Discriminative Feature Selection for Uncertain Graph Classification

Xiangnan Kong*   Philip S. Yu* †   Xue Wang‡   Ann B. Ragin‡

Abstract

Mining discriminative features for graph data has attracted much attention in recent years due to its important role in constructing graph classifiers, generating graph indices, etc. Most measurement of interestingness of discriminative subgraph features are defined on certain graphs, where the structure of graph objects are certain, and the binary edges within each graph represent the "presence" of linkages among the nodes. In many real-world applications, however, the linkage structure of the graphs is inherently uncertain. Therefore, existing measurements of interestingness based upon certain graphs are unable to capture the structural uncertainty in these applications effectively. In this paper, we study the problem of discriminative subgraph feature selection from uncertain graphs. This problem is challenging and different from conventional subgraph mining problems because both the structure of the graph objects and the discrimination score of each subgraph feature are uncertain. To address these challenges, we propose a novel discriminative subgraph feature selection method, DUG, which can find discriminative subgraph features in uncertain graphs based upon different statistical measures including expectation, median, mode and φ-probability. We first compute the probability distribution of the discrimination scores for each subgraph feature based on dynamic programming. Then a branch-and-bound algorithm is proposed to search for discriminative subgraphs efficiently. Extensive experiments on various neuroimaging applications (*i.e.*, Alzheimers Disease, ADHD and HIV) have been performed to analyze the gain in performance by taking into account structural uncertainties in identifying discriminative subgraph features for graph classification.

1 Introduction

Graphs arise naturally in many scientific applications which involve complex structures in the data, *e.g.*, chemical compounds, program flows, *etc.* Different from traditional data with flat features, these data are usually not directly represented as feature vectors, but as graphs with nodes and edges. Mining discriminative features for graph data has attracted much attention in recent years due to its important role in constructing graph classifiers, generating graph indices, etc. [22, 11, 4, 14, 20]. Much of the past research in discriminative subgraph feature mining has focused on certain graphs, where the structure of the graph objects are certain, and the binary edges represent the "presence" of linkages between the nodes. Conventional subgraph mining methods [22] utilize the structures of the certain graphs to find discriminative subgraph features. However, in many real-world applications, there is inherent uncertainty about the graph linkage structure. Such uncertainty information will be lost if we directly transform uncertain graphs into certain graphs.

For example, in neuroimaging, the functional connectivities among different brain regions are highly uncertain [6, 8, 7, 25]. In such applications, each human brain can be represented as an uncertain graph as shown in Figure 1, which is also called the "brain network" [2]. In such brain networks, the nodes represent brain regions, and edges represent the probabilistic connections, *e.g.*, resting-state functional connectivity in fMRI (functional Magnetic Resonance Imaging). Since these functional connectivities are derived based upon processing steps, such as temporal correlations in spontaneous blood oxygen level-dependent (BOLD) signal oscillations, each edge of the brain network is associated with a probability to quantify the likelihood that the functional connection exists in the brain. Resting-state functional connectivity has shown alterations related to many neurological diseases, such as ADHD (Attention Deficit Hyperactivity Disorder), Alzheimer's disease and virus infections that may affect the brain functioning, such as HIV [21]. Researchers are interested in analyzing the complex structure and uncertain connectivities of the human brain to find biomarkers for neurological diseases. Such biomarkers are clinically im-

---

*Department of Computer Science, University of Illinois at Chicago, USA. xkong4@uic.edu, psyu@cs.uic.edu.

†Computer Science Department, King Abdulaziz University, Jeddah, Saudi Arabia

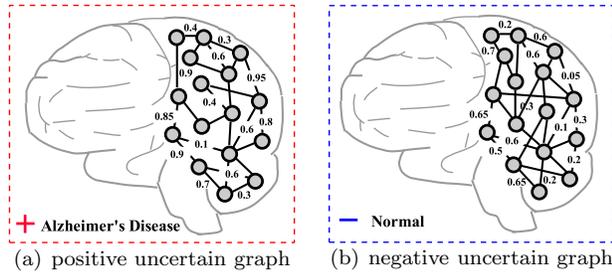‡Department of Radiology, Northwestern University, Chicago, IL, USA. {xue-wang, ann-ragin}@northwestern.edu.

Figure 1: An example of uncertain graph classification task.

Figure 2: Different types of subgraph features for uncertain graph classification

perative for detecting injury to the brain in the earliest stages before it is irreversible. Valid biomarkers can be used to aid diagnosis, monitor disease progression and evaluate effects of intervention.

Motivated by these real-world neuroimaging applications, in this paper, we study the problem of mining discriminative subgraph features in uncertain graph datasets. Discriminative subgraph features are fundamental for uncertain graphs, just as they are for certain graphs. They serve as primitive features for the classification tasks on uncertain graph objects. Despite the value and significance, the discriminative subgraph mining for uncertain graph classification has not been studied in this context. If we consider discriminative subgraph mining and uncertain graph structures as a whole, the major research challenges are as follows:

Structural Uncertainty: In discriminative subgraph mining, we need to estimate the discrimination score of a subgraph feature in order to select a set of subgraphs that are most discriminative for a classification task. In conventional subgraph mining, the discrimination scores of subgraph features are defined on certain graphs, where the structure of each graph object is certain, and thus the containment relationships between subgraph features and graph objects are also certain. However, when uncertainty is presented in the structures of graphs, a subgraph feature only exists within a graph object with a probability. Thus the discrimination scores of a subgraph feature are no longer deterministic values, but random variables with probability distributions.

Thus, the evaluation of discrimination scores for subgraph features in uncertain graphs is different from conventional subgraph mining problems. For example, in Figure 2, we show an uncertain graph dataset containing 4 uncertain graphs $\widetilde{G}_1, \cdots, \widetilde{G}_4$ with their class labels, + or −. Subgraph g_1 is a frequent pattern among the uncertain graphs, but it may not relate to the class labels of the graphs. Subgraph g_2 is a discriminative subgraph features when we ignore the edge uncertain-
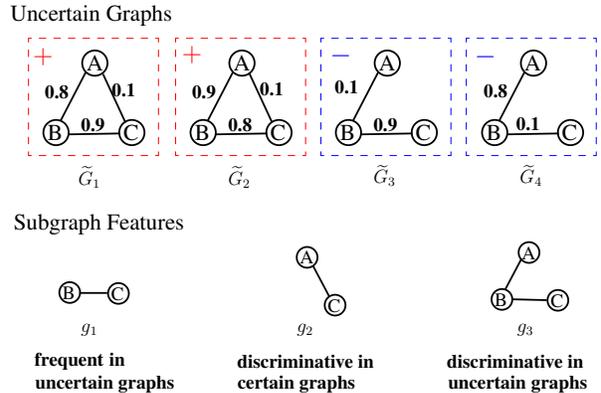
ties. However, if such uncertainties are considered, we will find that g_2 can rarely be observed within the uncertain graph dataset, and thus will not be useful in graph classification. Accordingly, g_3 is the best subgraph feature for uncertain graph classification.

Efficiency & Robustness: There are two additional problems that need to be considered when evaluating features for uncertain graphs: 1) In an uncertain graph dataset, there are an exponentially large number of possible instantiations of a graph dataset [26]. How can we efficiently compute the discrimination score of a subgraph feature without enumerating all possible implied datasets? 2) When evaluating the subgraph features, we should choose a statistical measure for the probablity disctribution of discrimination scores which is robust to extreme values. For example, given a subgraph feature with (score, probability) pairs as $(0.01, 99.99\%)$ and $(+\infty, 0.01\%)$, the *expected* score of the subgraph is $+\infty$, although this value is only associated with a very tiny probability.

In order to address the above problems, we propose a general framework for mining discriminative subgraph features in uncertain graph datasets, which is called DUG (Discriminative feature selection for Uncertain Graph classification). The DUG framework can effectively find a set of discriminative subgraph features by considering the relationship between uncertain graph structures and labels based upon various statistical measures. We propose an efficient method to calculate the probability distribution of the scoring function based on dynamic programming. Then a branch-and-bound algorithm is proposed to search for the discriminative subgraphs efficiently by pruning the subgraph search space. Empirical studies on resting-state fMRI images of different brain diseases (i.e., Alzheimer's Disease, ADHD and HIV) demonstrate that the proposed method can obtain better accuracy on uncertain graph classification

tasks than alternative approaches.

For the rest of the paper, we first introduce preliminaries in Section 2. Then we introduce our Dug subgraph mining framework in Section 3. Discrimination score functions based upon different statistic measures are discussed in Section 3.1. An efficient algorithm for computing the score distribution based upon dynamic programming is proposed in Section 3.2. Experimental results are discussed in Section 4. In Section 6, we conclude the paper.

2 Problem Formulation

In this section, we formally define the model of uncertain graphs and the problem of discriminative subgraph mining in uncertain graph datasets. Suppose we are given an uncertain graph dataset $\widetilde{\mathcal{D}} = \{\widetilde{G}_1, \cdots, \widetilde{G}_n\}$ that consists of n uncertain graphs. $\widetilde{G}_i$ is the i-th uncertain graph in $\widetilde{\mathcal{D}}$. $\mathbf{y} = [y_1, \cdots, y_n]^\top$ denotes the vector of class labels, where $y_i \in \{+1, -1\}$ is the class label of $\widetilde{G}_i$. We also denote the subset of $\widetilde{\mathcal{D}}$ that contains only positive/negative graphs as $\widetilde{\mathcal{D}}_+ = \{\widetilde{G}_i | \widetilde{G}_i \in \widetilde{\mathcal{D}} \bigwedge y_i = +1\}$ and $\widetilde{\mathcal{D}}_- = \{\widetilde{G}_i | \widetilde{G}_i \in \widetilde{\mathcal{D}} \bigwedge y_i = -1\}$ respectively.

