IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 19, 2021, accepted July 8, 2021, date of publication July 13, 2021, date of current version July 19, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3096845

Graph Neural Networks Using Local Descriptions in Attributed Graphs: An Application to Symbol Recognition and Hand Written Character Recognition

NADEEM IQBAL KAJLA[1], MALIK MUHAMMAD SAAD MISSEN[1], MUHAMMAD MUZZAMIL LUQMAN[2], AND MICKAEL COUSTATY[2], (Member, IEEE)

[1]Department of IT, The Islamia University of Bahawalpur, Bahawalpur 63100, Pakistan
[2]Laboratoire Informatique, Image et Interaction (L3i), La Rochelle University, 17000 La Rochelle, France

Corresponding authors: Malik Muhammad Saad Missen (saad.missen@iub.edu.pk) and Mickael Coustaty (mickael.coustaty@univ-lr.fr)

This work was supported in part by the European Regional Development Fund Program: Interreg Atlantic Area Program in the framework of the CircularSeas Project, and in part by the LabCom IDEAS under Grant ANR-18-LCV3-0008.

ABSTRACT Graph-based methods have been widely used by the document image analysis and recognition community, as the different objects and the content in document images is best represented by this powerful structural representation. Designing of novel computation tools for processing these graph-based structural representations has always remained a hot topic of research. Recently, Graph Neural Network (GNN) have been used for solving different problems in the domain of document image analysis and recognition. In this article we take forward the state of the art by presenting a new approach to gather the symbolic and numeric information from the nodes and edges of a graph. We use this information to learn a Graph Neural Network (GNN). The experimentation on the recognition of handwritten letters and graphical symbols shows that the proposed approach is an interesting contribution to the growing set of GNN-based methods for document image analysis and recognition.

INDEX TERMS Graph Neural Networks (GNN), attributed graphs, graph matching, local descriptions, graph similarity, graph learning, graph classification, document image analysis (DIA), pattern recognition (PR).

I. INTRODUCTION

Graphs have become very efficient and adequate field of research over the past few decades and many pattern recognition problems are being solved by graphs particularly in document analysis [1]. Although statistical methods have historically been more computationally efficient in solving problems in document image analysis, the graph-based methods are reaching the benchmarks [2] since past few years. In document analysis graphs can capture the structural information of shape (of characters, symbols and other content) in documents. Graph-based techniques are nowadays commonly used in document analysis not as an alternative to statistical methods but as a complementary. Statistical methods are less complex and need some relatively simple mathematical operations [3] but on the other side, the graphs are able to present both symbolic and structural information which can be more useful in solving pattern recognition problems. Graphs are very efficient in dealing with similarity measurement in pattern recognition which is known as graph matching. It is an important property to measure the similarity or distance between two graphs. This problem basically consists of finding the minimum common subgraph to find if an isomorphism exists between the graphs. [4] proposed a Graph Edit Distance (GED) for graph matching. The main drawback of GED is its computational complexity [5]. GED has exponential time complexity with respect to number of nodes in the graph, which makes it unfeasible to apply GED in real world scenarios. To cope with this drawback lots of efforts have been done so far, for instance [5] have presented an error tolerant inexact graph edit distance. The recent success in Convolutional Neural Networks (CNN) [6] in computer

The associate editor coordinating the review of this manuscript and approving it for publication was Huiyu Zhou.

vision and other fields, has also captured the interest of researchers, focused on using structural pattern recognition, to extend these frameworks to non-Euclidean structures like manifolds and graphs. These extensions in CNNs are commonly called Geometric Deep Learning (GDL) [7]. Among many other methods, Graph Neural Networks (GNN) are becoming more famous and achieving remarkable success. Keeping ourself aligned with the modern trends in the structural pattern recognition research community (for document analysis and recognition), in this paper we present our work on the use of Geometric Deep Learning (GDL) [8] to improve the state of the art in hand written character recognition and symbol recognition. Our proposed approach is an improvement and extension to the work of [3]. We propose to make use of all the symbolic and numeric attributes of a graph for learning for learning the graph characterizations and graph distances with Graph Neural Networks (GNN).

In Section II we present an overview of the state of the art. In Section III our proposed approach is described and in Section IV the implementation details and results are presented. The paper finishes with conclusions and future research directions in Section V.

II. RELATED WORK

In this section first we present the state of the art about Graph Neural Networks (GNNs) and Graph Edit Distance (GED). It will be followed by highlighting the research gap in the literature, the problem statement and our contribution.

A. GRAPH NEURAL NETWORKS (GNN)

Recent success and advancements of Convolutional Neural Networks (CNNs) in different areas of Pattern Recognition (PR), such as natural language processing, image processing and computer vision, has increased the interest of researchers to extend these frameworks to the non-Euclidean structures i.e. manifolds, graphs etc. These advancements in architectures are known as Geometric Deep Learning (GDL) [8]. The emerging field of GDL has opened new directions for researchers and has provided new learning tools when dealing with graphs and manifolds. GDL provides frameworks and tools which enable us to learn characterisations of graphs and provide information about its topology [6]. [9] presented a machine translation algorithm based on Graph Neural Networks(GNN) for Natural Language Processing (NLP).

