PSM-Flow: Probabilistic Subgraph Mining for Discovering Reusable Fragments in Workflows

Ken Cheong
CS Department
HK Baptist University
Hong Kong

Daniel Garijo
Information Sciences Institute
U. of Southern California
USA

William K. Cheung
CS Department
HK Baptist University
Hong Kong

Yolanda Gil
Information Sciences Institute
U. of Southern California
USA

***Abstract*—Scientific workflows define computational processes needed for carrying out scientific experiments. Existing workflow repositories contain hundreds of scientific workflows, where scientists can find materials and knowledge to facilitate workflow design for running related experiments. Identifying reusable fragments in growing workflow repositories has become increasingly important. In this paper, we present PSM-Flow, a probabilistic subgraph mining algorithm designed to discover commonly occurring fragments in a workflow corpus using a modified version of the Latent Dirichlet Allocation algorithm. The proposed model encodes the geodesic distance between workflow steps into the model for implicitly modeling fragments. PSM-Flow captures variations of frequent fragments while maintaining its space complexity bounded polynomially, as it requires no candidate generation. We applied PSM-Flow to three real-world scientific workflow datasets containing more than 750 workflows for neuroimaging analysis. Our results show that PSM-Flow outperforms three state of the art frequent subgraph mining techniques. We also discuss other potential future improvements of the proposed method.**

I. Introduction

Scientific workflows describe computational experiments which typically involve computational steps, along with the datasets used and generated by those steps. Scientific workflows are created in workflow systems that manage their execution with the required computational resources [13]. Representing workflows explicitly improves the reproducibility of scientific experiments [12].

Scientific workflow repositories contain collections of recorded scientific workflows. [23] Users may explore and reuse workflows created by others to facilitate the development of their computational experiments. While one can directly reuse an existing workflow, only a portion or fragment of a workflow is often reused. In addition, identifying commonly used fragments of workflows facilitates overviewing and exploring the contents of a workflow repository. [11]

In [10], the authors formulated reusable workflow fragment identification as a frequent subgraph mining problem and applied frequent subgraph mining algorithms (FSM) to detect the subgraphs with high support count (number of occurrences) as candidate fragments. However, frequent subgraph mining techniques present several limitations. First, these techniques typically involve a candidate fragment generation process and a subgraph isomorphism test. Both present a time complexity (combinatorial exploration of candidate fragments) and a space complexity (a large number of candidate subgraphs are generated in memory) that are exponential in the worst case. Second, frequent fragments may appear in different workflows with small variations (e.g., with changes in node labels, or with an additional node). Conventional frequent subgraph mining techniques use exact matching for counting fragment occurrences, and thus do not take those variations into account. Adopting stochastic models for fragment discovery can implicitly group structurally similar subgraphs together for more robust results. In other words, stocastic models create a higher level abstraction for grouping fragments based on their commonality which makes the discovery of infrequent but potentially useful fragments (e.g., similar to those frequent fragments) possible.

In this paper we propose PSM-Flow (Probabilistic Subgraph Mining for Reusable Workflow Fragment Identification) a topic modeling approach that modifies the Latent Dirichlet Allocation (LDA) algorithm so that those latent topics to be inferred correspond to different groups of workflow fragments. Specifically, we encode the geodesic distance of nodes in workflows into LDA and introduce a soft constraint into the model formulation so that closer nodes will have a higher probability to share the same topic label. Therefore, a topic is represented by a probability distribution of node labels in which the node labels co-occur not only frequently but also closely. Under our formulation, a topic can be interpreted as a kind of abstract subgraph or a cluster of subgraphs in which the subgraphs are structurally similar to each other. In other words, stochastic variations of subgraphs are captured.

When compared to most of frequent subgraph mining techniques, which take exponential space in the worst case, PSM-Flow has the advantage that the required space is polynomially bounded by O(N^2) where N is the number of nodes (computational steps) in the workflow corpus.

We evaluate our approach using three workflow corpora created by scientists using the LONI Pipeline workflow system [6] for neuroimaging analysis. We compare the quality of fragments extracted using PSM-Flow to the three different subgraph mining algorithms (gSpan [26] , SUBDUE [4] and READUM[19]) by measuring whether the extracted fragments can capture sub-workflows defined by users. Promising results are obtained.

The remainder of the paper is organized as follows. First,

we introduce related work on workflow fragment mining and topic modeling in Section 2. Section 3 presents the problem formulation along with details about PSM-Flow. The experiment setup and our evaluation results can be found in Section 4. Section 5 concludes the paper with a discussion of future lines of work.

II. Related Work

Scientific workflow reuse is common when workflows are shared in a repository. Garijo et al.(2012) [9] report that at least 20% of the analyzed workflows from different workflow systems and domains were composed of other workflows. Other efforts [24] confirm this practice in community workflow repositories, such as MyExperiment [23].