DEFINITION 1. (CERTAIN GRAPH) *A certain graph is an undirected and deterministic graph represented as* $G = (V, E)$. $V = \{v_1, \cdots, v_{n_v}\}$ *is the set of vertices.* $E \subseteq V \times V$ *is the set of deterministic edges.*

DEFINITION 2. (UNCERTAIN GRAPH) *An uncertain graph is an undirected and nondeterministic graph represented as* $\widetilde{G} = (V, E, p)$. $V = \{v_1, \cdots, v_{n_v}\}$ *is the set of vertices.* $E \subseteq V \times V$ *is the set of nondeterministic edges.* $p : E \to (0, 1]$ *is a function that assigns a probability of existence to each edge in* E. $p(e)$ *denotes the existence probability of edge* $e \in E$.

Consider an uncertain graph $\widetilde{G}(V, E, p) \in \widetilde{\mathcal{D}}$, where each edge $e \in E$ is associated with a probability $p(e)$ of being present. As in previous works [27, 26], we assume that the uncertainty variables of different edges in an uncertain graph are independent from each other, though most of our results are still applicable to graphs with edge correlations. We further assume that all uncertain graphs in a dataset $\widetilde{\mathcal{D}}$ share a same set of nodes V and each node in V has a unique node label, which is reasonable in many applications like neuroimaging, since each human brain consists of the same number of regions. The main difference between different uncertain graphs is on their linkage structures, *i.e.*, the edge sets $E(\widetilde{G})$ and the edge probabilities $p(e)$.

Possible instantiations of an uncertain graph $\widetilde{G}$ are usually referred to as *worlds* of $\widetilde{G}$, where each world

corresponds to an implied certain graph G. Here G is *implied* from uncertain graph $\widetilde{G}$ (denoted as $\widetilde{G} \Rightarrow G$), iff all edges in $E(G)$ are sampled from $E(\widetilde{G})$ according to their probabilities of existence in $p(e)$ and $E(G) \subseteq E(\widetilde{G})$. There are $2^{|E(\widetilde{G})|}$ possible worlds for uncertain graph $\widetilde{G}$, denoted as $\mathcal{W}(\widetilde{G}) = \{G | \widetilde{G} \Rightarrow G\}$. Thus, each uncertain graph $\widetilde{G}$ corresponds to a probability distribution over $\mathcal{W}(\widetilde{G})$. We denote the probability of each certain graph $G \in \mathcal{W}(\widetilde{G})$ being implied by the uncertain graph $\widetilde{G}$ as $\Pr(\widetilde{G} \Rightarrow G)$, and we have

$$\Pr\left[\widetilde{G} \Rightarrow G\right] = \prod_{e \in E(G)} \Pr_{\widetilde{G}}(e) \prod_{e \in E(\widetilde{G}) - E(G)} \left(1 - \Pr_{\widetilde{G}}(e)\right)$$

Similarly, possible instantiations of an uncertain graph dataset $\widehat{\mathcal{D}} = \{\widetilde{G}_1, \cdots, \widetilde{G}_n\}$ are referred to as *worlds* of $\widetilde{\mathcal{D}}$, where each world corresponds to an implied certain graph dataset $\mathcal{D} = \{G_1, \cdots, G_n\}$. A certain graph dataset $\mathcal{D}$ is called as being *implied* from uncertain graph dataset $\widetilde{\mathcal{D}}$ (denoted as $\widetilde{\mathcal{D}} \Rightarrow \mathcal{D}$), iff $|\mathcal{D}| = |\widetilde{\mathcal{D}}|$ and $\forall i \in \{1, \cdots, |\mathcal{D}|\}$, $\widetilde{G}_i \Rightarrow G_i$. There are $\prod_{i=1}^{|\widetilde{\mathcal{D}}|} 2^{|E(\widetilde{G}_i)|}$ possible worlds for uncertain graph dataset $\widetilde{\mathcal{D}}$, denoted as $\mathcal{W}(\widetilde{\mathcal{D}}) = \{\mathcal{D} \mid \widetilde{\mathcal{D}} \Rightarrow \mathcal{D}\}$. An uncertain graph dataset $\widetilde{\mathcal{D}}$ corresponds to a probability distribution over $\mathcal{W}(\widetilde{\mathcal{D}})$. We denote the probability of each certain graph dataset $\mathcal{D} \in \mathcal{W}(\widetilde{\mathcal{D}})$ being implied by $\widetilde{\mathcal{D}}$ as $\Pr(\widetilde{\mathcal{D}} \Rightarrow \mathcal{D})$. By assuming that different uncertain graphs are independent from each other, we have

$$\Pr\left[\widetilde{\mathcal{D}} \Rightarrow \mathcal{D}\right] = \prod_{i=1}^{|\widetilde{\mathcal{D}}|} \Pr[\widetilde{G}_i \Rightarrow G_i]$$

The concept of *subgraph* is defined based upon certain graphs. Different from conventional subgraph mining problems where each subgraph feature can have multiple embeddings within one graph object, in our data model, each subgraph feature g can only have one unique embedding within a certain graph G.

DEFINITION 3. (SUBGRAPH) *Let* $g = (V', E')$ *and* $G = (V, E)$ *be two certain graphs.* g *is a subgraph of* G *(denoted as* $g \subseteq G$*) iff* $V' \subseteq V$ *and* $E' \subseteq E$. *We use* $g \subseteq G$ *to denote that graph* g *is a subgraph of* G. *We also say that* G *contains subgraph* g.

For an uncertain graph $\widetilde{G}$, the probability of $\widetilde{G}$ containing a subgraph feature g is defined as follows:

$$\Pr(g \subseteq \widetilde{G}) = \sum_{G \in \mathcal{W}(\widetilde{G})} \Pr(\widetilde{G} \Rightarrow G) \cdot I(g \subseteq G)$$

$$= \begin{cases} \prod_{e \in E(g)} p(e) & \text{if } E(g) \subseteq E(\widetilde{G}) \\ 0 & \text{otherwise} \end{cases}$$

Table 1: Important Notations.

Symbol	Definition
$\widetilde{\mathcal{D}} = \{\widetilde{G}_1, \cdots, \widetilde{G}_n\}$	uncertain graph dataset, $\widetilde{G}_i$ denotes the i-th uncertain graph in the dataset.
$\mathbf{y} = [y_1, \cdots, y_n]^\top$	class label vector for graphs in $\widetilde{\mathcal{D}}$, $y_i \in \{+1, -1\}$.
$\widetilde{\mathcal{D}}_+$ and $\widetilde{\mathcal{D}}_-$	the subset of $\widetilde{\mathcal{D}}$ with only positive/negative graphs, $\widetilde{\mathcal{D}}_+ = \{\widetilde{G}_i | \widetilde{G}_i \in \widetilde{\mathcal{D}}, y_i = +1\}$.
n_+ and n_-	number of positive graphs and number of negative graphs in $\widetilde{\mathcal{D}}$, $n_+ = |\widetilde{\mathcal{D}}_+|$ and $n_- = |\widetilde{\mathcal{D}}_-|$.
$\mathcal{D} = \{G_1, \cdots, G_n\}$	a certain graph dataset implied from $\widetilde{\mathcal{D}}$, G_i denotes the certain graph implied from $\widetilde{G}_i$.
$g \subseteq G$	graph G contains subgraph feature g
n_+^g and n_-^g	number of graphs in $\mathcal{D}_+$ / $\mathcal{D}_-$ that contains subgraph g, $n_+^g = |\{G_i | g \subseteq G_i, G_i \in \mathcal{D}_+\}|$.
$\widetilde{G} \Rightarrow G$ and $\widetilde{\mathcal{D}} \Rightarrow \mathcal{D}$	certain graph G is implied from uncertain graph $\widetilde{G}$; $\mathcal{D}$ is implied from $\widetilde{\mathcal{D}}$.
$\mathcal{W}(\widetilde{G})$ and $\mathcal{W}(\widetilde{\mathcal{D}})$	the possible worlds of $\widetilde{G}$ and $\widetilde{\mathcal{D}}$, $\mathcal{W}(\widetilde{G}) = \{G | \widetilde{G} \Rightarrow G\}$, $\mathcal{W}(\widetilde{\mathcal{D}}) = \{\mathcal{D} | \widetilde{\mathcal{D}} \Rightarrow \mathcal{D}\}$.
$E(\widetilde{G}_i)$ and $E(G_i)$	the set of edges in $\widetilde{G}_i$ and G_i
$\widetilde{\mathcal{D}}_+(k)$ and $\widetilde{\mathcal{D}}_-(k)$	the first k graphs in $\widetilde{\mathcal{D}}_+$ or $\widetilde{\mathcal{D}}_-$

which corresponds to the probability that a certain graph G implied by $\widetilde{G}$ contains subgraph g.

We focus on mining a set of discriminative subgraph features to define the feature space of graph classification. It is assumed that a graph object $\widetilde{G}_i$ is represented as a feature vector $\mathbf{x}_i = [x_i^1, \cdots, x_i^m]^\top$ associated with a set of subgraph features $\{g_1, \cdots, g_m\}$. Here, $x_i^k = \Pr(g_k \subseteq \widetilde{G}_i)$ is the probability that $\widetilde{G}_i$ contains the subgraph feature g_k. Now suppose the full set of subgraph features in the graph dataset $\widetilde{\mathcal{D}}$ is $\mathcal{S} = \{g_1, \cdots, g_m\}$, which we use to predict the class labels of the graph objects. The full feature set $\mathcal{S}$ is very large. Only a subset of the subgraph features ($\mathcal{T} \subseteq \mathcal{S}$) is relevant to the graph classification task, which is the target feature set we want to find within uncertain graphs.

