Efficient Graph Classification in Shifted Datasets Using Weighted Correlated Feature Selection

Md. Samiullah[1], Chowdhury Farhan Ahmed[2], Anna Fariha[1], Akiz Uddin Ahmed[1]

[1] Department of Computer Science and Engineering
University of Dhaka, Bangladesh

[2] ICube Laboratory, University of Strasbourg, France
samiullah@cse.univdhaka.edu, cfahmed@unistra.fr, anna@cse.univdhaka.edu, shawpan.du@gmail.com

Abstract. The field of dataset shift has become one of the most interesting field in machine learning and data mining. Real-life applications involve shift of dataset properties where transferring knowledge among domains greatly help in making intelligent decisions. Moreover, graphs effectively model real life scenarios, hence, its classification is of great interest; which requires some informative graph segments to encode label information. In this paper, we propose a new approach of graph classification where distribution of training and operating dataset are different. Unlike existing graph classification works, those classify graphs by treating frequent graphs as significant features in transfer learning scenario, our proposed approach constructs a classifier model based on Correlated Diversified Discriminative Feature set that are non-overlapping, non-redundant and more effective for classification compared to the frequent features as selected by existing algorithms. Also a neural network model is constructed to train the classifier in new environment, which assigns weights to the features for better classification in transferred environment.

Keywords: Graph Classification, Dataset Shift, Transfer Learning, Correlated Feature Selection, Neural Network, Backpropagation

1 Introduction

Classification, nowadays, has become a crucial data mining task to acquire knowledge from huge amount of data to make a meaningful and effective decision. Due to similarity among homogeneous class objects, classification compact the representation of a problem and knowledge extraction process. Graph data structure can capture and model the correlation among entities most effectively and has been successively used in a wide range of real life application domains, such as social and cyber group monitoring in cyber world, mining correlation among chemical compounds and atomic structure in chemistry, fraud, criminal and anomaly detection in various sensitive secured systems. For predicting the behavior, class or underlying complex structures of unknown objects, an automated classification model construction is an interesting problem in recent studies.

The overall classification task is subdivided into discriminative feature selection and classification in almost all the recent works. However, extracting discriminative graph patterns from large number of graph features is a key issue. The key task in such classification process is feature selection. For higher accuracy of classification, discriminative feature selection from the set of frequent sub-graphs is performed in existing approaches e.g. COM [1] and D&D [2] using various scores. Some of the very popular discriminative feature selection scores are *Log Ratio* [1], edge-cover probability [2] and *Fisher-score* [3].

The above stated classification models deal with predicting labels for an unseen graph with the underlying assumption that the domain and distribution of training and operating (where the classifier is used) dataset are same. However, such assumption is unrealistic in most of the real life scenarios due to real-life applications involve some sorts of shift/transfer of dataset properties and/or distribution. In such scenarios knowledge, gathered from one domain, transferred to different domain would greatly improve the performance of the system in making intelligent decisions.

The process of leveraging the information from a source domain to construct and train a classifier for a target domain can be referred to as Transfer Learning. It is one of the most attractive fields of machine learning in recent years which allows sharing supervision knowledge among various domains of different attribute and/or distribution. The data used for training a machine learning model can be dissimilar to the data used for predicting their class labels. Typically there are plenty of labeled examples in the source domain, whereas very few or no labeled examples in the target domain. As example, disease diagnosis in medical science, drug synthesis in pharmacology and population attribute detection in geography.

Transfer learning for graph dataset faces some difficulties [4] and has several issues. Some authors proposed graph based transfer learning capable algorithms, e.g. Jingrui et al. [5] propagated label information to a different domain using a graph based framework; however, they didn't contribute for graph classification except Shi et al. [4], proposed a semi-supervised method to extract significant (frequent) subgraphs for graph classification in transfer learning. However, a good number of studies have been performed that exhibits higher classification accuracy by discriminative and/or diversified feature selection rather than only using frequent features e.g. $D\&D$[2] in traditional domain. Moreover, the classifiers constructed using frequent subgraphs as features face several strains including large number of overlapping and redundant features. Hence we focus on efficient selection of effective significant subgraphs and transferring the features to new domain by appropriate pre-polishing.