Frequent subgraph mining algorithms have been used to automatically detect common workflow fragments in a corpus of workflows. Workflows are represented as a graph where the nodes represent computational steps and the edges correspond to the dataflow among them. Garijo et al.(2014) [10] proposed the FragFlow framework, which makes use of two graph-based frequent subgraph mining algorithms, namely gSpan [26] and SUBDUE [4]. Other work has also applied SUBDUE to workflow corpora [5]. In general, frequent subgraph mining aims to extract frequent substructures in a set of graphs. It usually consists of two steps: candidate subgraph generation and subgraph isomorphism detection. Existing algorithms can be categorized into two types - exact and inexact match. Exact match algorithms extract all frequent subgraphs in a data set. Most of them perform efficiently only on sparse graphs with a large number of labels for nodes and edges [18]. But there also exist some efficient exact match algorithms such as FSG [17], GASTON [21], and gSpan [26]. In contrast, inexact match algorithms consider the similarity between subgraphs and allow a subgraph in the corpus to match a given candidate subgraph even though they have slight structural differences. Examples of inexact match algorithms include SUBDUE [4], GREW [18], gApprox [3] and REAFUM. [19]

In this paper, we formulate the workflow fragment discovery problem based on Latent Dirichlet Allocation (LDA) [2] which is a popular model used for topic discovery. LDA was first proposed for text documents, where each document is modeled as a mixture of topics. A topic is represented as a multinomial distribution over word labels. Words that have a high probability mass in a topic tend to co-occur frequently in different documents. Griffiths and Steyvers (2004) applied Gibbs sampling for learning LDA. The number of topics may be determined automatically by using Dirichlet Process (DP), [22] a Bayesian nonparametric tool adopted in the topic modeling community. [25]

A related notion similar to this work is called stochastic network motif. [15] A network motif is formed by patterns of interactions (or subgraph patterns) which appear in different parts of a network more frequently than those found in a randomized network. A stochastic network motif takes the uncertainty of edges into account, where each motif is a subgraph in which edges in the subgraph are assigned with probability values representing the presence of the edges. The resulting patterns turn out to be more robust than deterministic motifs, as rare situations are also captured. Liu, Cheung, and Liu (2015) also generalize the work to multiple motifs detection. Our method is different from stochastic network motif in several aspects. First, stochastic network motif models the absence of edges explicitly while we only encode the geodesic distance between two nodes. In addition, stochastic network motif deals with relational patterns where each node in graph data is a distinct object of the homogeneous type. In our workflow setting, nodes have node labels (name of computational steps) which repeatedly appear in the dataset.

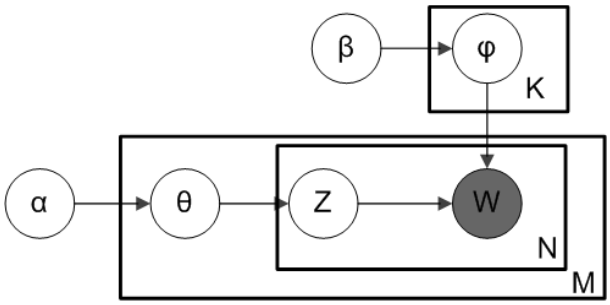


Fig. 1: Graphical representation of the LDA model.

III. Methodology

This section presents the problem formulation and PSM-Flow, our proposed algorithm for the workflow fragment identification. We extend Latent Dirichlet Allocation (LDA) [2] which is a generative model widely used for topic modeling in a text corpus and extend it to model reusable fragments in a workflow corpus.

A. Latent Dirichlet Allocation

The goal of LDA is to extract topics from a text corpus. LDA assumes that each document is generated from a mixture of topics called *topic distribution*, and each topic is represented as a distribution of words called *word distribution*. The graphical representation of LDA is shown in Figure 1. Let θ_d be the topic distribution of document d, ϕ_k the word distribution of topic k, z_n the topic assignment of word instance n,w_n the word label of word instance n, and α and β the hyperparameters for the conjugate priors (Dirichlet) of the multinomial distribution over topics and words respectively. The generative process of LDA is as follows:

1) Draw ϕ_k from Dir(β) for each topic.
2) Draw θ_d from Dir(α) for each document.
3) Draw topic z from Multi(θ_d) for each word instance.
4) Draw word w from Multi(ϕ_k) for each word instance.

where $Dir()$ and $Multi()$ denote the Dirichlet distribution and multinomial distribution respectively. We use Gibbs sampling to infer the model parameters, which is a Markov chain Monte Carlo algorithm for estimating the posterior probability [14].

LDA assumes words in a document given a topic are independent. However, when we move from a text corpus to

a workflow corpus, the node instances in a workflow should not be considered independent and their dependency on usage is in fact related by the node connectivity. We propose a generative model named PSM-Flow for extracting reusable workflow fragments from a workflow corpus.

PSM-Flow extends LDA as follows. We consider a "workflow" as a "document", and the type (label) of a "workflow step" as a "word". Instead of assuming that generation of word instances are independent given the topic as in LDA, PSM-Flow assumes that "nearby" node instances in a workflow tend to have identical topic labels (as detailed in the next section). Therefore, those node instances assigned with the same topic are more likely to be connected as subgraphs in a workflow. We use the word "fragment" to refer to "subgraph" in a workflow. A topic, hence, is a distribution of node labels in which most probability mass is concentrated on those frequently co-occur labels. A fragment F belongs to a topic T means that all topic assignments of node instances from F are equal to topic T.