[10] proposed a method in which Convolution Neural Networks(CNNs) are generalized from low dimensional regular grids, where speech, video and image are represented, to high-dimensional irregular domains, such as brain connections embedding, social networks or words embedding represented by graphs. [11] proposed a Message Passing Neural Network (MPNN). They used weight tying for each time step. They also used an update function for each time step. The Gated Recurrent Unit (GRU) is used which was proposed by [12]. [13] presented a MPNN to perform object reasoning and relation centric by analogous to simulation. An input graph is used in deep Neural Networks to implement this simulation. This approach works at node level and on graphs level as well.

[14] presented a generalization of Convolution Neural Networks (CNN). Their model is based on the hierarchical clustering and on the graph Laplacian spectrum. Results shows that a powerful and efficient deep network can be built by learning the convolutional layers independent from the input size.

[15] proposed a supervised learning architecture for graphs and they named it ''Message Passing Neural Network'' (MPNN). They redefined the previously used spatial and spectral architectures and proposed new framework with two steps, a message passing step and the readout step. A node update function is used to update the hidden state in message passing step. This message is collected from the neighbouring nodes. In message passing step the structural information of graphs are gathered and is embedded as node labels. In the readout step, a feature vector is computed for the complete graph. This feature vector is based on the set of hidden states of the nodes.

B. GRAPH EDIT DISTANCE (GED)

To find the similarity among two graphs, Graph Edit distance is a very efficient method. It is an error tolerant graph matching method but the main drawback of GED is its high computational complexity [2]. GED has exponential time complexity with respect to the number of nodes, this makes it unfeasible to apply this method in real world application. To overcome the issue of time complexity, numerous algorithms have been proposed by the researchers. [5] proposed an approximate graph edit distance which is a bipartite graph matching method and based on the assignment problems solution.

$$D(g_1, g_2) = \min_{(e_1,\ldots,e_k)\in\gamma(g_1,g_2)} \sum_{i=1}^{K} c(e_i) \tag{1}$$

Equation 1 is used to find the distance. It uses cost matrix with edit operations and provide an upper bound of the original GED. Housdorff Edit Distance (HED) by [16] provide the lower bound order of the GED. It is based on the housdorff matching. HED is defined as Equation 2.

$$HED(g_1, g_2, C) = \sum_{u\in V_1} \min_{v\in V_2\cup\{\in\}} C_n^*(u, v) + \sum_{v\in V_2} \min_{u\in V_1\cup\{\in\}} C_n^*(u, v) \tag{2}$$

where $C_n^*(u, v)$ is the cost function and it is defined as in Equation 3.

$$C_n^*(u, v) = \begin{cases} \dfrac{C_n(u, v)}{2} & if\ (u \to v)\ is\ a\ substitution \\ C_n(u, v) & Otherwise \end{cases} \tag{3}$$

[17] provide a brief overview of different GED methods their variations and their results.

[18] proposed an attributed graph distance using Heterogeneous Euclidean Overlap Metric (HEOM) [18] to handle

the numeric and symbolic attributes. These attributes are used to find the distance between two graphs. It considered local descriptions of graph instead of global. The method is based on simple vector representation of graph and the local descriptions are straightforwardly computed from an adjacency matrix.

C. PROBLEM STATEMENT AND OUR CONTRIBUTION

The work in [3], while calculating the similarity, gathers all similarity scores from pair of nodes and for each node it finds a node in another graph with the highest similarity scores. It simply calculates the final result by summing up all the similarity scores of the nodes. In this methods the distance is calculated specifically for the case of Hausdorff Edit Distance (HED) [5] in which all nodes are substituted and there are no insertion or deletion operations. HED embeds the local structure during the message passing phases so the edges are not considered in this scenario. Edges have very important information so if these edges are taken into account then discriminating information can be gathered. In this paper, we emphasize on the importance of the information of the symbolic and numeric attributes of the local descriptions of the graph. This results in a robust model which will perform better even when numeric or symbolic attributes needs a more complex metric. In this paper, we ameliorate the method of [3] to learn the graph characterizations and graph distances with Graph Neural Network (GNN). In the proposed model, a Siamese architecture learns the weights and computes distance by using the same model. This approach collects the local symbolic and numeric attributes of the nodes and edges and utilize that information to calculate the distance between two graphs. The next section will present more details on our proposed improvement and extension to the work of [3].

III. PROPOSED APPROACH

On the basis of the ideas presented by [19] and [3], in this paper we have proposed a spatial based siamese architecture which computes graph distances by calculating the symbolic and numeric information of neighbouring nodes and edges in a graph.

Each node is associated to a vector space, the attributes of nodes and the edge attributes are the components of that particular node. So, algorithm gathers the structural information of the local context of the nodes and the Graph Neural Network (GNN) learns an ameliorated characterization of the original graph. We use the GED to obtain a similarity metric between two graphs. GED can be calculated with different methods. We used Heterogeneous Euclidean Overlap Metric distance (HEOM) [18] because it uses the symbolic and numeric attributes of local descriptors. Figure 1 shows the complete architecture of proposed model and Algorithm 1 presents the pseudo code of the proposed model.