Neural networks (NN) are widely used for training systems and presenting knowledge in effective ways, and are applied in various fields. As instance, in a social network with several groups, where the user behavior analysis can be performed by modeling the social network with persons as nodes and their interactions as edges. For predicting effective user behaviors in a group of the network, the interactions of users can be analyzed. The analytical knowledge earned from a group can be directly used for predicting user behavior of the other, if data distribution and the domain of knowledge of the groups are similar. However, in real life scenarios this can be found hardly and requires extensive computation to adapt the knowledge of one group to fit into other.

A typical NN can be used to leverage the anomaly among the dataset and domain characteristics and assist in analysing and predicting future data. The system adjusts itself according to input and output patterns. The processing elements (Cells), can be either input, intermediate or output associated with weights W_{ij}, which determines the effect of cell X_j on cell X_i. Each cell X_i, except input cells, calculates its output, X_i, by applying an activation function, f, on the weighted sum S_i, of its incoming edges.

The Backpropagation (*BP*) is a commonly used method for training a neural network. Using *BP*, *NN* learns by iteratively processing a set of training tuples and comparing the network's prediction for each tuple with the actual known target. For each training tuple, the weights are modified in order to minimize the mean squared error between the prediction and the actual target value. These modifications are done in the backward direction.

These facts motivated us to propose an approach for constructing an efficient graph classifier that can be used to share the supervision knowledge acquired from one domain/dataset to different dataset with varying data distribution property. Pivotal contributions of the paper are: 1. A new diversity computation measure is proposed to extract **C**orrelated **D**iversified **D**iscriminative **F**eatures (*CDDF*) for constructing effective classifier. 2. A classification algorithm is designed to classify graphs among two classes. 3. A backpropagation based learning algorithm is proposed to adjust weights of features in new environments for effective classification in case of data shift.

The rest of the manuscript is organized as follows; The targeted problem, effective feature extraction, proposed measures for feature score calculation and feature adjustment for classification over multiple contexts are presented in Section 2. Finally, Section 3 concludes the work with future development plan.

2 Our Proposed Approach

For efficient classification, we have considered correlation among entities and the diversity as well as discriminative property of features. Some graph correlation calculation metrics are $Jaccard\,Coefficient$ [3], *Log Ratio* [1] and $gConfidence$ [6]. In *D&D* [2], authors considered and calculated correlation between two features using Jaccard Coefficient [3], which leads them to achieve high classifier accuracy. Particularly, correlation among entities are important property observed in various data mining tasks including classification. Hence capturing correlations among entities, while classification, will lead to more accurate and effective classification. As a consequence, while selecting discriminative features, correlation among features along with diversity is considered to select features with highly correlated entities and high diversity with respect to already selected features.

In this manuscript, we have concentrated on classifying undirected labeled graphs into two classes. Minor modifications will enable the approach to tackle other variations. For graph dataset $GD = \{G_1, G_2, ..., G_N\}$; where $G_i = \{V(G_i), E(G_i)\}$ with $V(G_i)$ and $E(G_i)$ are the set of vertices and edges, respectively; support and confidence are two metrices those indicate frequentness and association of graph patterns, respectively.

2.1 Correlated Diversified Feature Capturing Score

Frequent subgraphs, used alone in some existing approaches for classification, are not discriminative enough and not equally important for constructing classifiers. Some authors proposed various single feature discriminating score which could not cope overlapping tendency of frequent features. Authors in *D&D* [2] proposed a new direction for gaining higher accuracy in classification of graphs by considering the diversity as well as discrimination power of features. However, we have proposed a new direction of discriminative feature selection by a new measure considering correlation as well as feature diversity. The measure is defined as follows.