B. Encoding Geodesic Distance in PSM-Flow

The fundamental idea of PSM-Flow is adding a soft constraint to LDA so that if the distance of the shortest (undirected) path (i.e., geodesic distance) between two nodes in a workflow is smaller, there will be a higher probability for the two nodes share the same topic label. Therefore, nodes sharing a topic will tend to be grouped together as a cluster which corresponds to a fragment. As an illustration, Figure 2 shows a simple example where three workflows have co-occurring nodes with the same labels (indicated by colors). We can then extract all the connected nodes with the same topic labels as the workflow fragments for reuse. Notice that a topic may thus include multiple fragments, as shown in Figure 3. Fragments under the same topic have slight differences, such as labels or topological variations.

In our problem formulation, we incorporate the geodesic distance between nodes only and ignore other topological information of the workflow for simplicity reasons. We argue that this simplification suffices for scientific workflows. For example, let's assume that D1 is a data filtering component and D2 is a data analysis component. It will be natural for D1 to be followed by D2, and it is less likely to have D2 to be followed by D1. At least, it is less likely for the output format of D2 to be consistent with the input format required by D1. In other words, even if the order of the workflow steps is ignored in our model, the effectiveness of the fragment extraction will not be significantly affected.

In the following sections, we present PSM-Flow which integrates the embedding trick into the LDA model. Also, a Gibbs sampling procedure derived for the model learning is then described. Table I depicts the notations used in this paper.

C. PSM-Flow

To encode the constraint defined in Section III-B, we assume that each node in the workflow has a corresponding node embedding which preserves the geodesic distance of nodes.

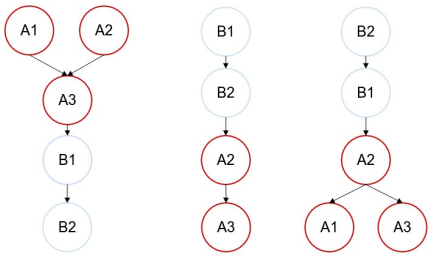


Fig. 2: A synthetic corpus with three workflows. The colors show different topics inferred for the nodes.

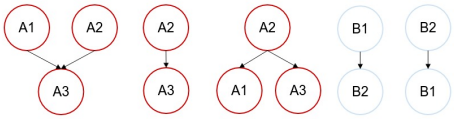


Fig. 3: Fragments extracted from Figure 2. Each color represents a different topic.

We introduce the notion of Gaussian clusters for modeling clustered node embeddings to indicate that they often co-occur as neighbors in the workflow corpus. A Gaussian cluster is a multivariate Gaussian distribution with an associated topic label θ. The cluster generates embeddings for nodes and all nodes in the same Gaussian cluster share the same topic label as the cluster's associated topic label θ. To automatically learn the optimal number of topics in PSM-Flow, we make use of the Dirichlet Process (DP) [7] which has been shown effective for LDA. The overall graphical model of PSM-Flow is shown in Figure 4.

The generative process of PSM-Flow is defined as follow:

1) Choose $\phi_k \sim Dir(\beta)$ where $k \in \{1, ..., \infty\}$ to get node label distribution.
2) Choose $G \sim DP(\tau H)$ where τ is a hyper parameter for the Dirichlet process.

Symbol	Description
D	Number of workflows in the corpus.
W	Number of distinct node labels.
N	Total number of nodes in the corpus.
T	Total number of topics.
M_d	Number of Gaussian clusters in workflow d.
G	Global topic distribution.
H	Base distribution of Dirichlet process.
w_i	Observed node label of node i.
z_i	Topic label of node i.
x_i	Embedding of node i.
c_i	Index of Gaussian cluster that generates node i.
μ_e	Mean of Gaussian cluster e.
θ_e	Topic label of Gaussian cluster e.
C_{wk}^{WT}	Count of node label w assigned to topic k.
C_{dk}^{DT}	Count of topic label k in a workflow d.
ϕ_k	Distribution of node labels for topic k.
D_{ij}	Observed geodesic distance between nodes i and j.

TABLE I: A summary of notations.

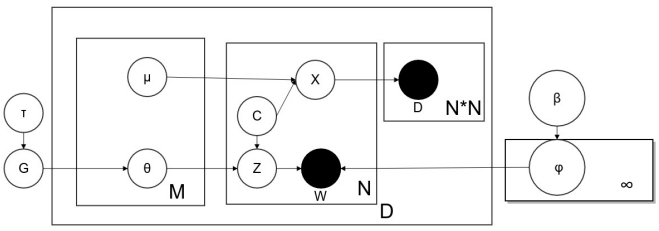


Fig. 4: PSM-Flow Model.

3) For each workflow d, it is assumed that there are M_d Gaussian clusters. For each Gaussian cluster $e \in \{1...M_d\}$, the topic label θ_e is drawn from G. Also, each cluster is associated with a cluster mean μ_e as its parameter.
4) Generate a set of N_d nodes in workflow d. For each node $j \in \{1...N_d\}$, we first draw a Gaussian cluster label c_j from a uniform distribution. We then set the topic label z_j to be the same as the topic label of c_j (*i.e.*, θ_{c_j}), and draw node label w_i from Multi(ϕ_{z_j}). Also, we draw the corresponding embedding $\boldsymbol{x_j}$ from $N(\mu_{c_j}, I)$[1]
5) Among all the embeddings, the observed (geodesic) distance between nodes i and j D_{ij} is drawn from $N(||\boldsymbol{x_i} - \boldsymbol{x_j}||, 1)$ where $||.||$ is the Euclidean distance function.

