefined meta-paths derived from domain knowledge [6]. Notably, GTN demonstrated superior performance in all three benchmark node classification tasks, thereby eliminating the need for domain-specific graph preprocessing.

D. Dynamic Self-Attention Networks

Dynamic Self-Attention Networks (DySAT) was introduced by Aravind et al. [7]. DySAT is a deep neural network that leverages self-attention mechanisms to learn node representations on dynamic graphs. DySAT has some advantages over existing methods, such as being able to capture both the local and global structural changes of the graph over time, being able to handle graphs with varying sizes and densities, and being able to learn from both node and edge features. DySAT has demonstrated its effectiveness on link prediction tasks on different types of dynamic graphs, such as communication networks and bipartite rating networks.

III. Proposed Methods

A. Task Definition

In the DSAA 2023 Challenge, participants are given a dataset of node pairs (u, v) and need to predict if there is an edge between them. The task focuses on link prediction for Wikipedia articles, where the goal is to determine if a link exists between two Wikipedia pages. The dataset has been modified to remove 20% of the information, including positive and negative pairs. Positive pairs indicate the presence of an edge, while negative pairs indicate its absence.

B. Data Pre-processing

In order to carefully prepare the data for the modeling step, we applied some pre-processing techniques as detailed in the following:

1) Removing unnecessary characters: Before starting to train the model, we removed unnecessary characters such as punctuation, special characters, numbers or other special characters that have no language meaning.
2) Removing stop words: We removed common words in the English language and words that appear a lot in the dataset but these words do not have important meanings.
3) Lowercasing: We finished this with the main purpose of converting text to lowercase to homogenize text data and avoid unnecessary duplication.

C. Vanilla Fully Connected

We conduct experiments to discover whether or not there is a correlation between the nodes by their id. To this end, we propose the Vanilla Fully Connected Method (VFC), which contains an embedding module to vectorize the id of nodes as input. Then we concatenate the outcome vectors in order to form the final features for classifying the output showing whether there are any edges between given nodes or not. A detailed structure of VFC method is illustrated in Figure 1.

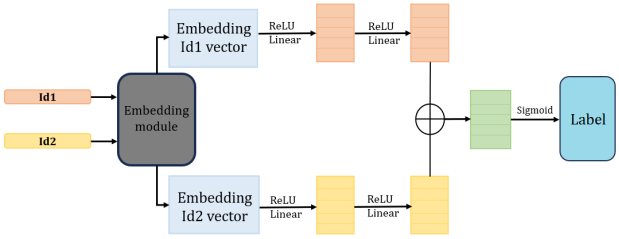

Fig. 1: The structure of Vanilla Fully Connected.

However, when training a deep neural network, an important problem that can arise is gradient vanishing. This occurs when the gradient of the loss function degenerates to zero as it propagates back through the layers of the network, causing weight updating to become very slow or nothing to be updated. To avoid gradient vanishing, two important weighted initialization ways, Xavier Initialization and He Initialization are used. The results are shown in TableV.

Xavier Initialization [8], designed to maintain approximately the same gradient magnitude across each layer of the neural network. The VFC method uses Xavier Initialization for embedding weights.

He Initialization [8], similar to Xavier Initialization, but more suitable for models that use an activation function with no bounds. With FC layers, this initialization way is applied.

D. Mutual Attention Transformer

In the following section, we will show that the VFC method successfully achieved the nearly absolute score on the public test and private test in the dataset given in the DSAA Competition 2023, we can simply model the assigned task using the id of nodes only. However, in the context of the link prediction task, we want to research the underlying relationships between two nodes of passages rather than using their id. The VFC method can be effective on this particular dataset, but it is not fit for the link prediction task in general. Accordingly, we developed another method to meet the natural requirement of the link prediction task.

We want to jointly learn the mutual relation between the information of given two nodes of passages, then classify them as connected or not depending on the extracted mutual information. From that on, we inspired the proposed method for the multimodal learning task. In particular, we inherited the co-attention technique of ViLBERT [9]. The co-attention technique parallelly learns the information which is presented

in the image that benefits the given question and vice versa in order to obtain the guided features. After that, they fuse these two guided features to finally select the appropriate answer. In the DSAA competition 2023, we designed the Mutual Attention (MA) module (Figure 2) following the co-attention technique to learn the mutual information of two nodes and then determine the existence of a link between them.