The key issues of discriminative subgraph mining for uncertain graphs can be described as follows:

(P1) How can one properly evaluate the discrimination scores of a subgraph feature considering the uncertainty of the graph structures?

(P2) How can one efficiently compute the probability distribution of a subgraph's discrimination score by avoiding the exhaustive enumeration of all possible worlds of the uncertain graph dataset? Moreover, since the subgraph enumeration is NP-hard, it is also infeasible to fully enumerate all the subgraph features for an uncertain graph dataset.

In the following sections, we will introduce the proposed framework for mining discriminative subgraphs from uncertain graphs.

3 The Proposed Framework

3.1 Discrimination Score Distribution In this subsection, we address the problem (P1) discussed in the previous section. In conventional discriminative subgraph mining, the discrimination scores of subgraph features are usually defined for certain graph datasets, e.g., information gain and G-test score [22]. The score of a subgraph feature is a fixed value indicating the dis-

criminative power of the subgraph feature for the graph classification task. However, such concepts don't make sense to uncertain graph datasets, since an uncertain graph only contains a subgraph feature in a probabilistic sense. Now we extend the concept of discriminative subgraph features in uncertain graph datasets. Suppose we have an objective function $F(g, \mathcal{D})$ which measures the discrimination score of a subgraph g in a certain graph dataset $\mathcal{D}$. The corresponding objective function on an uncertain graph dataset $\widetilde{\mathcal{D}}$ can be written as $F(g, \widetilde{\mathcal{D}})$ accordingly. Note that $F(g, \widetilde{\mathcal{D}})$ is no longer a deterministic function. $F(g, \widetilde{\mathcal{D}})$ corresponds to a random variable over all possible outcomes of $F(g, \mathcal{D})$ (*i.e.*, Range(F)) with probability distribution:

s_1	s_2	$\cdots$
$\Pr[F(g, \widetilde{\mathcal{D}}) = s_1]$	$\Pr[F(g, \widetilde{\mathcal{D}}) = s_2]$	$\cdots$

where $s_i \in \text{Range}(F)$.

The probability distribution of the discrimination score values can be defined as follows:

$$\Pr\Big[F(g, \widetilde{\mathcal{D}}) = s\Big] = \sum_{\mathcal{D} \in \mathcal{W}(\widetilde{\mathcal{D}})} \Pr[\widetilde{\mathcal{D}} \Rightarrow D] \cdot I\left(F(g, \mathcal{D}) = s\right)$$

where $I(\pi) \in \{0, 1\}$ is an indicator function, and $I(\pi) = 1$ iff π holds. In other words, $\forall s \in Range(F)$, $\Pr[F(g, \widetilde{\mathcal{D}}) = s]$ is the summation over the probabilities of all worlds of $\widetilde{\mathcal{D}}$ in which the discrimination score $F(g, \mathcal{D})$ is exactly s. Based on the discrimination score function on uncertain graphs, we define four statistical measures that evaluate the properties of the distribution of $F(g, \widetilde{\mathcal{D}})$ from different perspectives.

DEFINITION 4. (MEAN-SCORE) *Given an uncertain graph dataset $\widetilde{\mathcal{D}}$, a subgraph feature g and a discrimination score function $F(\cdot, \cdot)$, we define the expected discrimination score* $\text{EXP}(F(g, \widetilde{\mathcal{D}}))$ *as the mean score*

among all possible worlds of $\widetilde{\mathcal{D}}$:

$$\text{Exp}\left(F(g, \widetilde{\mathcal{D}})\right) = \sum_{\mathcal{D} \in \mathcal{W}(\widetilde{\mathcal{D}})} \text{Pr}[\widetilde{\mathcal{D}} \Rightarrow D] \cdot F(g, \mathcal{D})$$

$$= \sum_{s=-\infty}^{+\infty} s \cdot \text{Pr}[F(g, \widetilde{\mathcal{D}}) = s]$$

The mean discrimination score is the expectation of the random variable $F(g, \widetilde{\mathcal{D}})$. The expectation is usually used in conventional frequent pattern mining on uncertain datasets [27, 26]. However, it's worth noting that the expectation of discrimination scores may not be robust to extreme values. In discriminative subgraph mining, the value of a score function (*e.g.*, frequency ratio[10], G-test score[22]) can be $+\infty$. Such cases can easily dominate the computation of expectation, even if the probabilities are extremely small. For example, suppose we have a subgraph feature with the (score, probability) pairs as $(0.01, 99.99\%)$ and $(+\infty, 0.01\%)$. The expected score will be $+\infty$. In order to address this problem, we either need to bound the maximum value of the objective function like $\min(F(g, \widetilde{\mathcal{D}}), \frac{1}{\epsilon})$, or we need to introduce other statistical measures which are robust to extreme values.

DEFINITION 5. (MEDIAN-SCORE) *Given an uncertain graph dataset $\widetilde{\mathcal{D}}$, a subgraph feature g and a discrimination score function $F(\cdot, \cdot)$ on certain graphs, we define the median discrimination score $\text{MEDIAN}(F(g, \widetilde{\mathcal{D}}))$ as the median score among all possible worlds of $\widetilde{\mathcal{D}}$:*

$$\text{MEDIAN}\left(F(g, \widetilde{\mathcal{D}})\right) = \arg\max_{S} \left\{ \sum_{s=-\infty}^{S} \text{Pr}\left[F(g, \widetilde{\mathcal{D}}) = s\right] \le \frac{1}{2} \right\}$$

The median score is relatively more robust to extreme values than expectation, although in some cases the median score can still be infinite. The same results can also hold for any quantile or k-th order statistic.

Another commonly used statistic is the mode score, *i.e.*, the score value that has the largest probability. The mode score of a distribution means that the score is most likely to be observed within all possible worlds of $\widetilde{\mathcal{D}}$.

DEFINITION 6. (MODE-SCORE) *Given an uncertain graph dataset $\widetilde{\mathcal{D}}$, a subgraph feature g and a discrimination score function $F(\cdot, \cdot)$, we define the mode discrimination score $\text{MODE}(F(g, \widetilde{\mathcal{D}}))$ as the score that is most likely among all possible worlds of $\widetilde{\mathcal{D}}$:*

$$\text{MODE}\left(F(g, \widetilde{\mathcal{D}})\right) = \arg\max_{s} \text{Pr}\left[F(g, \widetilde{\mathcal{D}}) = s\right]$$

Next we consider the probability of a subgraph feature being observed as a discriminative pattern within all possible worlds of $\widetilde{\mathcal{D}}$, *i.e.*, $\text{Pr}[F(g, \widetilde{\mathcal{D}}) \ge \varphi]$. It is

Table 2: Summary of Discrimination Score Functions.

Name	$f(n_+^g, n_-^g, n_+, n_-)$
confidence	$\frac{n_+^g}{n_+^g + n_-^g}$
frequency ratio	$\left\lvert \log \frac{n_+^g \cdot n_-}{n_-^g \cdot n_+} \right\rvert$
G-test	$2n_+^g \cdot \ln \frac{n_+^g \cdot n_-}{n_-^g \cdot n_+} + 2(n_+ - n_+^g) \cdot \ln \frac{n_- \cdot (n_+ - n_+^g)}{n_+ \cdot (n_- - n_-^g)}$
HSIC(linear)	$\frac{(n_+^g \cdot n_- - n_-^g \cdot n_+)^2}{(n_+ + n_- - 1)^2 (n_+ + n_-)^2}$

called φ-probability. The higher the value, the more likely that the subgraph feature is a discriminative pattern with a score larger or equals to a threshold φ.

DEFINITION 7. (φ-PROBABILITY) *Given an uncertain graph dataset $\widetilde{\mathcal{D}}$, a subgraph feature g and a discrimination score function $F(\cdot, \cdot)$, we define the φ-probability for discrimination score function $F(g, \widetilde{\mathcal{D}})$ as the sum of probabilities for all possible worlds of $\widetilde{\mathcal{D}}$, where the score is greater than or equals to φ:*

$$\varphi\text{-Pr}\left(F(g, \widetilde{\mathcal{D}})\right) = \sum_{\mathcal{D} \in \mathcal{W}(\widetilde{\mathcal{D}})} \text{Pr}[\widetilde{\mathcal{D}} \Rightarrow D] \cdot I\left(F(g, \mathcal{D}) \ge \varphi\right)$$

$$= \sum_{s=\varphi}^{+\infty} \text{Pr}[F(g, \widetilde{\mathcal{D}}) = s]$$

The φ-probability is robust to extreme values of the objective function. For the previous example, we have a subgraph feature with score distribution: $(0.01, 99.99\%), (+\infty, 0.01\%)$. The φ-probability is 0.01%, when $\varphi = 1$.

We have already introduced four statistical measures of the distribution of a discrimination score function. Now the central problem for calculating all these measures is how to calculate $\text{Pr}[F(g, \widetilde{\mathcal{D}}) = s]$ efficiently, which we will discuss in the following section.