To learn the model using Gibbs sampling, the posterior distributions of $\boldsymbol{\mu}$, $\boldsymbol{\theta}$, z , c and $\boldsymbol{x}$ are needed, given as:

$$P(\boldsymbol{\mu}_e|\boldsymbol{x}, \boldsymbol{c}) \sim N(\boldsymbol{\mu}_{base}, \boldsymbol{\sigma}_{base})$$
$$\boldsymbol{\mu}_{base} = \sum_j \frac{\boldsymbol{x}_j I(c_j = e)}{n_e}$$
$$\boldsymbol{\sigma}_{base} = I * (n_e + 1)^{-1}$$

where n_e is the number of points in cluster e, and $I(s)$ is an indicator function that gives 1 when s is true, and 0 otherwise.

$$P(c_j = e|w_j, \boldsymbol{x}_j, \boldsymbol{\theta}, \boldsymbol{\mu}) \propto P(w_j|\theta_e) \cdot P(\boldsymbol{x}_j|\boldsymbol{\mu}_e)$$
$$P(w_j|\theta_e) \propto \frac{C^{WT}_{w_j,\theta_e} + \beta}{\sum_{w_j=1}^{W} C^{WT}_{w_j,\theta_e} + W\beta}$$
$$P(\boldsymbol{x}_j|\boldsymbol{\mu}_e) \propto exp(-\frac{1}{2}(\boldsymbol{x}_j - \boldsymbol{\mu}_e)^T(\boldsymbol{x}_j - \boldsymbol{\mu}_e))$$

where $C^{WT}_{w_j,\theta_e}$ is the number of node label w_j assigned to θ_e.

$$P(\theta_i = t|\boldsymbol{w}) \propto \begin{cases} \frac{\tau}{N+\tau} & \text{if } t \text{ is a new topic} \\ \frac{\sum_{w=1}^{W} C^{WT}_{w,t}}{N+\tau} \prod_{\forall j|c_j=i} \frac{C^{WT}_{w_j,t}+\beta}{\sum_{w=1}^{W} C^{WT}_{w,t}+W\beta}, & \end{cases}$$
$$z_j = \theta_{c_j},$$

[1]We adopt the identity matrix to make it computationally more efficient.

and the embedding $\boldsymbol{x}_i$ is sampled according to

$$P(\boldsymbol{x}_i|\boldsymbol{x}_{-i}, \boldsymbol{\mu}_{c_i}, \boldsymbol{D}, ..)$$
$$\propto \quad exp(-\frac{(||\boldsymbol{x}_i - \boldsymbol{\mu}_{c_i}||))}{2} \prod_{j \neq i} exp(-\frac{(||\boldsymbol{x}_j - \boldsymbol{x}_i|| - D_{ij})^2}{2}).$$

In our experiment, we set the number of Gaussian clusters for each workflow as the number of nodes in each workflow. We consider this to be sufficient, as eventually only some of the clusters will be sampled. Also, we need to set the dimension for our node embedding. In general, the higher the dimension, the better the distance information can be preserved but at the expense of the computational complexity. In our experiments, we use 10 as the dimension which is empirically shown to be effective for preserving distance.

D. Extracting Frequent Fragments After Topic Assignment

The output of Gibbs sampling is the assignments of different latent topic labels to all the nodes in all the workflows of a given corpus. To obtain the reusable fragments, the algorithm removes all edges which link up nodes with different topic labels in the corpus. The final output consists on a set of connected components where all nodes in each connected component have the same topic. They form the final set of reusable workflow fragments.

In some cases, there are outlier nodes which have topics different from the topic shared by all their neighbors, which thus break a fragment into multiple pieces. In order to handle this situation, before applying Algorithm 1, we detect such outliers and replace the topics of the outlier nodes to have the same topic as their neighbors'. After extracting the candidate fragments, we remove all the repeated or one-step fragments as we are interested in fragments with at least two steps.

Algorithm 1 Extracting fragments from a workflow corpus with topic labels

function PartitionCorpus(G,Topic)
Input: G, *the workflow corpus*
Output: $Topic$, *topic labels of nodes*
  for $(inNode, outNode) \in$ *edges in G* **do**
    if $Topic(inNode) \neq Topic(outNode)$ **then**
      remove edge (inNode,outNode) in G
    end if
  end for
  return *Partition of G*

A prototype implementation of PSM-FLow is available online [16].