The equation 4 is used to learn the matrix for possible edges label, where M is a function to collect the message and m is a matrix learned for each possible edge label, Algorithm 2

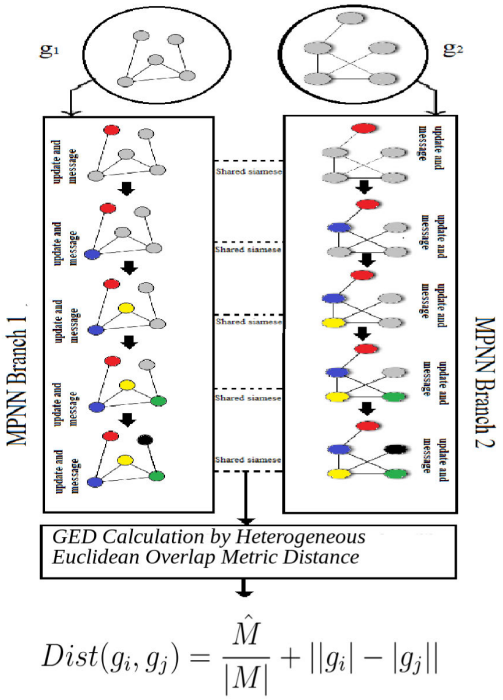

FIGURE 1. Spatial based Siamese Graph Neural Network Architecture.

Algorithm 1 Siamese NN Algorithm

```
PROGRAM trainModel
INPUT g1, g2
FOR t in T time steps for each MPNN network
call: update
call: message
END FOR
call: calculate distance
call: calculate loss
(back propagate loss)
END
```

shows the pseudo code of message function.

$$M(x_v, x_w, i_{vw}) = m_{i_{vw}}, x_w \tag{4}$$

we compute the update function suggested by [12] the equation 5 is for calculation the update function, where GRU is Gated Recurrent Unit [12] and Algorithm 3 is the pseudo code to calculate Update function.

$$U(h_v, M_v) = GRU(h_v, M_v) \tag{5}$$

Readout function is the function which maps the characterizations of a graph into a vector space. The main limitation of this readout function is that it does not consider the individual attributes of edges and nodes of the graph. This drawback is removed in this work by calculating the Heterogeneous Euclidean Overlap Metric (HOEM), originally proposed by [18]. HEOM [18] distance gathers the local information of the symbolic and numeric attributes of the graphs. So, there is correspondence among the nodes

Algorithm 2 Message Function

Message Function
Pass In: x_v, x_w, i_{vw}
$M(x_v, x_w, i_{vw}) = m_{i_{vw}} x_w$
PassOut: M_v
Endfunction

Algorithm 3 Update Function

Update Function
Pass In: h_v, M_v
$U(h_v, M_v) = GRU(h_v, M_v)$
PassOut: *nothing*
Endfunction

of the graphs. It uses symbolic information along with the numeric information. This improves the learning of GNN and results into an efficient and robust model. We compare the results of model using readout phase presented in [11] and the readout function is defined as in equation 6 in which i and j are two neural networks and the $\cdot$ shows the element-wise multiplication.

$$RO = \sum_{v \in V} \delta(i(h_v^{(T)}), h_v^o . (j(h_v^T)) \tag{6}$$

Classically, a distance measurement is required when the similarity between two entities in multidimensional feature space is needs to be determined. For measuring the similarity between the graphs we use graph edit distance method, for this purpose we use heterogeneous euclidean overlap metric approach suggested by [18]. This metric can handle the numeric and symbolic attributes of nodes as well as the edges. While calculating information of node of a graph, we use the complete information into the graph to the related node. These information consists of the node degree, node attributes, the attribute of the incident node. The equation 7 is used to calculate the heterogeneous Euclidean distance between two nodes.

$$HD(g, f) = \sqrt{\sum_{a=0}^{A} \delta(g_a, f_a)^2} \tag{7}$$

where a is an attribute of A and δ is the overlapping normalized range or information, which is calculated by the equation 8. This function helps to handle the missed information. if any information is missed then the overlapping function and normalization can handle it by returning the attribute distance of "1", which is maximal distance.

$$\delta(g_a, f_a) = \begin{cases} 1 & \text{if } g_a \text{ or } f_a \text{ are missing} \\ overlap(g_a, f_a) & \text{if } a \text{ is symbolic} \\ range - dif_a(g_a, f_a) & \text{if } a \text{ is numeric} \end{cases} \tag{8}$$

We calculate the value of range difference with equation 9, this is used to scale the attribute to the point where the difference is less than "1".

$$range - dif_a(g_a, f_a) = \frac{|g_a, f_a|}{range_a} \tag{9}$$

To compute the distance between every two nodes in a graph the above mentioned equations are used. These distances define a cost function. This cost function is used for node to node assignment for different graphs. We calculate permutation of graph using Hungarian method [20]. So finally we calculate the distance between two graphs with formula in equation 10.

$$Dist(g_i, g_j) = \frac{\hat{M}}{|M|} + \big||g_i| - |g_j|\big| \tag{10}$$

where $|M|$ is the number of matching operations, and $\hat{M}$ is the matching cost and it is calculated by taking the sum of all matching operations costs. So, while calculating the distance we calculate the information about numeric and symbolic attributes of local nodes and edges. The distance represent the matching cost which is normalized by the matching size and is increased by the difference sizes of graphs. Figure 2 shows the flowchart of proposed Spatial based Siamese Neural Network.

Edges have very important information so if these edges are taken into account then very distinctive information can be gathered. In our approach we emphasis on the importance of the information in the symbolic and numeric attributes of the local descriptions of the graph, so it is less affected by the distortions in the graphs.

IV. EXPERIMENTATION

In this section we will first present the details about the datasets and our experimentation setup. This will be followed by the obtained results and a discussion.

A. DATASETS

For the evaluation of proposed approach, We have used two datasets, originally presented by [21]. These datasets are widely used by the research community of graph-based methods for document analysis and recognition. These have become a defacto standard for evaluating and becnhmarking the graph-based methods for document image analysis and recognition. We have used *letters* dataset for hand written character recognition and *GREC* for the symbol recognition.