3.2 Efficient Computation In this subsection, we address the problem (P2) discussed in Section 2. Given a certain graph dataset $\mathcal{D}$, we denote the subsets of all positive graphs and all negative graphs as $\mathcal{D}_+$ and $\mathcal{D}_-$, respectively. Suppose the supports of subgraph feature g in $\mathcal{D}_+$ and $\mathcal{D}_-$ are n_+^g and n_-^g. $n_+^g = |\{G; G \in \mathcal{D}_+, g \subseteq G\}|$. Most of the existing discrimination score functions can be written as a function of n_+^g, n_-^g, n_+ and n_-:

$$(3.1) \qquad F(g, \mathcal{D}) = f\left(n_+^g, n_-^g, n_+, n_-\right)$$

The definition in Eq. 3.1 covers many discrimination score functions including confidence[5], frequency ratio[10], information gain, G-test score[22] and HSIC[13], as shown in Table 2. For example, frequency
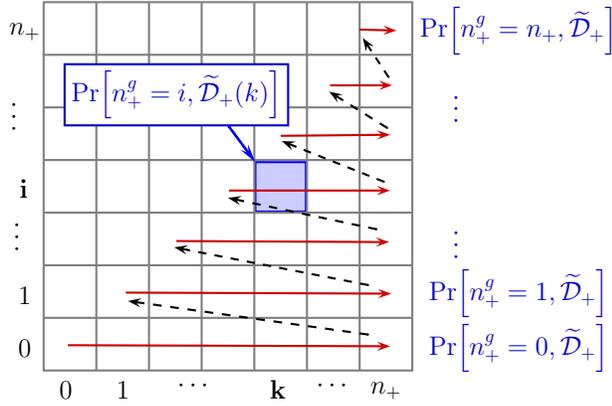
Figure 3: The dynamic programming process for computing $\Pr\left[n_+^g, \widetilde{\mathcal{D}}_+\right]$. The same process applies for $\Pr\left[n_-^g, \widetilde{\mathcal{D}}_-\right]$.

ratio can be written as $\mathrm{r}(g) = |\log \frac{n_+^g \cdot n_-}{n_-^g \cdot n_+}|$. The G-test score can be written as G-test$(g) = 2n_+^g \cdot \ln \frac{n_+^g \cdot n_-}{n_-^g \cdot n_+} + 2\left(n_+ - n_+^g\right) \cdot \ln \frac{n_- \cdot (n_+ - n_+^g)}{n_+ \cdot (n_- - n_-^g)}$. Because n_+ and n_- are fixed numbers for different subgraph features, we simply use $f(n_+^g, n_-^g)$ for $f(n_+^g, n_-^g, n_+, n_-)$.

Based on the above definitions, we find that the number of possible outcomes of $F(g, \widetilde{\mathcal{D}})$ is bounded by $n_+ \times n_-$, because $0 \le n_+^g \le n_+$ and $0 \le n_-^g \le n_-$. Thus, the probabilities $\Pr[F(g, \widetilde{\mathcal{D}}) = s]$ can be exactly computed via dynamic programming in $O(n^2)$ time, without enumerating all possible worlds of $\widetilde{\mathcal{D}}$. Instead, we can just enumerate all possible combinations of (n_+^g, n_-^g) and calculate the probability for each pair (n_+^g, n_-^g), denoted as $\Pr[n_+^g, n_-^g, \widetilde{\mathcal{D}}] = \Pr\left[F(g, \widetilde{\mathcal{D}}) = f(n_+^g, n_-^g)\right]$. Then the values of $F(g, \widetilde{\mathcal{D}})$ in all possible worlds with non-zero probabilities can be covered by the $n_+ \times n_-$ cases.

Moreover, because different uncertain graphs are independent from each other, we have

$$(3.2) \qquad \Pr[n_+^g, n_-^g, \widetilde{\mathcal{D}}] = \Pr[n_+^g, \widetilde{\mathcal{D}}_+] \cdot \Pr[n_-^g, \widetilde{\mathcal{D}}_-]$$

where $\Pr[n_+^g, \widetilde{\mathcal{D}}_+]$ denotes the probability of the cases when there are n_+^g graphs in $\widetilde{\mathcal{D}}_+$ that contain the subgraph g. $\Pr[n_-^g, \widetilde{\mathcal{D}}_-]$ corresponds to the cases when there are n_-^g graphs in $\widetilde{\mathcal{D}}_-$ that contain subgraph g. Now we just need to compute the probabilities $\Pr[n_+^g, \widetilde{\mathcal{D}}_+]$ ($\forall n_+^g, 0 \le n_+^g \le n_+$) and $\Pr[n_-^g, \widetilde{\mathcal{D}}_-]$ ($\forall n_-^g, 0 \le n_-^g \le n_-$) separately.

Let $\widetilde{\mathcal{D}}(k)$ denote the first k uncertain graphs in $\widetilde{\mathcal{D}}$, i.e., $\widetilde{\mathcal{D}}(k) = \{\widetilde{G}_1, \cdots, \widetilde{G}_k\}$. $\widetilde{\mathcal{D}}_+(k)$ and $\widetilde{\mathcal{D}}_-(k)$

denote the first k graphs in $\widetilde{\mathcal{D}}_+$ and $\widetilde{\mathcal{D}}_-$ respectively. All the values of $\Pr[n_+^g, \widetilde{\mathcal{D}}_+]$ and $\Pr[n_-^g, \widetilde{\mathcal{D}}_-]$ can be calculated using the recursive equation in Figure 5. The $\Pr[i, \widetilde{\mathcal{D}}(k)]$ denotes the probability when there are i graphs containing g in $\widetilde{\mathcal{D}}(k)$. And the target values to calculate are $\Pr[i, \widetilde{\mathcal{D}}_+(n_+)]$ ($\forall i, 0 \le i \le n_+$) and $\Pr[i, \widetilde{\mathcal{D}}_-(n_-)]$ ($\forall i, 0 \le i \le n_-$) by substituting the $\widetilde{\mathcal{D}}_+$ and $\widetilde{\mathcal{D}}_-$ into the Eq. 3.3, respectively. In Figure 4, we showed the dynamic programing algorithm to compute the target values using Eq. 3.3. Figure 3 illustrates the computation process of the dynamic programing algorithm for $\Pr[n_+^g, \widetilde{\mathcal{D}}_+]$, while the same process also applies for $\Pr[n_-^g, \widetilde{\mathcal{D}}_-]$.

For details of the recursive equations in Figure 5, we have the base cases, $\Pr[0, \widetilde{\mathcal{D}}_0] = 1$ and $\Pr[i, \widetilde{\mathcal{D}}(k)] = 0$ (if $i > k$ or $i < 0$). For other cases, the probability value can be calculated through the recursive equation in Eq. 3.3. Then, $\Pr[n_+^g, n_-^g, \widetilde{\mathcal{D}}]$ can be calculated via Eq. 3.2. Thus all the statistical measures mentioned in Section 3.1 can be calculated within $O(n^2)$ time as follows:

$$\mathrm{EXP}\left(F(g, \widetilde{\mathcal{D}})\right) = \sum_{n_+^g = 0}^{n_+} \sum_{n_-^g = 0}^{n_-} \Pr[n_+^g, n_-^g, \widetilde{\mathcal{D}}] \cdot f(n_+^g, n_-^g)$$

$$\mathrm{MEDIAN}\left(F(g, \widetilde{\mathcal{D}})\right) = \arg\max_s \left\{ \sum_{x = -\infty}^{s} \sum_{f(n_+^g, n_-^g) = x} Pr[n_+^g, n_-^g, \widetilde{\mathcal{D}}] \le \frac{1}{2} \right\}$$

$$\mathrm{MODE}\left(F(g, \widetilde{\mathcal{D}})\right) = \arg\max_s \sum_{n_+^g = 0}^{n_+} \sum_{n_-^g = 0}^{n_-} \Pr[n_+^g, n_-^g, \widetilde{\mathcal{D}}] \cdot I(f(n_+^g, n_-^g) = s)$$

$$\varphi\text{-}\Pr\left(F(g, \widetilde{\mathcal{D}})\right) = \sum_{n_+^g = 0}^{n_+} \sum_{n_-^g = 0}^{n_-} \Pr[n_+^g, n_-^g, \widetilde{\mathcal{D}}] \cdot I(f(n_+^g, n_-^g) \ge \varphi)$$

We will show later that the dynamic programming process is highly efficient in all the applications studied in Section 4. For dataset with even larger number of graphs, the divid-and-conquer method in [19] could also be used here to further optimize the computational cost.

3.3 Upper-Bounds for Subgraph Pruning In order to avoid the exhaustive enumeration of subgraph features, we derive some subgraph pruning methods. One natural pruning bound for subgraph search is the expected frequency of a subgraph feature, Exp-Freq$(g, \widetilde{\mathcal{D}}) = \frac{\sum_i^n \Pr(g \subseteq \widetilde{\mathcal{G}}_i)}{n}$, since it's can be easily proved with anti-monotonic property. For the expectation and φ-probability, we can also derive additional bounds for subgraph pruning. Let $\hat{F}(g, \mathcal{D}) = \hat{f}(n_+^g, n_-^g)$ be the estimated upper-bound function for g and its supergraphs in certain graph dataset $\mathcal{D}$. We can derive

$$\text{(3.3)} \qquad \Pr\Big[i, \widetilde{\mathcal{D}}(k)\Big] = \begin{cases} \Big(1 - \Pr[g \subseteq \widetilde{G}_k]\Big) \cdot \Pr[i, \widetilde{\mathcal{D}}(k-1)] \ + \ \Pr[g \subseteq \widetilde{G}(k)] \cdot \Pr[i-1, \widetilde{\mathcal{D}}(k-1)] & \text{if } i \leq k \\ 1 & \text{if } i = k = 0 \\ 0 & \text{if } i > k \text{ or } i < 0 \end{cases}$$

Figure 5: Recursive equation for dynamic programming.