Definition 1 ***(Graph Distance Measure: gDistance)*** *Let* $G_1 = \{V(G_1), E(G_1)\}$ *and* $G_2 = \{V(G_2), E(G_2)\}$ *are two graphs. The Graph Distance measure gDistance between* G_1 *and* G_2 *is the specialized sum of costs of over all operations required to transform* G_2 *into* G_1 *or vice versa. Hence, it can be denoted and defined as*

$$gDistance(G_1, G_2) = \sum_{i=1}^{5} a_i \times cf_i \tag{1}$$

where, a_i represents a value for addition of (1) vertices, (2) edges, deletions of (3) vertices, (4) edges and (5) reversing edge directions to convert G_2 into G_1 and cf_i is the cost factor (weight of operations). In particular, the cost factor of operations required to transform graphs varies in different scenarios based on the correlation and interactions among objects/entities in a dataset. For brevity, in the proposed approach, operation cost factors are considered similar, that is, $cf_1 = cf_2 = cf_3 = cf_4 = cf_5 = 1$. It can be varied to adjust with the different scenarios as per the requirement.

Definition 2 ***(Diversity Measure)*** *For any graph* G *and* $\forall_{sG}, sG \subset G$*, the proposed diversity capturing measure is the minimum difference among the differences between its all possible subgraphs and can be defined as,*

$$Diversity(G) = \min_{\forall sG, sG \subset G} gDistance(G, sG) \tag{2}$$

One of our major objectives is to construct a classifier model that achieves higher accuracy than the existing graph classification approaches, which requires some discriminative features and the diversity of the features accelerates the accuracy of classification. However, the correlation among entities should be brought into consideration along with the diversity of the features. Hence, we have proposed a score named $CDDF$ score (**C**orrelated **D**iversified **D**iscriminative **F**eature) score for classification which considers both correlation and diversity of features.

Definition 3 ***(Correlated Diversified Discriminative Feature Score)*** *Let $GD = \{G_1, G_2, ..., G_N\}$ is graph dataset and $G \subseteq G_i \in GD$ be a frequent feature, i.e. $supp(G) \geq \sigma$, where σ is user specified minimum support threshold. $gConfidence(G)$ is the correlation value of the entities and $diversity(G)$ is the diversity score of the feature G. Hence, for the weight $w \in [0, 1]$, the score can be denoted and defined as,*

$$CDDF(G) = w \times gConfidence(G) + (1 - w) \times diversity(G) \tag{3}$$

$$Score_{CDDF}(G) = \log \frac{CDDF^{+}(G)}{CDDF^{-}(G)} \tag{4}$$

where, $CDDF^{+}(G)$ is the weighted correlated diversity of G for D^{+} (set of graphs labeled positively) and $CDDF^{-}(G)$ is the weighted correlated diversity of G for D^{-} (negatively labeled graph set).

The proposed *CDDF* selection approach captures the correlation and diversity among the entities involved in the n^{th} feature with the already selected n-1 features. Moreover, it considers the weighted average of the two important factors, that is, diversity and correlation, where the weight can be adjusted to fine tune the performance of the classifier. The varying weights enable the measure to cope up with scenarios where significance of correlation and diversity varies.

The problem in classifying a set of graphs and extending the concept to realize the classification in transfer learning scenario can be defined as follows:

Weighted CDDF Formation for Shifted Dataset Classification: *Let $GD_s = \{G_1,\ G_2,\ ...,\ G_N\}$ and $GD_t = \{G'_1,\ G'_2,\ ...,\ G'_M\}$ be set of N and M number of source and target graphs, respectively. Let, $GD_s = D_s^{+} \bigcup D_s^{-}$ with the constraint $D_s^{+} \bigcap D_s^{-} = \emptyset$; and label$(G_i)$ = '+', label(G_j) = '-', where $G_i \in D_s^{+}$ and $G_j \in D_s^{-}$, respectively. Let, $F = \{f_1, f_2, ..., f_k\}$ be a set of top-k CDDF extracted from source datasets for the given threshold σ i.e. $f_m \subseteq G_i \in GD_s$, $supp(f_m) \geq \sigma$. Now the research problem, aiming at determining whether features in F are useful or not, can be described as follows: How to utilize the feature set F to predict the class labels of the incoming unseen graphs sharing same distribution of the target dataset GD_t with the setting that first l ($l \ll M$) graphs within GD_t are labeled by $\{y_1,\ y_2,\ ...,\ y_l\}$, where y_i = '+'/'-' denotes the class label assigned to G'_i.*