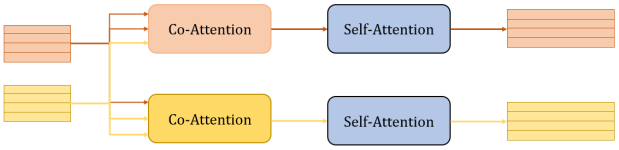

Fig. 2: Mutual Attention Module.

The passages of two input nodes are first forwarded through an embedding module to obtain the feature vectors. Then, these feature vectors are fed into the MA module where the information between the passages in two nodes is mutually attended by the other and self-attention by themselves. Then, these two attended features will be concatenated. This fused vector is then passed through the linear layer to classify into two categories. This method uses cross entropy loss instead of binary cross entropy loss because we want the MAT method can be adaptable to other related tasks when the number of distinct labels is more than two such as fact-checking, sentence pair classification, natural language inference, etc. Figure 3 below describes the method we designed, which uses the MA module to solve this problem.

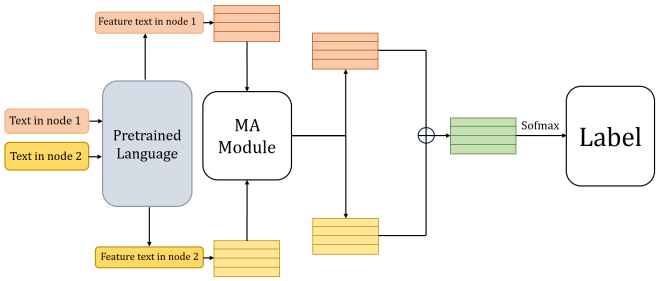

Fig. 3: The structure of Mutual Attention Transformer.

Moreover, through testing, we found that the id of the nodes in this dataset plays a very important role. By using them, we can train the VFC method to get a high score. In order to incorporate the pattern of id to this method, we redesigned the MAT method to be able to use the id in combination with the passages in each node. The result of using both the id and the passages is more effective than using only passages. We will analyze this behavior carefully in Section IV-D.

IV. Experimental Results

A. The evaluation metric

The evaluation metric for this competition [1] is Macro F1-Score determined by:

$$F1 = 2\frac{p \cdot r}{p + r} \tag{1}$$

where

$$p = \frac{tp}{tp + fp} \tag{2}$$

$$r = \frac{tp}{tp + fn} \tag{3}$$

B. Experimental Configuration

For the MAT method, the pre-trained language model which is in our experiment is BERT-base-uncased [10]. This pre-trained language model is used as the embedding module for the passages of nodes as input. By default, we did not freeze it for best performance. Pre-trained weights of BERT-base-uncased were loaded from the checkpoint of Huggingface.

We used PyTorch's Embedding (torch.nn.Embedding) as the embedding module for the VFC method. Detailed information about the hyperparameters is presented in Table I.

We trained our proposed methods using an Intel Xeon CPU consisting of two virtual CPUs (vCPUs) and 13GB of RAM. To efficiently handle the substantial data size, we have opted for an NVIDIA Tesla K80 GPU equipped with 15GB of VRAM (Video Random Access Memory).

TABLE I: Configurations for each method.

Hyperparameters	MAT	VFC
embedding	bert-base-uncased	torch.nn.Embedding
freeze	False	-
batch size	150	1024
max length	64	-
epoch	100	100
patience	5	5
optimizer	AdamW	Adam
learning rate	3 x 10^{-5}	1 x 10^{-3}
metric evaluate	F1-Score	F1-Score
dropout	0.2	0.2
dim of model	128	256
loss function	Cross entropy	Binary cross entropy

C. Results

The competition results of our methods are displayed in Table II. Both methods yielded excellent results, which can be seen as successfully solving the problem presented by the competition. Each of the methods we propose has its own advantages and disadvantages.

With the VFC method, although the training time is very short, only a couple of minutes for each epoch with F1-Score is almost perfect, but this method is unable to exploit textual information. Consider the MAT method, although the training time is longer than the VFC method, it has maximized the information from the provided dataset along with the excellent performance it brings.

TABLE II: Results on the public and private tests.