Input:
   $\widetilde{\mathcal{D}}$: the uncertain graph dataset $\{\widetilde{G}_1, \cdots, \widetilde{G}_n\}$ t: the maximum number of subgraphs.
   y: the vector of class labels for uncertain graphs, min_sup: the minimum expected frequency.
   M: the statistic measure (Expectation/Median/Mode/φ-Pr)

Recursive Subgraphs Mining:
  - Depth-first search the gSpan's code tree and update the feature list as follows:
    1. Update the candidate feature list using the current subgraph feature g_c:
       Calculate the probability vector $\Pr[n_+^{g_c}, \widetilde{\mathcal{D}}_+]$ and $\Pr[n_-^{g_c}, \widetilde{\mathcal{D}}_-]$ using the dynamic programing algorithm in Figure 4
       Compute the statistic measure $M\left(F(g_c, \widetilde{\mathcal{D}})\right)$ based on the discrimination score function $F(g_c, \widetilde{\mathcal{D}})$.
       If the score is larger than the worst feature in $\mathcal{T}$, replace it and update $\theta = \min_{g \in \mathcal{T}} M\left(F(g, \widetilde{\mathcal{D}})\right)$
    2. Test pruning criteria for the sub-tree rooted from node g as follows:
      if Exp-Freq$(g_c) \leq min\_sup$, prune the sub-tree of g_c
      if Bound-$M\left(F(g_c, \widetilde{\mathcal{D}})\right) \leq \theta$, prune the sub-tree of g_c
    3. Recursion: Depth-first search the sub-tree rooted from node g_c
Output:
   $\mathcal{T}$:   the discriminative subgraph features for uncertain graph classification.

Figure 6: The DUG framework for discriminative subgraph mining.

Input:
  $\widetilde{\mathcal{D}}_+$: the set of positive graphs
  $\widetilde{\mathcal{D}}_-$: the set of negative graphs

Dynamic Programming:
  for $n_+^g \leftarrow 0$ to n_+
    for $k \leftarrow n_+^g$ to n_+
      compute $\Pr[n_+^g, \widetilde{\mathcal{D}}_+(k)]$ via Eq. 3.3;
  for $n_-^g \leftarrow 0$ to n_-
    for $k \leftarrow n_-^g$ to n_-
      compute $\Pr[n_-^g, \widetilde{\mathcal{D}}_-(k)]$ via Eq. 3.3;
Output:
  $\Pr[n_+^g, \widetilde{\mathcal{D}}_+]$ $(\forall\, n_+^g,\ 0 \leq n_+^g \leq n_+)$
  $\Pr[n_-^g, \widetilde{\mathcal{D}}_-]$ $(\forall\, n_-^g,\ 0 \leq n_-^g \leq n_-)$

Figure 4: The dynamic programming algorithm for probability computation.

the corresponding upper-bounds as follows:

$$\text{UB-EXP}(g, \widetilde{\mathcal{D}}) = \sum_{n_+^g=0}^{n_+} \sum_{n_-^g=0}^{n_-} \Pr[n_+^g, n_-^g, \widetilde{\mathcal{D}})] \cdot \hat{f}(n_+^g, n_-^g)$$

$$\text{UB-}\varphi\text{-Pr}(g, \widetilde{\mathcal{D}}) = \sum_{n_+^g=0}^{n_+} \sum_{n_-^g=0}^{n_-} \Pr[n_+^g, n_-^g, \widetilde{\mathcal{D}})] \cdot I(\hat{f}(n_+^g, n_-^g) \geq \varphi)$$

For the median and mode measures, it is difficult to derive a meaningful bound, thus we simply use the expected frequency to perform the subgraph pruning.

We now utilize the above bounds to prune the DFS-code tree in gSpan [23] by the branch-and-bound pruning. The top-t best features are maintained in a candidate list. During the subgraph mining, we calculate the upper-bound of each subgraph feature in the search tree. If a subgraph feature with its children pattern cannot update the candidate feature list, we can prune the subtree of gSpan rooted from this node. It is guaranteed by the upper-bounds that we will not miss any better subgraph features. Thus, the subgraph mining process can be speeded up without loss of performance. The algorithm of DUG is summarized in Figure 6.

4 Experiments

In order to evaluate the performance of the proposed approach for uncertain graph classification, we tested our algorithm on real-world fMRI brain images as summarized in Table 3.

4.1 Data Collection In order to evaluate the performance of the proposed approach for uncertain graph classification, we tested our algorithm on real-world fMRI brain images.

• *Alzheimer's Disease (ADNI)*: The first dataset is collected from the Alzheimer's Disease Neuroimaging Initiative[1]. The dataset consists of records of patients with Alzheimer's Disease (AD) and Mild Cognitive Impairment (MCI). We downloaded all records of resting-state fMRI images and treated the normal brains as

---
[1]http://adni.loni.ucla.edu/

negative graphs, and AD+MCI as the *positive* graphs. We applied Automated Anatomical Labeling (AAL[2]) to extract a sequence of responds from each of of the 116 anatomical volumes of interest (AVOI), where each AVOI represents a different brain region. The correlations of brain activities among different brain regions are computed. Positive correlations are used as uncertain links among brain regions. For details, we used SPM8 toolbox[3], and functional images were realigned to the first volume, slice timing corrected, and normalized to the MNI template and spatially smoothed with an 8-mm Gaussian kernel. Resting-State fMRI Data Analysis Toolkit (REST[4]) was then used to remove the linear trend of time series and temporally band-pass filtering (0.01-0.08 Hz). Before the correlation analysis, several sources of spurious variance were then removed from the data through linear regression: (i) six parameters obtained by rigid body correction of head motion, (ii) the whole-brain signal averaged over a fixed region in atlas space, (iii) signal from a ventricular region of interest, and (iv) signal from a region centered in the white matter. Each brain is represented as an uncertain graph with 90 nodes corresponding to 90 cerebral regions, excluding 26 cerebellar regions.

• *Attention Deficit Hyperactivity Disorder (ADHD)*: The second dataset is collected from ADHD-200 global competition dataset [5]. The dataset contains records of resting-state fMRI images for 776 subjects, which are labeled as real patients (positive) and normal controls (negative). Similar to the ADNI dataset, the brain images are preprocessed using Athena Pipeline[6]. The original dataset is unbalanced, we randomly sampled 100 ADHD patients and 100 normal controls from the dataset for performance evaluation.

• *Human Immunodeficiency Virus Infection (HIV)*: The third dataset is collected from the Chicago Early HIV Infection Study in Northwestern University [21]. The dataset contains fMRI brain images of patients with early HIV infection (positive) as well as normal controls (negative). The same preprocessing steps as in ADNI dataset were used to extract a functional connectivity network from each image.

4.2 Comparative Methods We compared our method using different statistical measures and discrimination score functions summarized as follows:

Table 3: Summary of experimental datasets.

	$|\widetilde{\mathcal{D}}|$	$|\widetilde{\mathcal{D}}_+|$	$|\widetilde{\mathcal{D}}_-|$	$|V|$	avg. $|E|$	avg. edge prob
ADHD	200	100	100	116	484.7	0.55
ADNI	36	18	18	90	2019.8	0.59
HIV	50	25	25	90	480.48	0.88

• *Frequent Subgraphs + Expectation (Exp+Freq)*: The first baseline method is finding frequent subgraph features within uncertain graphs. This baseline is similar to the method introduced in [27]. In our data model, this baseline method computes the exact expected frequency of each subgraph features, instead of approximated values. The top ranked frequent patterns are extracted as used as features for graph classification.

• DUG *with HSIC based discrimination scores*: we compare with four different versions of our DUG method based upon HSIC criterion, which maximize the dependence between subgraph features and graph labels [13]. "Exp-HSIC" computes the expected HSIC value for each subgraph feature, and find the top-k subgraphs with the largest values. "Med-HSIC" computes the median HSIC value for each subgraph feature, while "Mod-HSIC" computes the mode HSIC value. "φ Pr-HSIC" computes the φ-probability of HSIC value for each subgraph feature.

• DUG *with Frequency Ratio based discrimination scores*: we also compare our method based upon Frequency Ratio, *i.e.*, "Exp-Ratio", "Med-Ratio", "Mod-Ratio" and "φ Pr-Ratio".

• DUG *with G-test based discrimination scores*: we then compare our method based upon G-test criterion, *i.e.*, "Exp-Gtest", "Med-Gtest", "Mod-Gtest" and "φ Pr-Gtest".

• DUG *with Confidence based discrimination scores*: the 5th group of methods are based upon G-test criterion, *i.e.*, "Exp-Conf", "Med-Conf", "Mod-Conf" and "φ Pr-Conf".

• *Simple Thresholding*: Another group methods we have compared are the feature selection methods for certain graphs. In order to get the certain graphs from the uncertain graphs in the dataset, we perform simple tresholding over the weights of the links to get the binary links. These baseline methods include: "*Freq*", "*HISC*", "*Ratio*", "*Gtest*" and "*Conf*", which correspond to the discrimination scores used in previous 5 groups separately.