2.2 Feature Extraction from Source Dataset

The proposed algorithm captures (1) graph dissimilarity among different frequent features by using proposed measure *gDistance*; and (2) correlation among the entities of the features to select a set of features from the given training dataset. Consequently, the algorithm selects a group of top-k features as positive feature set having larger discriminative score in positive graph set D^+ compared to negative graph set D^- and similarly top-k negative features are selected.

The algorithm takes two datasets, positive dataset D^+ and negative dataset D^-, as training dataset; minimum support threshold σ for frequent feature mining; the weight value w for varying the impact of correlation and/or diversity in calculating the feature selection score, and a value K, for selecting top-K discriminative features from both type of dataset (positive and negative). Then it mines frequent 1-length features, calculates correlation, diversity and feature score with respect to both datasets, D^+ and D^-. Next it extends the frequent features by growing it with an edge and performs the computation as before.

Continuing in this way, the algorithm finds all of its frequent features in the tree structure in association of calculated values of corresponding nodes. Then it selects top-k positive and top-k negative features. These two feature set can be treated as the classifier model constructed for target dataset.

2.3 Feature Adjustment for Target Dataset

This section is assembled with approaches to adjust features extracted from source dataset to construct a classifier for target dataset classification i.e. weighted feature construction and neural network model developing for compensating the changes in data distribution using neural network.

Weighted Correlated Diversified Discriminative Features: Section 2.2 dealt with extraction of two set of features for predicting class label of any unseen graph in same domain. However, the dataset properties are different in target domain and some source domain features may be unnecessary, some may be overlapping and some of them can be smaller/larger in cardinality than required. Our approach to the problem is to assign weights to the features and treat the smaller amount of target domain data with associated class label as input training data for the *NN* to adjust features weights using *BP* algorithm according to the new data distribution. Hence, the *CDDF* score of Eqn. 4 for source domain will be **W**eighted **C**orrelated **D**iversified **D**iscriminative **F**eature (*WCDDF*) score in the target domain as follows:

$$WCDDF(f_i) = W_{f_i} \times Score_{CDDF} \tag{5}$$

where, f_i is any *CDDF* with score in source domain as $Score_{CDDF}$ and with an adjusted weight W_{f_i} for effective classification of unseen target graphs.

Neural Network Model, NN: The recent vast research activities established *NN* as a promising alternative to various conventional classification methods. The advantages of this tool lies on its self adaptive nature, arbitrary function approximation capability, nonlinearity of its model and capability of estimating posterior probabilities.

In particular, any *NN* consists of input layer with some input units (representing neurons), some hidden layer with certain number of hidden units in each layer and output layer with appropriate number of output units corresponding to the class labels. Samet et al. depicted a typical pseudo code for training an *NN* using *BP* algorithm in [7]. For our problem, the initialization of the entire network is done as usual and ultimate classification output, the class label is chosen as the dependent variable. The class label will be application specific and in our case it is considered as binary (0 or 1). The features extracted from source dataset are treated as independent variables.

For a model of m number of hidden layers, with an average of h number of units per layer, that is, $H_{1,1}, H_{1,2}, ..., H_{1,h}, H_{2,1}, H_{2,2}, ..., H_{2,h}, ..., H_{m,1}, ..., H_{m,h}$ are the hidden layer units. The connecting edges of input units X_i to first hidden layer units $H_{1,p}$ having weights $W_{i,(1,p)}$, where $P = [1, h]$. Consequently, the connecting edges of output units Y_j and m^{th} hidden layer units $H_{m,p}$ have weights $W_{(m,p),j}$, where $p = [1, h]$. Similarly, the weights associated with connected edges between q^{th} and $(q + 1)^{th}$ hidden layer units, $H_{q,r}$ and $H_{q+1,s}$, can be denoted as $W_{(q,r),(q+1,s)}$, where $r, s = [1, h], q = [1, m]$.