Method	Public test	Private test
VFC (ours)	0.99999	1.00000
MAT (ours)	0.99996	1.00000

D. Ablation study

As described in the previous section, we have various scenarios to provide input for our proposed methods: using only id, using only passages, and using both id and passages.

From Table II, the VFC method performed excellently in the required training time and performance in the public and private tests. However, in order to make full usage of the information from the dataset and to fit the task definition, we extended to use the passages of nodes that the VFC method can not handle. While training the MAT method, we found that if we omit the id information, the performance is significantly worse than the VFC method, while the combination of both the text information and the id information leads to better results.

The results of the MAT method are equal to those of the VFC method when being provided id and passages, but they have a longer training time. The MAT method has completely solved the problem of giving two nodes and each node's information, indicating whether there is a link edge or not between them.

TABLE III: Detailed results for each method.

Method		Public test	Private test
VFC	id	0.99999	1.00000
	passages	-	-
	id + passages	-	-
MAT	id	-	-
	passages	0.99953	0.99950
	id + passages	0.99996	1.00000

TABLE IV: Total training time. *w Init* stands for with initialization and *w/o Init* stands for without Initalization.

Method	number of epochs	total training time
VFC w Init	2	4 mins
VFC w/o Init	5	10 mins
MAT	2	3 hours

When training the VFC method on the dataset, we conducted two distinct scenarios: with and without initialization. Upon the completion of the training process, the results exhibited that utilizing the initializer leads to superior performance on both the public and private test, in comparison to without using the initialization (VFC w/o Init) scenario. Furthermore, the incorporation of initialization during training significantly improved the training time for both the public and private tests, as opposed to training without using initialization. Detailed comparison results can be found in Table IV and Table V.

TABLE V: Results of VFC with/without using initialization.

	Using initialization	Public test	Private test
VFC	✗	0.99996	0.99998
	✓	0.99999	1.00000

V. Conclusion And Future Work

In this paper, we have presented the VFC (Vanilla Fully Connected) method and the MAT (Mutual Attention Transformer) method to tackle the link prediction task in the competition. The results show that our proposed methods have an almost absolute performance and also effectively exploit the information obtained from the dataset.

Because the data pre-processing is not good enough, using only text information in the MAT method will not yield the expected results. But when this method uses only 64 tokens for each piece of passages and incorporates with their id, the performance is still as good as desired. On the other hand, we have limited computing resources, so training time is also a barrier. But in terms of efficiency, the MAT method gives almost perfect results in competition.

In the future, we will apply methods based on graph neural networks for further experiments with link prediction task.

Acknowledgment

This work has been funded by The VNUHCM-University of Information Technology's Scientific Research Support Fund.

References

[1] A. N. Papadopoulos, "Dsaa 2023 competition," 2023. [Online]. Available: https://kaggle.com/competitions/dsaa-2023-competition

[2] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI open*, vol. 1, pp. 57–81, 2020.

[3] J. Chen, X. Wang, and X. Xu, "Gc-lstm: Graph convolution embedded lstm for dynamic network link prediction," *Applied Intelligence*, pp. 1–16, 2022.

[4] S. Yun, M. Jeong, R. Kim, J. Kang, and H. J. Kim, "Graph transformer networks," *Advances in neural information processing systems*, vol. 32, 2019.

[5] Z. Ying, D. Bourgeois, J. You, M. Zitnik, and J. Leskovec, "Gnnexplainer: Generating explanations for graph neural networks," *Advances in neural information processing systems*, vol. 32, 2019.

[6] Z. Chen, J. Xu, C. Alippi, S. X. Ding, Y. Shardt, T. Peng, and C. Yang, "Graph neural network-based fault diagnosis: a review," *arXiv preprint arXiv:2111.08185*, 2021.

[7] A. Sankar, Y. Wu, L. Gou, W. Zhang, and H. Yang, "Dysat: Deep neural representation learning on dynamic graphs via self-attention networks," in *Proceedings of the 13th international conference on web search and data mining*, 2020, pp. 519–527.

[8] L. Datta, "A survey on activation functions and their relation with xavier and he normal initialization," *arXiv preprint arXiv:2004.06632*, 2020.

[9] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," *Advances in neural information processing systems*, vol. 32, 2019.

[10] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Online]. Available: http://arxiv.org/abs/1810.04805