LibSVM [3] with the linear kernel is used as the base classifier for all compared methods. The min_sup in the gSpan for ADHD, ADNI and HIV datasets are 20%, 40% and 40% respectively. Since the range of different discrimination functions can be extremely different. We set the default φ for HSIC criterion, G-test score,

Table 4: Results on the ADNI (Alzheimer's Disease) dataset with different number of features($t = 100, \cdots, 500$). The results are reported as "average performance + (rank)".

| | Error Rate ↓ | | | | | F1 ↑ | | | | | Avg. |
Methods	$t=100$	$t=200$	$t=300$	$t=400$	$t=500$	$t=100$	$t=200$	$t=300$	$t=400$	$t=500$	Rank
Exp-HSIC	0.400 (9)	0.367 (8)	0.367 (10)	0.317 (4)	0.333 (9)	0.699 (9)	0.725 (9)	0.725 (9)	0.753 (6)	0.743 (10)	(8.3)
Med-HSIC	0.433 (14)	0.350 (5)	0.333 (6)	0.350 (8)	0.317 (7)	0.667 (13)	0.741 (7)	0.757 (4)	0.734 (9)	0.766 (7)	(8.0)
Mod-HSIC	0.367 (6)	0.333 (3)	0.300 (1)*	0.317 (4)	0.300 (2)	0.703 (8)	0.750 (4)	0.776 (3)	0.766 (3)	0.775 (4)	(3.8)
φPr-HSIC	0.283 (1)*	0.283 (1)*	0.333 (6)	0.333 (7)	0.300 (2)	0.778 (1)*	0.785 (1)*	0.757 (4)	0.750 (7)	0.776 (3)	(3.3)
HSIC	0.450 (16)	0.467 (19)	0.467 (17)	0.500 (18)	0.500 (18)	0.615 (18)	0.597 (19)	0.622 (17)	0.583 (18)	0.584 (18)	(18.1)
Exp-Ratio	0.433 (10)	0.383 (10)	0.317 (4)	0.300 (2)	0.300 (2)	0.667 (13)	0.715 (10)	0.756 (6)	0.766 (3)	0.766 (7)	(7.1)
Med-Ratio	0.450 (16)	0.417 (15)	0.450 (16)	0.383 (11)	0.383 (11)	0.639 (17)	0.653 (16)	0.608 (20)	0.689 (12)	0.684 (11)	(14.5)
Mod-Ratio	0.317 (3)	0.350 (5)	0.433 (15)	0.417 (13)	0.467 (15)	0.776 (2)	0.744 (6)	0.659 (13)	0.657 (13)	0.612 (15)	(9.9)
φPr-Ratio	0.400 (9)	0.317 (2)	0.300 (1)*	0.300 (2)	0.267 (1)*	0.692 (10)	0.764 (2)	0.784 (1)*	0.778 (2)	0.809 (1)*	(3.1)
Ratio	0.500 (19)	0.483 (20)	0.533 (22)	0.567 (22)	0.533 (20)	0.581 (20)	0.603 (18)	0.533 (21)	0.519 (22)	0.550 (20)	(20.4)
Exp-Gtest	0.300 (2)	0.367 (8)	0.317 (4)	0.350 (8)	0.383 (11)	0.774 (3)	0.693 (11)	0.729 (9)	0.702 (10)	0.672 (12)	(7.8)
Med-Gtest	0.517 (21)	0.450 (18)	0.400 (11)	0.500 (18)	0.483 (17)	0.562 (21)	0.597 (19)	0.655 (14)	0.567 (19)	0.589 (17)	(17.5)
Mod-Gtest	0.517 (21)	0.550 (22)	0.500 (21)	0.500 (18)	0.517 (19)	0.531 (22)	0.491 (22)	0.527 (22)	0.545 (20)	0.558 (19)	(20.6)
φPr-Gtest	0.450 (16)	0.417 (15)	0.417 (13)	0.383 (11)	0.300 (2)	0.648 (16)	0.675 (14)	0.665 (12)	0.701 (11)	0.768 (6)	(11.6)
Gtest	0.500 (19)	0.500 (21)	0.467 (17)	0.433 (14)	0.550 (21)	0.583 (19)	0.580 (21)	0.612 (19)	0.656 (14)	0.547 (21)	(18.6)
Exp-Conf	0.367 (7)	0.333 (3)	0.300 (1)*	0.283 (1)*	0.300 (2)	0.744 (6)	0.762 (3)	0.780 (2)	0.795 (1)*	0.780 (2)	*(2.8)
Med-Conf	0.333 (4)	0.350 (5)	0.350 (8)	0.350 (8)	0.317 (7)	0.760 (4)	0.747 (5)	0.752 (7)	0.740 (8)	0.770 (5)	(6.1)
Mod-Conf	0.417 (12)	0.383 (10)	0.350 (8)	0.317 (4)	0.333 (9)	0.690 (11)	0.728 (8)	0.742 (8)	0.759 (5)	0.750 (9)	(8.4)
φPr-Conf	0.400 (9)	0.417 (15)	0.467 (17)	0.467 (16)	0.433 (13)	0.685 (12)	0.648 (17)	0.619 (18)	0.592 (17)	0.632 (13)	(14.7)
Conf	0.400 (9)	0.400 (13)	0.417 (13)	0.450 (15)	0.467 (15)	0.655 (15)	0.667 (15)	0.645 (15)	0.618 (15)	0.610 (16)	(14.1)
Exp-Freq	0.383 (8)	0.383 (10)	0.400 (11)	0.467 (16)	0.433 (13)	0.705 (7)	0.685 (13)	0.675 (11)	0.607 (16)	0.632 (13)	(11.8)
Freq	0.350 (5)	0.400 (13)	0.483 (20)	0.550 (21)	0.550 (21)	0.747 (5)	0.692 (12)	0.627 (16)	0.539 (21)	0.547 (21)	(15.5)

Table 5: Results on the ADHD (Attention Deficit Hyperactivity Disorder) dataset with different number of features ($t = 100, \cdots, 500$). The results are reported as "average performance + (rank)".

| | Error Rate ↓ | | | | | F1 ↑ | | | | | Avg. |
Methods	$t=100$	$t=200$	$t=300$	$t=400$	$t=500$	$t=100$	$t=200$	$t=300$	$t=400$	$t=500$	Rank
Exp-HSIC	0.423 (10)	0.438 (13)	0.455 (14)	0.455 (11)	0.448(12)	0.593 (10)	0.564 (13)	0.543 (14)	0.547 (11)	0.549 (12)	(12.0)
Med-HSIC	0.420 (9)	0.405 (8)	0.413 (8)	0.448 (10)	0.433 (6)	0.569 (13)	0.597 (7)	0.593 (5)	0.549 (10)	0.562 (7)	(8.3)
Mod-HSIC	0.390 (4)	0.405 (8)	0.403 (4)	0.393 (1)*	0.410 (2)	0.614 (3)	0.599 (6)	0.596 (4)	0.594 (1)*	0.584 (2)	*(3.5)
φPr-HSIC	0.432 (12)	0.470 (17)	0.475 (16)	0.513 (22)	0.503 (21)	0.597 (7)	0.563 (14)	0.554 (13)	0.508 (17)	0.525 (18)	(15.7)
HSIC	0.529 (22)	0.510 (20)	0.488 (17)	0.455 (11)	0.485 (17)	0.505 (22)	0.494(18)	0.498 (18)	0.538 (13)	0.526 (17)	(17.5)
Exp-Ratio	0.388 (3)	0.400 (5)	0.415 (10)	0.440 (8)	0.420 (4)	0.613 (4)	0.604 (5)	0.587 (9)	0.556 (8)	0.576 (4)	(6.0)
Med-Ratio	0.450 (16)	0.418 (11)	0.388 (1)*	0.428 (6)	0.410 (2)	0.554 (15)	0.586 (12)	0.619 (1)*	0.571 (5)	0.579 (3)	(7.2)
Mod-Ratio	0.400 (7)	0.370 (1)*	0.408 (5)	0.435 (7)	0.428 (5)	0.595 (8)	0.634 (1)*	0.591 (7)	0.558 (7)	0.560 (9)	(5.7)
φPr-Ratio	0.372 (1)*	0.430 (12)	0.410 (7)	0.415 (2)	0.408 (1)*	0.630 (1)*	0.589 (9)	0.590 (8)	0.591 (2)	0.589 (1)*	(4.4)
Ratio	0.515 (20)	0.520 (21)	0.490 (18)	0.475 (17)	0.498 (19)	0.550 (16)	0.461 (22)	0.461 (21)	0.503 (19)	0.517 (20)	(19.3)
Exp-Gtest	0.393 (6)	0.403 (7)	0.413 (8)	0.420 (3)	0.435 (9)	0.610 (5)	0.588 (10)	0.582 (10)	0.586 (3)	0.563 (5)	(6.6)
Med-Gtest	0.437 (13)	0.400 (5)	0.408 (5)	0.420 (3)	0.453 (15)	0.559 (14)	0.590 (8)	0.600 (2)	0.580 (4)	0.551 (10)	(7.9)
Mod-Gtest	0.448 (15)	0.383 (4)	0.398 (2)	0.428 (5)	0.433 (6)	0.571 (12)	0.622 (4)	0.593 (5)	0.565 (6)	0.551 (10)	(6.9)
φPr-Gtest	0.450 (16)	0.445 (14)	0.443(12)	0.455 (11)	0.433 (6)	0.544 (19)	0.555 (16)	0.552 (12)	0.538 (13)	0.562 (7)	(12.6)
Gtest	0.440 (14)	0.505 (19)	0.501 (21)	0.486 (19)	0.471 (16)	0.542 (20)	0.492 (19)	0.490 (20)	0.499 (21)	0.534 (16)	(18.5)
Exp-Conf	0.405 (10)	0.415 (10)	0.453 (13)	0.455 (11)	0.448 (12)	0.595 (8)	0.587 (11)	0.539 (15)	0.543 (12)	0.535 (15)	(11.5)
Med-Conf	0.378 (2)	0.373 (2)	0.438 (11)	0.463 (15)	0.435 (9)	0.629 (2)	0.632 (2)	0.555 (11)	0.536 (15)	0.545 (13)	(8.2)
Mod-Conf	0.392 (5)	0.373 (2)	0.400 (3)	0.440 (8)	0.435 (9)	0.606 (6)	0.627 (3)	0.600 (2)	0.556 (8)	0.563 (5)	(5.1)
φPr-Conf	0.468 (19)	0.460 (15)	0.495 (20)	0.505 (21)	0.485 (17)	0.547 (18)	0.556 (15)	0.519 (16)	0.507 (18)	0.540 (14)	(17.3)
Conf	0.455 (18)	0.500 (18)	0.460 (15)	0.464 (16)	0.450 (14)	0.514 (21)	0.479 (20)	0.510 (17)	0.498 (22)	0.519 (19)	(18.0)
Exp-Freq	0.423 (10)	0.465 (16)	0.508 (22)	0.498 (20)	0.505 (22)	0.579 (11)	0.549 (17)	0.496 (19)	0.513 (16)	0.498 (22)	(17.5)
Freq	0.515 (20)	0.520 (21)	0.490 (18)	0.475 (17)	0.498 (19)	0.550 (16)	0.461 (21)	0.461 (21)	0.503 (19)	0.517 (20)	(19.2)

frequency ratio and confidence as 0.03, 200, 1 and 0.5, respectively.