1) LETTERS

Letters dataset consists on 15 classes. Each class represents a capital letter. The dataset consists of A, E, F, H, I, K, L, M, N, T, V, W, X, Y, Z letters drawn manually. These handwritten letters are converted into graphs in which lines are represented by edges and the ending points of these lines are presented by nodes. Two dimensional attributes are labelled to position the nodes. 6750 graphs representations are divided into three categories each having 2250 total graphs. These 2250 graphs are then divided into test, train and validation and each set contain 750 graphs. The state of the art results

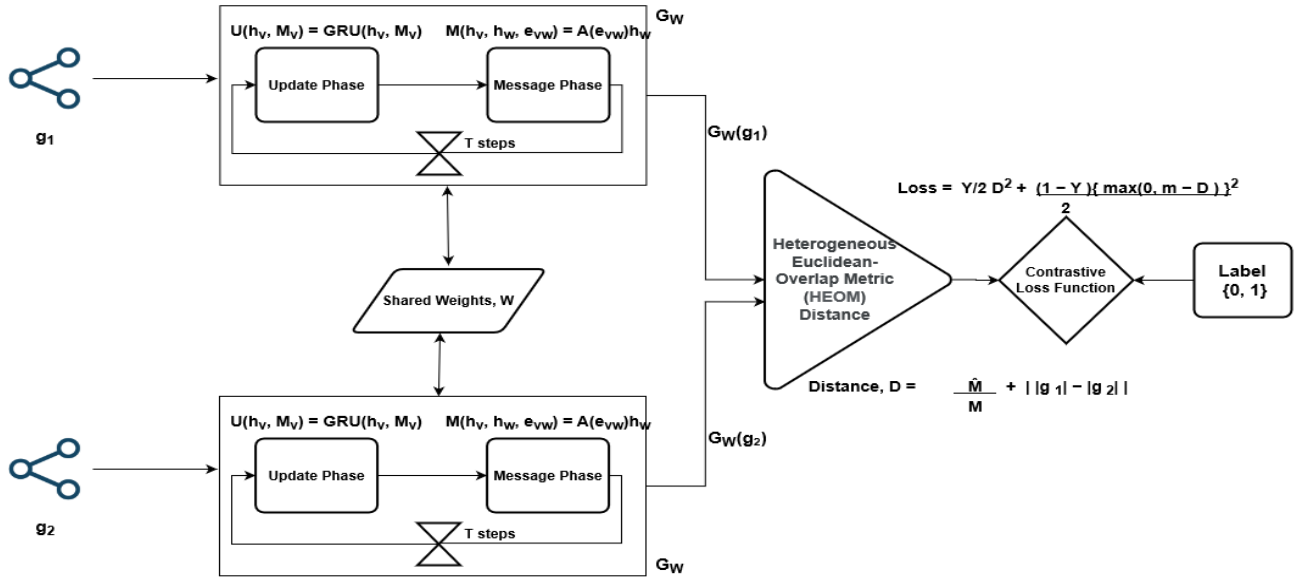

FIGURE 2. Flow Chart of proposed Siamese Graph Neural Network.

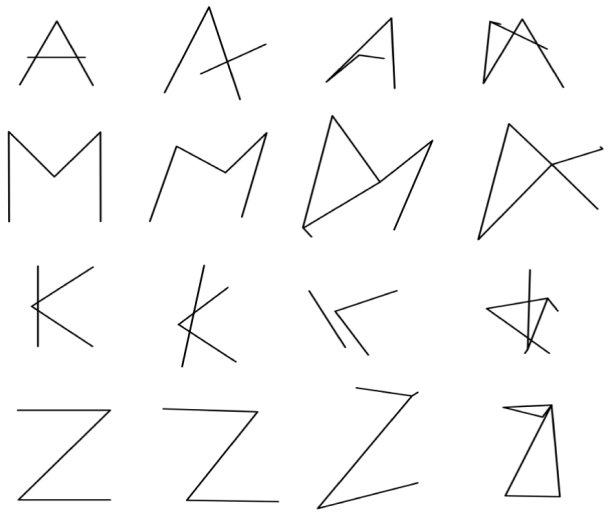

FIGURE 3. Some examples of letters with different level of distortions(low/medium/high from left to right) from Letters Dataset [21].

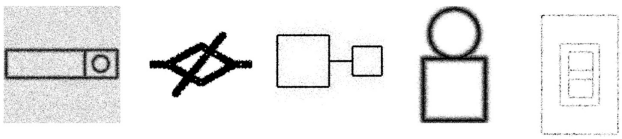

FIGURE 4. Some images from GREC Dataset of IAM repoistory [21].

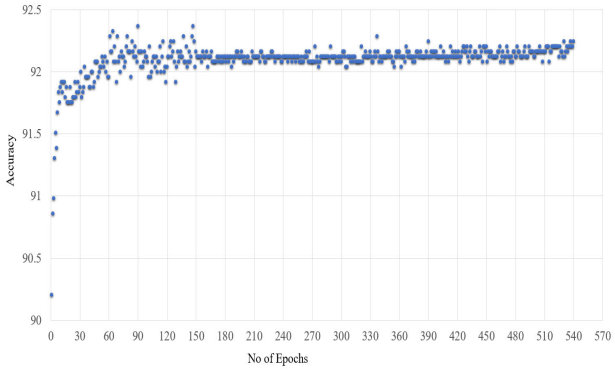

FIGURE 5. The Validation accuracy of Letters LOW Dataset.

on this dataset are slightly less than 100% which shows the dataset is very simple and easy. The dataset consists of three categories LOW, MED and HIGH on the basis of no of nodes and edges and their distortion level. (as shown in Figure 3).