IV. Experiment Setup

We use three workflow corpora from LONI Pipeline [6], a widely-used workflow system for neuroimaging analysis. These corpora were used to evaluate gSpan and SUBDUE in [10]:

1) Workflow corpus 1 (WC1): A set of 441 workflows designed mostly by a single user. Some of the workflows

Corpus	Approach	Frequency	Frag num	Precision	Recall	F score
WC1	SUBDUE MDL	2 occur	264	0.506	0.704	0.589
	SUBDUE SIZE	2 occur	381	0.471	0.749	0.578
	gSpan	2.00%	637	0.677	0.645	0.661
		10.00%	110	**0.714**	0.385	0.5
		15.00%	33	0.689	0.213	0.325
	REAFUM	5.00%	208	0.606	0.462	0.525
		10.00%	44	0.648	0.25	0.36
	PSM-Flow	1 occur	708(37.7)	0.533(0.012)	**0.797(0.019)**	0.639(0.014)
		2 occur	481(25.8)	0.585(0.015)	0.773(0.016)	**0.666(0.011)**
WC2	SUBDUE MDL	2 occur	95	0.366	0.435	0.398
	SUBDUE SIZE	2 occur	88	0.383	0.469	0.422
	gSpan	2.00%	127	0.545	0.44	**0.487**
		5.00%	14	0.559	0.151	0.238
		10.00%	2	0.389	0.062	0.107
	REAFUM	5.00%	227	0.485	0.376	0.423
		10.00%	123	**0.607**	0.163	0.257
	PSM-Flow	1 occur	150(12.4)	0.382(0.019)	**0.536(0.03)**	0.446(0.02)
		2 occur	46(8)	0.569(0.026)	0.418(0.042)	0.481(0.033)
WC3	SUBDUE MDL	2 occur	186	0.237	0.487	0.319
	SUBDUE SIZE	2 occur	178	0.217	0.456	0.294
	gSpan	2.00%	108	0.391	0.444	0.416
		5.00%	29	0.255	0.052	0.086
		10.00%	9	0.226	0.016	0.029
	REAFUM	2.00%	76	0.332	0.314	0.323
		5.00%	45	0.269	0.047	0.08
		10.00%	4	**0.5**	0.016	0.03
	PSM-Flow	1 occur	318(9.66)	0.307(0.008)	**0.586(0.014)**	0.403(0.009)
		2 occur	150(10.9)	0.374(0.018)	0.514(0.022)	**0.433(0.019)**

TABLE II: Performance Comparison of PSM-Flow, SUBDUE, and gSpan on three workflow corpora.

are products of collaboration with other users. The domain of the workflows is in general medical imaging (brain image understanding, 3D skull imaging, genetic modeling of the face, etc.).

2) Workflow corpus 2 (WC2): A set of 94 workflows from one user, sometimes done in collaboration with others. Most of the workflows have been made public.
3) Workflow corpus 3 (WC3): A set of 269 workflows, submitted to the LONI pipeline for execution by 62 different users.

The data is available at [8].

A. Workflow Data

When designing workflows, users tend to consider several related computational steps as a unit. Here we refer to this unit as a *user grouping*. User groupings occur frequently in workflow corpora as users reuse them. In order to assess PSM-Flow, we compare the groupings created by the users of the LONI Pipeline for the workflows in the evaluation corpora to the fragments automatically extracted by PSM-Flow, following a similar approach to [10].

We compare PSM-FLow to three popular frequent graph mining techniques, namely, gSpan, SUBDUE and REAFUM. gSpan uses an exact match and depth-first search strategy to discover all possible frequent fragments, while SUBDUE and REAFUM use inexact match approximation to discover a set of frequent fragments. For SUBDUE, two heuristics, Minimum Description Length (MDL) and size are used for hierarchically reducing the search space when mining for fragments [4].

For a fair comparison, we filter the candidate fragment set by removing single-step fragments and non-closed fragments (i.e., all those fragments that are included in another fragment with the same number of occurrences).

B. Evaluation Metrics

Since our extracted fragments are in many cases highly similar to user groupings but not exactly the same, we define *soft version* of precision and recall metrics for the measuring the performance. Let F denote the extracted fragments set, U the user grouping set, and $|X|$ the cardinality of the set X.

$$\text{Soft precision} = \frac{\sum_{f\in F} \max_{u\in U} overlap(f,u)}{|F|}$$

$$\text{Soft recall} = \frac{\sum_{u\in U} \max_{f\in F} overlap(f,u)}{|U|}$$

where the overlap between a fragment and a grouping is defined as the number of common node labels between them. Thus, the overlap value will be one if the fragment and the grouping are the same. Notice that the overlap value will be high as far as the node labels are the same, even though they may have different topologies. The validity of the proposed metric relies on the assumption that in a workflow corpus node labels in a fragment imply a certain topology (due to the compatibility between workflow steps) To take both precision and recall into account, we also calculate the F-score for comparison where $\textit{F-score} = 2 \cdot \frac{precision \cdot recall}{precision + recall}$.

C. Evaluation Results and Discussion

Table II shows our evaluation results. We run PSM-Flow for ten times for each data set and report the average and standard deviation (shown in parenthesis) of performance measures including precision, recall and the number of fragments for all corpora. In addition, the frequency of gSpan and REAFUM (i.e., the minimum number of times a common fragment has to appear to be detected) is normalized to the size of the corpus (2% means that any fragments appearing in more than 2% of the workflows of a corpus will be detected).

It should be noticed that looking merely at precision and recall could be misleading because different methods with different parameter settings give different numbers of fragments. For example, it is normal that the more number of fragments, the higher the recall, although not necessarily higher precision. Because of that, we cannot declare one method is better than the others based on either recall or precision, as they are highly affected by the number of detected fragments which can be adjusted by the minimum support. F-score is a commonly used as the performance metric to address the bias as it balances the precision and recall. In terms of F-score, PSM-Flow performs best in WC1 and WC3 when removing all candidate fragments with a single occurrence and comparable to gSpan in WC2.