4.3 Performance on Uncertain Graph Classification In our experiments, we first randomly sample 80% of the uncertain graphs as the training set, and the remaining graphs as the test set. This random sampling experiment was repeated 20 times. The average performances with the rank of each method are reported. The reason for using classification performances to evaluate the quality of subgraph features is that classification methods can usually achieve higher accuracy with features of better discriminative powers. We measure the classification performance by error rate and F1 score.

Table 5 and Table 4 show the evaluation results in terms of classification error rates and F1 scores with different number of selected subgraph features ($t = 100, \cdots, 500$). The results of each method are shown with average performance values and their ranks among all the other methods. Values with ∗ stand for the best performance for the corresponding evaluation criterion. It is worth noting that the neuroimaging tasks are generally very hard to predict very accurately. According to a global competition on ADHD dataset[7], the average performance of all winning teams is about 8% over the prediction accuracy of chance (i.e., randomly assigning diagnoses). Thus in neuroimaging tasks, it is

[7]http://www.childmind.org/en/posts/press-releases/2011-10-12-johns-hopkins-team-wins-adhd-200-competition

Table 6: Results on the HIV (Human Immunodeficiency Virus) dataset with different number of features $(t = 100, \cdots, 500)$. The results are reported as "average performance + (rank)".

Methods	Error Rate ↓					F1 ↑					Avg. Rank
	$t=100$	$t=200$	$t=300$	$t=400$	$t=500$	$t=100$	$t=200$	$t=300$	$t=400$	$t=500$	
Exp-HSIC	0.480 (15)	0.470 (10)	0.489 (12)	0.505 (16)	0.498 (13)	0.526 (13)	0.531 (8)	0.517 (11)	0.491 (14)	0.492 (13)	(12.5)
Med-HSIC	0.498 (17)	0.500 (18)	0.470 (7)	0.484 (11)	0.507 (16)	0.501 (18)	0.493 (18)	0.526 (8)	0.510 (10)	0.474 (16)	(13.9)
Mod-HSIC	0.502 (18)	0.489 (15)	0.482 (11)	0.498 (14)	0.500 (14)	0.501 (18)	0.501 (16)	0.495 (14)	0.481 (17)	0.467 (19)	(15.6)
φPr-HSIC	0.523 (19)	0.511 (19)	0.516 (18)	0.525 (19)	0.523 (20)	0.484 (20)	0.492 (19)	0.481 (16)	0.474 (19)	0.482 (14)	(18.3)
HSIC	0.464 (6)	0.495 (17)	0.566 (21)	0.500 (15)	0.505 (15)	0.526 (13)	0.460 (20)	0.405 (21)	0.489 (15)	0.471 (18)	(16.1)
Exp-Ratio	0.475 (13)	0.477 (11)	0.491 (13)	0.516 (18)	0.484 (8)	0.541 (8)	0.533 (7)	0.509 (13)	0.477 (18)	0.519 (8)	(11.3)
Med-Ratio	0.466 (8)	0.464 (8)	0.470 (7)	0.457 (5)	0.473 (6)	0.541 (8)	0.528 (9)	0.524 (9)	0.534 (6)	0.521 (6)	(7.2)
Mod-Ratio	0.450 (3)	0.452 (5)	0.466 (4)	0.480 (9)	0.484 (8)	0.558 (5)	0.547 (5)	0.528 (6)	0.509 (11)	0.500 (12)	(6.8)
φPr-Ratio	0.473 (11)	0.480 (12)	0.466 (4)	0.470 (8)	0.468 (5)	0.544 (7)	0.519 (13)	0.538 (5)	0.531 (7)	0.538 (5)	(7.7)
Ratio	0.530 (21)	0.486 (13)	0.589 (22)	0.411 (1)*	0.520 (19)	0.456 (21)	0.495 (17)	0.376 (22)	0.562 (4)	0.443 (20)	(16)
Exp-Gtest	0.468 (9)	0.466 (9)	0.468 (6)	0.466 (7)	0.482 (7)	0.562 (4)	0.565 (4)	0.548 (4)	0.537 (5)	0.520 (7)	(6.2)
Med-Gtest	0.464 (6)	0.461 (7)	0.507 (17)	0.507 (17)	0.511 (17)	0.534 (11)	0.520 (11)	0.480 (17)	0.483 (16)	0.474 (16)	(10.9)
Mod-Gtest	0.477 (14)	0.486 (13)	0.475 (10)	0.491 (13)	0.489 (11)	0.529 (12)	0.507 (14)	0.523 (10)	0.497 (13)	0.501 (11)	(12.1)
φPr-Gtest	0.430 (1)*	0.420 (2)	0.425 (1)*	0.418 (2)	0.425 (2)	0.617 (1)*	0.633 (1)*	0.630 (1)*	0.637 (1)*	0.633 (1)*	*(1.3)
Gtest	0.473 (11)	0.550 (21)	0.493 (14)	0.534 (20)	0.493 (12)	0.514 (16)	0.426 (22)	0.491 (15)	0.509 (11)	0.477 (15)	(15.7)
Exp-Conf	0.457 (4)	0.430 (4)	0.441 (2)	0.443 (4)	0.441 (3)	0.576 (3)	0.590 (2)	0.572 (2)	0.570 (3)	0.573 (4)	(3.1)
Med-Conf	0.445 (2)	0.427 (3)	0.441 (2)	0.441 (3)	0.443 (4)	0.579 (2)	0.588 (3)	0.572 (2)	0.579 (2)	0.574 (3)	(2.6)
Mod-Conf	0.457 (4)	0.455 (6)	0.473 (9)	0.482 (10)	0.484 (8)	0.556 (6)	0.545 (6)	0.527 (7)	0.518 (9)	0.508 (9)	(7.4)
φPr-Conf	0.534 (22)	0.552 (22)	0.545 (19)	0.548 (21)	0.541 (22)	0.454 (22)	0.443 (21)	0.444 (20)	0.443 (22)	0.438 (21)	(21.2)
Conf	0.468 (9)	0.416 (1)*	0.502 (15)	0.489 (12)	0.339 (1)*	0.515 (15)	0.528 (9)	0.468 (19)	0.462 (20)	0.621 (2)	(10.3)
Exp-Freq	0.525 (20)	0.520 (20)	0.548 (20)	0.550 (22)	0.527 (21)	0.503 (17)	0.520 (11)	0.473 (18)	0.457 (21)	0.423 (22)	(19.2)
Freq	0.489 (16)	0.489 (15)	0.502 (15)	0.461 (6)	0.514 (18)	0.535 (10)	0.505 (15)	0.517 (11)	0.520 (8)	0.502 (10)	(12.4)

very hard for classification algorithms to achieve even moderate error rates. And in ADHD dataset, the best performance that DUG can achieve is with error rate 37%, which is 13% improvement over the prediction error rate of chance.

We find that our discriminative subgraph mining method with different settings outperforms the baseline method (Exp-Freq) for frequent subgraph mining, which selects subgraph features based upon expected frequencies in the uncertain graph dataset. This is because that frequent subgraph features in uncertain graph dataset may not be relevant to the classification task.

Moreover, we can see that almost all the DUG methods outperform the simple thresholding methods which directly convert the uncertain graphs into certain graphs and then use different discimination functions to select subgraph features. This is because that simply converting uncertain graphs into certain graph can loss the uncertainty information about the linkage structures of the graphs, thus the classification performances on certain graphs are not as good as the performance of uncertain graphs.

A third observation is that the performance of each method on different dataset can be quite different. However, the best methods that consistently outperforms other methods in all datasets are Med-Conf and φ-Pr-Ratio. They both have their advantages in different perspectives. Med-Conf method has one less parameter than that of φ-Pr-Ratio. φ-Pr-Ratio method has an additional subgraph pruning bound compared to Med-Conf method, which can be important for datasets with even larger graphs.

4.4 Influence of Parameter In the φ-Pr based methods, there is an additional threshold parameter than the other methods. In Figure 7(a) and Figure 7(b), we tested the φ-Pr-HSIC with φ values among $\{0.01, 0.02, \cdots 0.06\}$ separately. We can see that the method is not sensitive to the parameter φ. Generally, the performance of φ-Pr-HSIC with default setting ($\varphi = 0.03$) is pretty good. If we try to optimize the selection of φ value, the accuracy can be even better.