Weight Association with Features: In the proposed approach, two sets of source domain correlated diversified discriminative features (positive and negative features set) are calculated. These features are used for predicting labels of source domain unseen graph by treating all the features having same weighted vote. For effective prediction of class labels of unseen target domain graphs using these source domain features, the anomaly between different domain data distributions can be handled carefully. The effect of such anomaly can be reduced by adjusting the weights of the votes for each feature.

In our neural network model, where all source features are treated as independent variables and the class label as dependent variable, for each of the available target domain graphs G_t, with known class labels, we assign X_i = '1' as input for features $f_i \subseteq G_t$ and '0' for features $f_j \nsubseteq G_t$. If the class label of G_t is positive then the target value, Y_j is interpreted as '1' otherwise '0'.

All the available target domain graphs are used to train the model; in actual sense, to adjust the corresponding weights of the network edges by backpropagation algorithm, so that the system can effectively classify the available target domain graphs by adjusting network edge weights. At the termination of training phase, the feature weights can be calculated as follows for the feature f_i using the above stated *NN* model settings with $m \geq 2$:

$$W_{f_i} = \sum_{p=1}^{h} W_{i,(1,p)} \times \prod_{q=1}^{m-2} \sum_{p=1}^{h} W_{(q,p),(q+1,p)} \times \sum_{p=1}^{h} \{W_{(m-1,p),(m,p)} \times W_{(m,p),j=1}\} \quad (6)$$

Classifying Unseen Target Graphs: Using the features extracted from source graph set and associating weights to the *CDDF*s using the specified *NN* model, the transfer learning capable classifier is constructed. In another sense, a classifier model is constructed with a set of *WCDDF*s. For predicting the class label of unseen target graph, the weighted features are used as discriminative features and their weights are used as voting scores. For an unseen graph of target domain G_t, the score of the features are summed up those are covered by G_t.

3 Conclusions & Future Plan

There exists lots of approaches for efficiently predicting labels of unseen graphs with similar training and testing data domain. However, some facts such as correlation among entities and diversity capturing approaches are disregarded and need special deliberation. Hence, we have proposed new ways and measures for capturing more effective features to construct a better classifier model. Since, the assumption that training and operating dataset will exhibit similar properties is unrealistic in real life scenarios, a mechanism of annihilating the limitations of the assumption and the existing approaches for classifying unseen graphs of different domain is proposed. Our future plan is to conduct expansive experimental analysis to prove that the proposed feature selection and classification method outperforms the existing state-of-the-art graph classification methods.

Acknowledgments

This work was supported by the REFRAME project granted by the European Coordinated Research on Long-term Challenges in Information and Communication Sciences & Technologies ERA-Net (CHIST-ERA).

References

1. Jin, N., Young, C., Wang, W.: Graph classification based on pattern co-occurrence. In: CIKM. (2009) 573–582
2. Zhu, Y., Yu, J.X., Cheng, H., Qin, L.: Graph classification: a diversified discriminative feature selection approach. In: CIKM. (2012) 205–214
3. Cheng, H., Yan, X., Han, J., Hsu, C.W.: Discriminative frequent pattern analysis for effective classification. In: ICDE. (2007) 716–725
4. Shi, X., Kong, X., Yu, P.S.: Transfer significant subgraphs across graph databases. In: SDM. (2012) 552–563
5. He, J., Liu, Y., Lawrence, R.D.: Graph-based transfer learning. In: CIKM. (2009) 937–946
6. Samiullah, M., Ahmed, C.F., Fariha, A., Islam, M.R., Lachiche, N.: Mining frequent correlated graphs with a new measure. Expert Syst. Appl. **41** (2014) 1847–1863
7. Samet, S., Miri, A.: Privacy-preserving back-propagation and extreme learning machine algorithms. Data Knowl. Eng. **79-80** (2012) 40–61