2) GREC

GREC is famous dataset of IAM Graph repoistory [21] used in document analysis. It has twenty-two classes in total, each class has fifty examples, hence there are 1100 graphs in total. This dataset consists of images representing symbols from electronic, architecture and some other technical fields as well. Graphs are constructed in symbols by detecting the corners and intersections, these ending lines, corners and intersections are represented by nodes and edges.

In the experiments 286 graphs are used for training, 286 graphs are used for the validation purpose and 528 graphs are used for the testing. As we emphasize on the local descriptions of symbolic and numeric attributes of graphs and the GREC dataset has more numeric and symbolic attribute information comparing to the letters dataset.

B. IMPLEMENTATION

The proposed model is trained in an unsupervised training manner to check which particular pair of nodes belongs to which class. For training of the model we used 1000 epochs. Model is trained with Stochastic Gradient Descent optimiser

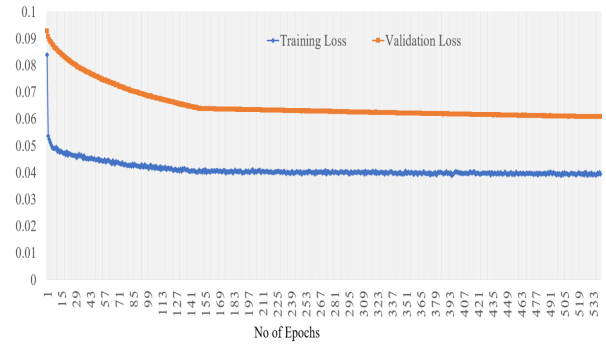

FIGURE 6. The "Validation vs Loss" of Letters LOW Dataset.

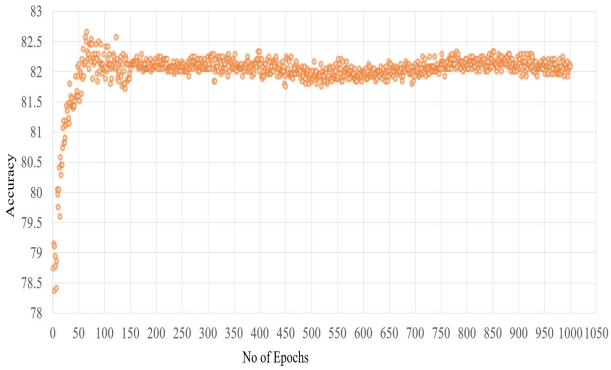

FIGURE 7. The Validation accuracy of Letters MED Dataset.

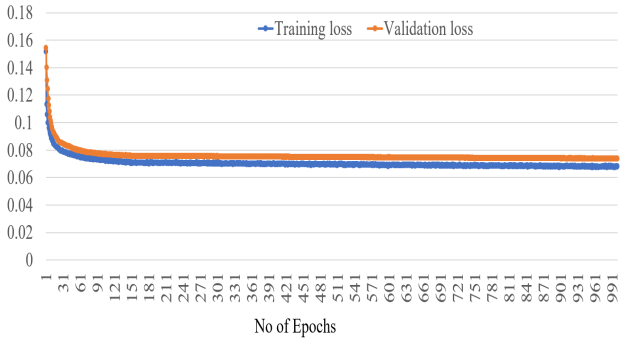

FIGURE 8. The "Validation vs Loss" of Letters MED Dataset.

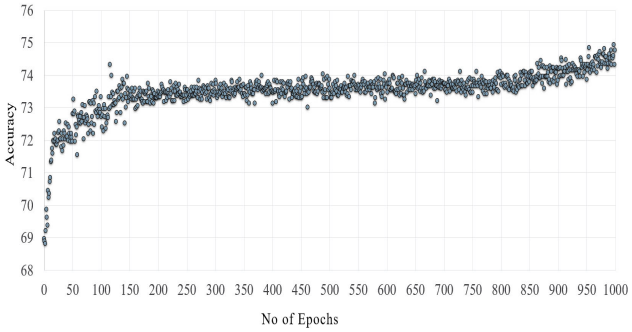

FIGURE 9. The Validation accuracy of Letters HIGH Dataset.

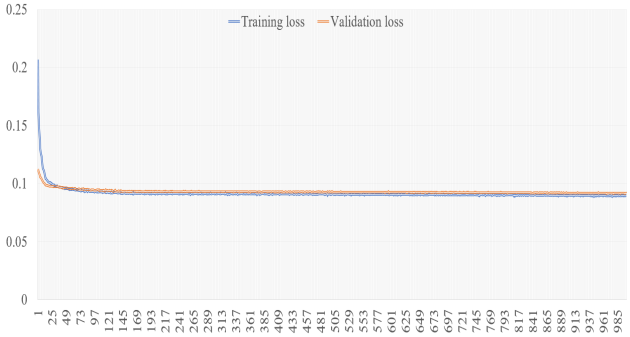

FIGURE 10. The "Validation vs Loss" of Letters HIGH Dataset.