Sometimes, precision could be more preferable over recall in our task due to the fact that what we are interested in is in getting useful fragments, i.e., those that are similar to what users have identified, rather than high recall, which is subjective to the way users design. Figure 5 compares performance in terms of precision and demonstrates that PSM-Flow tends to obtain higher precision for the more frequent fragments in all corpora.

In addition, we also plot the precision-recall graph for each method as shown in Figure 6. To compute it, we first sort the fragments by their support values so that we can get top-k fragments. Then we plot precision/recall with different values of k for all methods. PSM-Flow outperforms all methods for WC2 and WC3, and all except SUBDUE (SIZE) for WC1 when the number of fragments is large. The result implies that PSM-Flow can achieve better precision and recall trade-off.

For qualitative evaluation, two particular workflow fragments extracted from WC2 under the same topic are shown in Figure 7. The two fragments are used for "graphical model based multivariate analysis (GAMMA)" for neuroimaging. Compared to the fragment on the left, the fragment on the right replaces GAMMAalgorithm component with GAMMAEL (GAMMA with ensemble learning). This is a case where two fragments share similar functionality but using different versions of an algorithm. PSM-Flow can extract both fragments and put them into the same topic. This means that PSM-Flow can effectively capture variations of a fragment as a topic.

In terms of scalability, we have applied PSM-Flow to a synthetic workflow data generated using a procedure similar to [1]. We tested PSM-Flow with different numbers of workflows, the number of node label types to 100,the number of fragments contained in a workflow to 5, the size of generated fragments to 15 and the number of frequent fragment types to 20. The plot of run-time versus the number of workflows is presented in Figure 8. It shows that the run-time of PSM-Flow grows close to linearly with the size of workflow corpus, which is another favoring property of PSM-Flow.

V. Conclusions And Future Work

In this paper, we introduced PSM-Flow, a probabilistic subgraph mining algorithm which is a novel way to adopt the topic modeling approach for robust reusable fragment discovery in scientific workflows. The method tends to improve the robustness of discovered fragments by capturing their variations in a probabilistic framework. Promising results have been obtained using workflow data.

For future work, we may further leverage on the node attributes of workflows corresponding to different users to achieve customized fragment discovery and recommendation. It is also possible to further extend the evaluation of PSM-Flow in other graph mining tasks such as social network analysis.

Acknowledgement

This work was supported in part by HKBU CS Dept Overseas UG Summer Research Scheme, and in part by the US Defense Advanced Research Projects Agency with award FA8750-17-C-0106.

References

[1] Rakesh Agrawal and Ramakrishnan Srikant. "Fast Algorithms for Mining Association Rules in Large Databases". In: *Proceedings of the 20th International Conference on Very Large Data Bases*. VLDB '94. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994, pp. 487–499.

[2] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. "Latent Dirichlet Allocation". In: *Journal Machine Learning Research* 3 (Mar. 2003), pp. 993–1022.

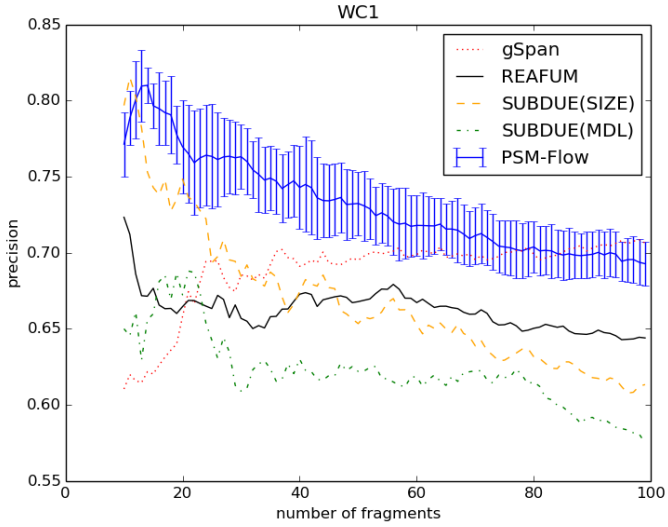
[3] C. Chen et al. "gApprox: Mining Frequent Approximate Patterns from a Massive Network". In: *Proceedings of the 7th IEEE International Conference on Data Mining*. Oct. 2007, pp. 445–450.

[4] Diane J. Cook and Lawrence B. Holder. "Substructure Discovery Using Minimum Description Length and Background Knowledge". In: *Journal of Artificial Intelligence Research* 1.1 (Feb. 1994), pp. 231–255.