We also compare DUG models with and without pruning in the subgraph search space as summarized in Figure 7(c). The CPU time with different min_sup for Exp-HSIC in ADNI dataset is reported. DUG can improve the efficiencies by pruning the subgraph search space. In other datasets DUG shows similar trends. Figure 8 shows the running time for mod-HISC with different number of graphs in the dataset. In addition to the dynamic programming method we used in DUG, we also find that the brute-force searching method that enumerates all possible worlds of the uncertain graph dataset cannot work on small datasets with even 40 graphs. The running time of DUG scales almost linearly with the number of graphs in the dataset. Althought the dynamic programming process of DUG is $O(n^2)$, which is quadratic to the number of graphs in the dataset. However, in the ADHD dataset, the main computational cost of DUG algorithm is for the subgraph enumeration step, which is linear to the number of the graphs in the dataset. In cases of even larger datasets, the dynamic programming process could eventually dominant the computational cost. In these cases, the divide-and-conquer method in [19] could be used to further optimize the computational cost.
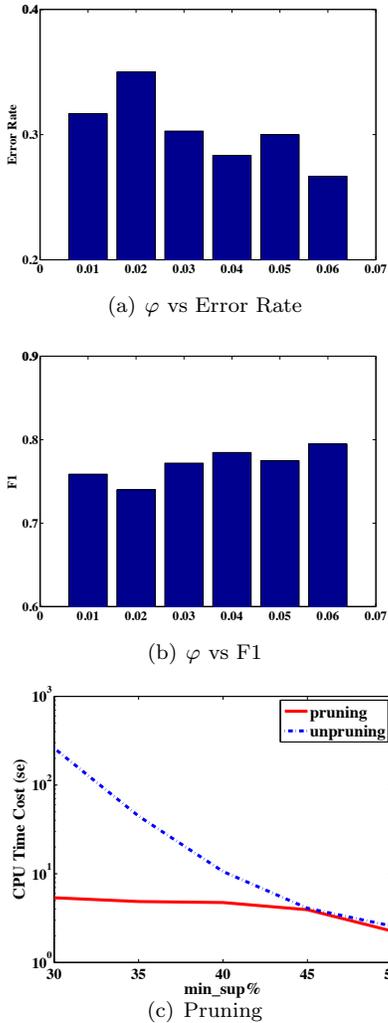
(a) φ vs Error Rate

(b) φ vs F1

(c) Pruning

Figure 7: Parameter Studies (ADNI dataset).

Figure 8: Running time on ADNI dataset.

5 Related Work

Our work is related to subgraph mining techniques for both certain graphs and uncertain graphs. We briefly discuss both of them.

Mining subgraph features in graph data has been studied intensively in recent years [15]. Most of the previous research has been focused on certain graph datasets, where the edges of the graph objects are binary/certain. The aim of these subgraph mining method is to extract important subgraph features based on the structure of the graphs. Depending on whether the class labels are considered in the feature mining steps, existing methods can roughly be categorized into two types: frequent subgraph mining and discriminative subgraph mining. In frequent subgraph mining, Yan and Han proposed a depth-first search algorithm, gSpan
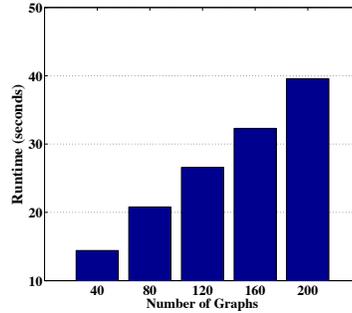
[23], which maps each graph to a unique minimum DFS code and use right-most extension technique for subgraph extension. There are also many other frequent subgraph mining methods that have been proposed, e.g., AGM [9], FSG [15], MoFa [1], and Gaston [16], etc. Discriminative subgraph mining have also been studied intensively in the literature, such as LEAP [22] and LTS [10], where the task is to find discriminative subgraph for graph classifications.

Recently, there has been a growing interest in exploiting data uncertainty, especially structural uncertainty in graph data. There are some recent works on mining frequent subgraph features for uncertain graphs [27, 26, 28, 17]. The problem of mining frequent subgraph in uncertain graphs are more difficult to those of certain graphs. The authors [27] proposed a method to estimate approximately the expected support of a subgraph feature in uncertain graph datasets. In [26], the authors studies the φ-probabilities for frequent subgraph features within uncertain graph datasets. The difference between these works and our paper are as follows: 1) In this paper, we study how to find discriminative subgraph features for uncertain graph data. The class labels of the graph objects are considered during the subgraph mining. 2) The graph model in our paper is different from previous uncertain graph data, since we assume different graph object shares the same set of nodes as inspired by the neuroimaging applications. Thus, our method compute the exact discrimination scores of each subgraph features, instead of approximate scores. There are also many other works on uncertain graphs, which focus on different problems, e.g., reliable subgraph mining [12], k-nearest neighbor discovery [18], subgraph retrieval [24] etc.

Our work is also motivated by the recent advances in analyzing neuroimaging data using data mining and machine learning approaches [6, 8, 7, 25]. Huang et. al. [6] developed a sparse inverse covariance estimation method for analyzing brain connectivities in

PET images of patients with Alzheimer's disease.

6 Conclusion

In this paper, we studied the problem of discriminative subgraph feature selection for uncertain graph classification. We proposed a general framework, called DUG, for finding discriminative subgraph feature in uncertain graphs based upon various statistical measures. The probability distributions of the scoring function are efficiently computed based on dynamic programming.

References

[1] C. Borgelt and M. Berthold. Mining molecular fragments: Finding relevant substructures of molecules. In *ICDM*, pages 211–218, 2002.

[2] E. Bullmore and O. Sporns. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience*, 10(3):186–198, 2009.

[3] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[4] H. Cheng, D. Lo, Y. Zhou, X. Wang, and X. Yan. Identifying bug signatures using discrimative graph mining. In *ISSTA*, pages 141–152, 2009.

[5] C. Gao and J. Wang. Direct mining of discriminative patterns for classifying uncertain data. In *KDD*, pages 861–870, 2010.

[6] S. Huang, J. Li, L. Sun, J. Ye, K. Chen, and T. Wu. Learning brain connectivity of azheimer's disease from neuroimaging data. In *NIPS*, pages 808–816, 2009.

[7] S. Huang, J. Li, J. Ye, A. Fleisher, K. Chen, T. Wu, and E. Reiman. Brain effective connectivity modeling for alzheimer's disease by sparse gaussian bayesian network. In *KDD*, pages 931–939, 2011.

[8] S. Huang, J. Li, J. Ye, A. Fleisher, T. Wu, K. Chen, and E. Reiman. Learning brain connectivity of alzheimer's disease by exploratory graphical models. *NeuroImage*, 50:935–949, 2010.

[9] A. Inokuchi, T. Washio, and H. Motoda. An apriori-based algorithm for mining frequent substructures from graph data. In *ECML/PKDD*, pages 13–23, 2000.

[10] N. Jin and W. Wang. LTS: Discriminative subgraph mining by learning from search history. In *ICDE*, pages 207–218, 2011.

[11] N. Jin, C. Young, and W. Wang. GAIA: graph classification using evolutionary computation. In *SIGMOD*, pages 879–890, 2010.

[12] R. Jin, L. Liu, and C. Aggarwal. Discovering highly reliable subgraphs in uncertain graphs. In *KDD*, pages 992–1000, 2011.

[13] X. Kong, W. Fan, and P. S. Yu. Dual active feature and sample selection for graph classification. In *KDD*, pages 654–662, 2011.

[14] X. Kong and P. S. Yu. Semi-supervised feature selection for graph classification. In *KDD*, pages 793–802, 2010.

[15] M. Kuramochi and G. Karypis. Frequent subgraph discovery. In *ICDM*, pages 313–320, 2001.

[16] S. Nijssen and J. Kok. A quickstart in frequent structure mining can make a difference. In *KDD*, pages 647–652, 2004.

[17] O. Papapetrou, E. Ioannou, and D. Skoutas. Efficient discovery of frequent subgraph patterns in uncertain graph databases. In *EDBT*, pages 355–366, 2011.

[18] M. Potamias, F. Bonchi, A. Gionis, and G. Kollios. k-nearest neighbors in uncertain graphs. In *VLDB*, pages 997–1008, 2010.

[19] L. Sun, R. Cheng, D. Cheung, and J. Cheng. Mining uncertain data with probabilistic guarantees. In *KDD*, pages 273–282, 2010.

[20] M. Thoma, H. Cheng, A. Gretton, J. Han, H. Kriegel, A. Smola, L. Song, P. Yu, X. Yan, and K. Borgwardt. Near-optimal supervised feature selection among frequent subgraphs. In *SDM*, pages 1075–1086, 2009.

[21] X. Wang, P. Foryt, R. Ochs, J. Chung, Y. Wu, T. Parrish, and A. Ragin. Abnormalities in resting-state functional connectivity in early human immunodeficiency virus infection. *Brain Connectivity*, 1(3):207, 2011.

[22] X. Yan, H. Cheng, J. Han, and P. Yu. Mining significant graph patterns by leap search. In *SIGMOD*, pages 433–444, 2008.

[23] X. Yan and J. Han. gSpan: Graph-based substructure pattern mining. In *ICDM*, pages 721–724, 2002.

[24] Y. Yuan, G. Wang, H. Wang, and L. Chen. Efficient subgraph search over large uncertain graphs. In *VLDB*, pages 876–886, 2011.

[25] J. Zhou, L. Yuan, J. Liu, and J. Ye. A multi-task learning formulation for predicting disease progression. In *KDD*, pages 814–822, 2011.

[26] Z. Zou, H. Gao, and J. Li. Discovering frequent subgraphs over uncertain graph databases under probabilistic semantics. In *KDD*, pages 633–642, 2010.

[27] Z. Zou, J. Li, H. Gao, and S. Zhang. Frequent subgraph pattern mining on uncertain graph data. In *CIKM*, pages 583–592, 2009.

[28] Z. Zou, J. Li, H. Gao, and S. Zhang. Mining frequent subgraph patterns from uncertain graph data. *IEEE Transactions on Knowledge and Data Engineering*, 22(9):1203–1218, 2010.