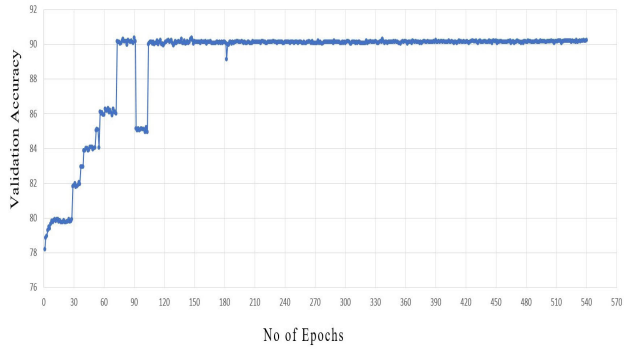

FIGURE 11. The Validation accuracy of GREC Dataset.

to avoid over fitting. Weight decay and momentum is also used for optimization so that model cannot stuck in local minima. K-nearest neighbor classifier is used for classifications of graph distances. ''m'' is neural network having edge feature of 64×64 matrix. Four input layers are used by neural network ''m'' and the ReLU activation function is used. For Gated Recurrent Unit (GRU) network [11] hidden state of size 64 is used by the update function. The hyper-parameter learning rate is set to 30%. For the optimization of results we used the value of momentum to 0.9 and value of decay parameter is set to 0.0005. Learning rate is decreased after every 150 iterations. Gamma hyper-parameter is used and it is multiplied with the learning rate after every 150 iterations. All experiments are done on the attributed graphs.

C. RESULTS AND DISCUSSION

Our model performed good in all three categories and more significant results on Letters HIGH, which has even more distortions as shown in Table 1. The reason behind this is that our model focuses on the symbolic and numeric attributes on the local descriptions and the Letters HIGH dataset consists of more edges and nodes (because of unrealistically highly distorted images) as compare to the Letters LOW and Letters MED datasets. Figures 5, 7, 9 shows the Validation Accuracy of Letters LOW, MED and HIGH. The training vs validation loss of LOW, MED and HIGH categories of Letters dataset

TABLE 1. Accuracy (percentage) of Different Methods on Letters and GREC, as reported by authors in respective publications.

Technique	Method	Year	LOW	MED	HIGH	GREC
Graph Edit Distance	IAM Graph repo. for PR and ML [21]	2008	99.60	94.00	90.00	95.50
	BP [5]	2009	–	–	–	86.30
	Bipartite Graph Matching [5]	2009	**99.73**	94.27	89.87	–
	Attributed GM [18]	2009	–	–	–	98.10
	Vector representations of graphs [22]	2009	91.86	–	–	95.83
	Bipartite-H [23]	2011	99.60	94.20	89.80	68.00
	Bipartite-V [23]	2011	99.60	94.30	89.90	67.00
	Hausdorff Edit Distance [16]	2015	97.87	86.93	79.20	–
Graph Embedding	ELG into occurrence matrix [24]	2009	92.53	–	–	97.53
	Dimensionality reduction PCA with SVM [25]	2009	92.70	81.10	73.30	91.80
	Dimensionality reduction MDA with SVM [25]	2009	89.80	68.50	60.50	91.80
	LGQ [26]	2010	81.50	–	–	86.20
	Maximum likelihood bayes [27]	2011	81.20	–	–	89.90
	Dissimilarity BE [28]	2011	99.30	94.90	92.90	92.40
	Discriminative prototype for GE [29]	2013	99.50	**95.40**	**93.40**	92.50
	FMGE with SVM [30]	2013	98.20	83.10	70.00	99.20
	FMGE with kNN [30]	2013	97.10	75.70	66.50	97.50
	GRALGv2 [31]	2014	97.60	89.60	82.60	97.60
	GRALGv1 [31]	2014	98.20	79.80	74.50	97.70
	PSGE [32]	2017	–	–	–	**99.81**
	SGE [33]	2017	–	–	–	99.62
Graph Neural Networks	MPNN [3]	2018	95.04	83.20	72.27	–
	Siamese MPNN [3]	2018	98.08	89.01	74.77	–
	Proposed Method: GNN using local descriptions	–	96.64, ±0.759	85.27, ±0.915	79.91,±1.103	96.93,±2.117

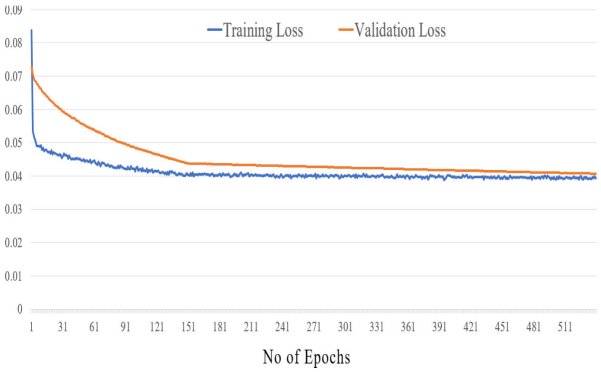

FIGURE 12. The "Validation vs Loss" of GREC Dataset.

is shown in Figure 6, 8, 10. Validation loss vs training loss in Figure 10 of letters HIGH dataset shows that the proposed model perfectly fit on letters HIGH dataset because it consists of more edges and nodes and having more symbolic and numeric information as compare to the MED and LOW. The validation accuracy of GREC dataset is shown in Figure 11 and Validation loss vs training loss of GREC dataset is shown in the Figure 12.