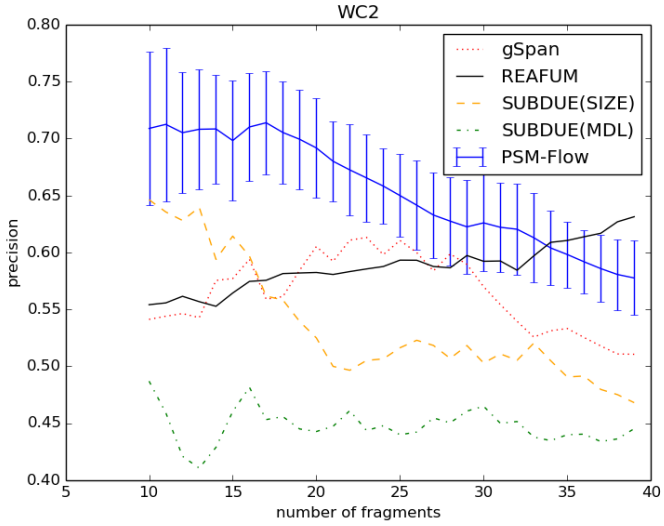
[5] Claudia Diamantini, Domenico Potena, and Emanuele Storti. "Mining Usage Patterns from a Repository of Scientific Workflows". In: *Proceedings of the 27th Annual ACM Symposium on Applied Computing*. SAC '12. Trento, Italy: ACM, 2012, pp. 152–157.

[6] Ivo D. Dinov et al. "Efficient, Distributed and Interactive Neuroimaging Data Analysis Using the LONI Pipeline". In: *Frontiers in Neuroinformatics*. Vol. 3. 22. 2009.

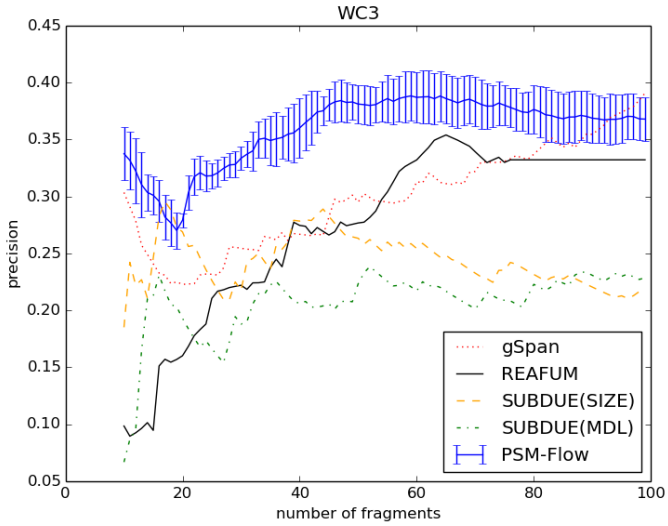
[7] Thomas S. Ferguson. "A Bayesian Analysis of Some Nonparametric Problems". In: *The Annals of Statistics* 1.2 (Mar. 1973), pp. 209–230.



(a) WC1

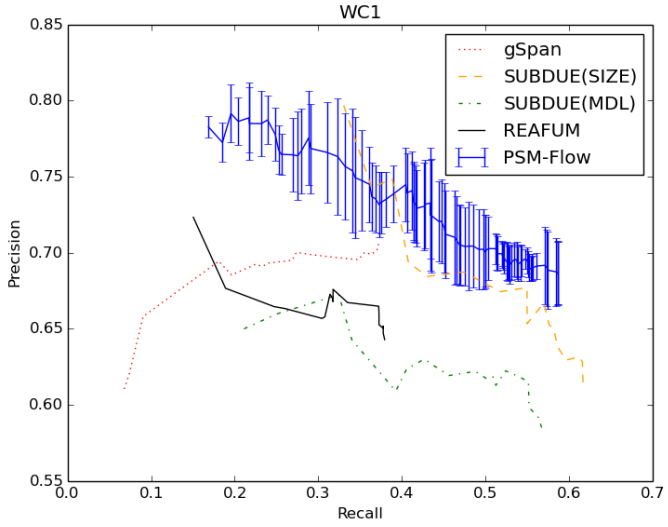


(b) WC2

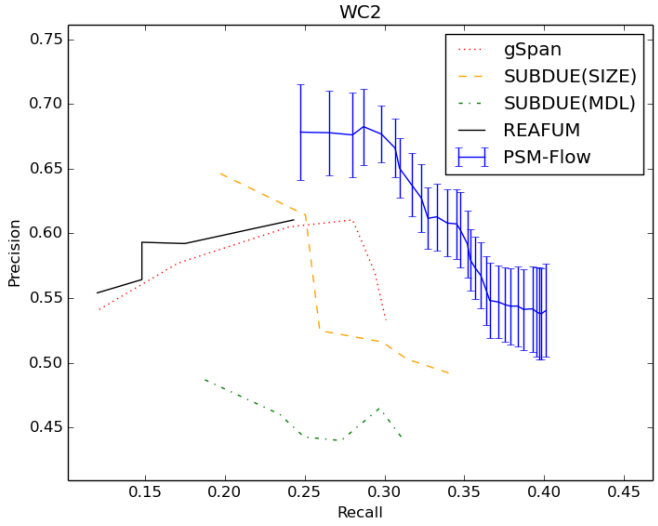


(c) WC3

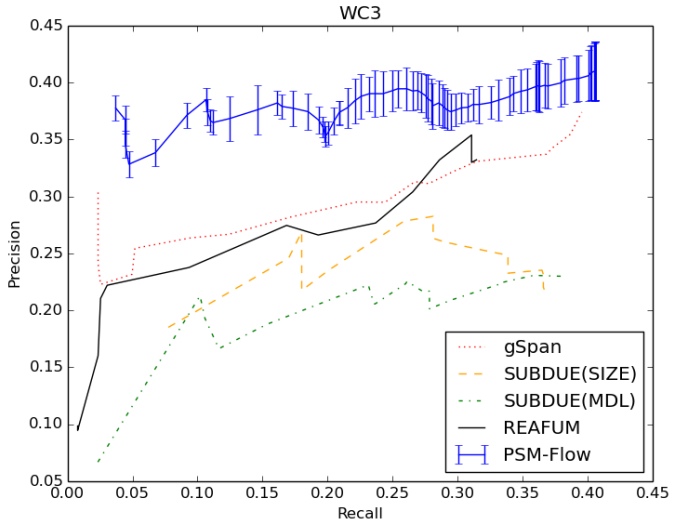
Fig. 5: Precision for top k fragments.