In our method we focus on the local information of graph. In the work [3] while calculating the similarity it gathers all similarity scores from pair of nodes and for each node it finds a node in another graph with the highest similarity scores. It simply calculates the final result by summing up all the similarity scores of the nodes. In this methods [3] the distance is calculated specifically for the case of Hausdorff edit distance [5] in which all nodes are substituted and there are no insertion or deletion operations. HED embeds the local structure during the message passing phases so the edges are not considered in this scenario. Edges have very important information so we consider these edges while calculating the structural information and these provide very discriminating information. In our approach we calculate the distance at the node level and at the graph level using heterogeneous Euclidean overlap metric [18]. we emphasis on the importance of the information in the symbolic and numeric attributes of the local descriptions of the graph, this results in a robust model which perform better even when numeric or symbolic attributes needs more complex metric. As shown in Table 1 Our results are significant on both dataset specially on HIGH because it has more symbolic and numeric attributes as compared to the LOW and MED. This shows that our method will be even more suitable for the complex graphs which have symbolic information both on its nodes and edges. The time complexity of the proposed method has among the state of the art algorithms for computing approximate graph edit distances. It computes the graph similarity on the basis of node level and edge level information. The time complexity of our approach is $O(DN^2)$, where N is denoted by the number of nodes in a graph.Computational complexity of our methods is not very expensive. The wall time for computing the similarity computation between the graphs is 33.12 sec for GREC dataset, 27.81 sec is for LETTER High dataset, 24.55 sec for LETTER MED and 19.31 sec for the LETTER Low dataset.

V. CONCLUSION

In this paper we have presented a GNN to learn the graph characterizations and graph distances using the local descriptions of a graph. We also compare some state of the art results of GED and graph embedding with GNN based approaches. The proposed model shows significant results on both datasets *i.e. Letters and GREC*; showing that our model performs more accurate when nodes and edges contain the symbolic and numeric attributes. In the light of these results, we conclude that information that we extract from the graph nodes and edges provide good structural local characterizations of graphs. The extracted information becomes

more discriminating when both nodes and edges are taken into account and their symbolic and numeric attributes are used. An ongoing future direction of our work is to produce a dataset with more meaningful attributes on nodes and edges, so that full potential of GNN-based methods for document image analysis and recognition could be exploited.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

REFERENCES

[1] D. Conte, P. Foggia, C. Sansone, and M. Vento, "Thirty years of graph matching in pattern recognition," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 18, no. 3, pp. 265–298, 2004.
[2] P. Foggia, G. Percannella, and M. Vento, "Graph matching and learning in pattern recognition in the last 10 years," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 28, no. 1, 2014, Art. no. 1450001.
[3] P. Riba, A. Fischer, J. Lladós, and A. Fornés, "Learning graph distances with message passing neural networks," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 2239–2244.
[4] H. Bunke, "On a relation between graph edit distance and maximum common subgraph," *Pattern Recognit. Lett.*, vol. 18, no. 8, pp. 689–694, 1997.
[5] K. Riesen and H. Bunke, "Approximate graph edit distance computation by means of bipartite graph matching," *Image Vis. Comput.*, vol. 27, no. 7, pp. 950–959, 2009.
[6] F. Monti, D. Boscaini, J. Masci, E. Rodola, J. Svoboda, and M. M. Bronstein, "Geometric deep learning on graphs and manifolds using mixture model CNNs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5115–5124.
[7] K. T. Schütt, F. Arbabzadah, S. Chmiela, K.-R. Müller, and A. Tkatchenko, "Quantum-chemical insights from deep tensor neural networks," *Nature Commun.*, vol. 8, no. 4, p. 13890, 2017.
[8] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: Going beyond Euclidean data," *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 18–42, Jul. 2017.
[9] Y. Wu *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," 2016, *arXiv:1609.08144*. [Online]. Available: http://arxiv.org/abs/1609.08144
[10] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3844–3852.
[11] Y. Li, D. Tarlow, M. Brockschmidt, and R. Zemel, "Gated graph sequence neural networks," 2015, *arXiv:1511.05493*. [Online]. Available: http://arxiv.org/abs/1511.05493
[12] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," 2014, *arXiv:1409.1259*. [Online]. Available: http://arxiv.org/abs/1409.1259
[13] P. W. Battaglia, R. Pascanu, M. Lai, D. Rezende, and K. Kavukcuoglu, "Interaction networks for learning about objects, relations and physics," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4502–4510.
[14] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," 2013, *arXiv:1312.6203*. [Online]. Available: http://arxiv.org/abs/1312.6203
[15] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 1263–1272.
[16] A. Fischer, C. Y. Suen, V. Frinken, K. Riesen, and H. Bunke, "Approximation of graph edit distance based on Hausdorff matching," *Pattern Recognit.*, vol. 48, no. 2, pp. 331–343, 2015.
[17] Z. Abu-Aisheh, B. Gaüzère, S. Bougleux, J.-Y. Ramel, L. Brun, R. Raveaux, P. Héroux, and S. Adam, "Graph edit distance contest: Results and future challenges," *Pattern Recognit. Lett.*, vol. 100, pp. 96–103, Dec. 2017.
[18] S. Jouili, I. Mili, and S. Tabbone, "Attributed graph matching using local descriptions," in *Proc. Int. Conf. Adv. Concepts Intell. Vis. Syst.* Berlin, Germany: Springer, 2009, pp. 89–99.
[19] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a 'Siamese' time delay neural network," in *Proc. Adv. Neural Inf. Process. Syst.*, 1994, pp. 737–744.
[20] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Res. Logistics Quart.*, vol. 2, nos. 1–2, pp. 83–97, 1955.
[21] K. Riesen and H. Bunke, "IAM graph database repository for graph based pattern recognition and machine learning," in *Joint IAPR Int. Workshops Stat. Techn. Pattern Recognit. (SPR) Struct. Syntactic Pattern Recognit. (SSPR)*. Berlin, Germany: Springer, 2008, pp. 287–297.
[22] N. Sidère, P. Héroux, and J.-Y. Ramel, "Vector representation of graphs: Application to the classification of symbols and letters," in *Proc. 10th Int. Conf. Document Anal. Recognit.*, Jul. 2009, pp. 681–685.
[23] S. Fankhauser, K. Riesen, and H. Bunke, "Speeding up graph edit distance computation through fast bipartite matching," in *Proc. Int. Workshop Graph-Based Represent. Pattern Recognit.* Berlin, Germany: Springer, 2011, pp. 102–111.
[24] N. Sidère, P. Héroux, and J.-Y. Ramel, "Embedding labeled graphs into occurence matrix," in *Proc. IAPR Workshop Graph. Recognit.*, 2009, pp. 44–50.
[25] K. Riesen and H. Bunke, "Reducing the dimensionality of dissimilarity space embedding graph kernels," *Eng. Appl. Artif. Intell.*, vol. 22, no. 1, pp. 48–56, Feb. 2009.
[26] B. J. Jain, S. D. Srinivasan, A. Tissen, and K. Obermayer, "Learning graph quantization," in *Proc. Joint IAPR Int. Workshops Stat. Techn. Pattern Recognit. (SPR) Struct. Syntactic Pattern Recognit. (SSPR)*. Berlin, Germany: Springer, 2010, pp. 109–118.
[27] B. J. Jain and K. Obermayer, "Maximum likelihood for Gaussians on graphs," in *Proc. Int. Workshop Graph-Based Represent. Pattern Recognit.* Berlin, Germany: Springer, 2011, pp. 62–71.
[28] H. Bunke and K. Riesen, "Recent advances in graph-based pattern recognition with applications in document analysis," *Pattern Recognit.*, vol. 44, no. 5, pp. 1057–1067, May 2011.
[29] E. Z. Borzeshi, M. Piccardi, K. Riesen, and H. Bunke, "Discriminative prototype selection methods for graph embedding," *Pattern Recognit.*, vol. 46, no. 6, pp. 1648–1657, 2013.
[30] M. M. Luqman, J.-Y. Ramel, J. Lladós, and T. Brouard, "Fuzzy multilevel graph embedding," *Pattern Recognit.*, vol. 46, no. 2, pp. 551–565, 2013.
[31] F. M. Bianchi, L. Livi, A. Rizzi, and A. Sadeghian, "A granular computing approach to the design of optimized graph classification systems," *Soft Comput.*, vol. 18, no. 2, pp. 393–412, Feb. 2014.
[32] A. Dutta, P. Riba, J. Lladós, and A. Fornés, "Pyramidal stochastic graphlet embedding for document pattern classification," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Nov. 2017, pp. 33–38.
[33] A. Dutta and H. Sahbi, "High order stochastic graphlet embedding for graph-based pattern recognition," 2017, *arXiv:1702.00156*. [Online]. Available: http://arxiv.org/abs/1702.00156

NADEEM IQBAL KAJLA received the master's degree in computer science, in 2016. He is currently pursuing the Ph.D. degree with The Islamia University of Bahawalpur. His research interests include pattern recognition, graph-based methods in pattern recognition, graph matching techniques, graph learning, and graph neural networks.

MALIK MUHAMMAD SAAD MISSEN received the master's and Ph.D. degrees from the University of Toulouse, France, in 2007 and 2011, respectively. He has also completed his postdoctoral training from the University of La Rochelle, France, in 2015. He is currently working as an Assistant Professor with The Islamia University of Bahawalpur, Pakistan. His main research interests include information retrieval, human–computer interaction, and document image analysis.

MUHAMMAD MUZZAMIL LUQMAN is currently a Research Scientist in document image analysis, pattern recognition, and computer vision. He has been working on the post of Research Engineer (Permanent) with the L3i Laboratory, University of La Rochelle, France, since November 2015. From 2012 to 2015, he was a Postdoctoral Researcher, with Prof. Jean-Marc Ogier, at the L3i Laboratory. In 2012, he received the Ph.D. degree in computer science from the François Rabelais University of Tours, France, and the Autonoma University of Barcelona, Spain. His Ph.D. thesis was co-supervised by Prof. Jean-Yves Ramel and Prof. Josep Llados, and was entitled Fuzzy Multilevel Graph Embedding for Recognition, Indexing and Retrieval of Graphic Document Images. He received the master's degree (research) in informatics from the Computer Science Department, Ecole Polytechnique of François Rabelais University of Tours, in 2008. In 2007, he was awarded the Scholarship by the Higher Education Commission, Government of Pakistan; for pursuing the master's degree and the Ph.D. research in France. In 2004, he was awarded the Gold Medal and the Academic Roll of Honor by Government College University Lahore, Pakistan, for achieving distinction in Bachelors of Computer Science (Hons.).

MICKAEL COUSTATY (Member, IEEE) is currently an Associate Professor with the Laboratoire Informatique, Image et Interaction, La Rochelle Université, France. His research interests include document analysis based on computer vision and natural language processing, digital trust in information exchange with the development of fraud detection techniques and document securing systems, segmentation and classification of large document flow, and pattern recognition.