(a) WC1



(b) WC2



(c) WC3

Fig. 6: Precision-Recall curve.

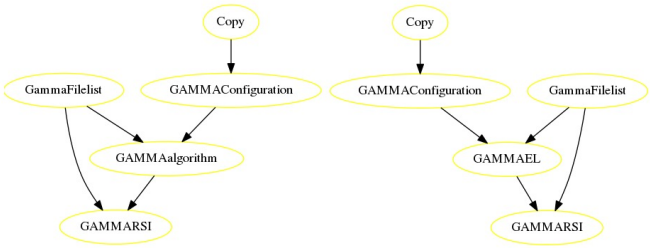


Fig. 7: Two fragments extracted by PSM-Flow from WC2.

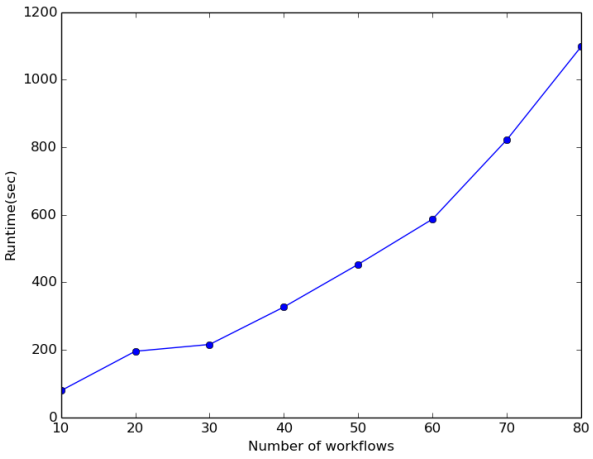


Fig. 8: Scalability on the number of workflows.

[8] Daniel Garijo. "The LONI Pipeline workflow inputs". In: (Nov. 2015). DOI: 10.6084/m9.figshare.1603175.v2. URL: https://figshare.com/articles/The_LONI_Pipeline_workflow_inputs/1603175.

[9] Daniel Garijo et al. "Common Motifs in Scientific Workflows: An Empirical Analysis". In: *Proceedings of IEEE 8th International Conference on e-Science*. Oct. 2012, pp. 1–8.

[10] Daniel Garijo et al. "FragFlow Automated Fragment Detection in Scientific Workflows". In: *Proceedings of 2014 IEEE 10th International Conference on e-Science*. Vol. 1. Oct. 2014, pp. 281–289.

[11] Daniel Garijo et al. "Workflow reuse in practice: a study of neuroimaging pipeline users". In: *Proceedings of 2014 IEEE 10th International Conference on e-Science*. Vol. 1. Oct. 2014, pp. 239–246.

[12] Yolanda Gil. "From Data to Knowledge to Discoveries: Artificial Intelligence and Scientific Workflows". In: *Sci. Program.* 17.3 (Aug. 2009), pp. 231–246.

[13] Yolanda Gil et al. "WINGS: Intelligent Workflow-Based Design of Computational Experiments". In: *IEEE Intelligent Systems*. 2011.

[14] T. L. Griffiths and M. Steyvers. "Finding scientific topics". In: *Proceedings of the National Academy of Sciences* 101.Suppl. 1 (Apr. 2004), pp. 5228–5235.

[15] Rui Jiang et al. "Network motif identification in stochastic networks". In: *Proceedings of the National Academy of Sciences* 103.25 (2006), pp. 9404–9409.

[16] KenCheong. *KenCheong/PSM-Flow v2.0*. June 2018. DOI: 10.5281/zenodo.1289781. URL: https://doi.org/10.5281/zenodo.1289781.

[17] M. Kuramochi and G. Karypis. "An efficient algorithm for discovering frequent subgraphs". In: *IEEE Transactions on Knowledge and Data Engineering* 16.9 (Sept. 2004), pp. 1038–1051.

[18] M. Kuramochi and G. Karypis. "GREW - A scalable frequent subgraph discovery algorithm". In: *Proceedings the 4th IEEE International Conference on Data Mining*. Nov. 2004, pp. 439–442.

[19] Ruirui Li and Wei Wang. "REAFUM: Representative Approximate Frequent Subgraph Mining". In: *Proceedings of the 2015 SIAM International Conference on Data Mining*. 2015.

[20] Kai Liu, William K. Cheung, and Jiming Liu. "Detecting Multiple Stochastic Network Motifs in Network Data". In: *Knowledge and Information Systems* 42.1 (Jan. 2015), pp. 49–74.

[21] Siegfried Nijssen and Joost N. Kok. "A Quickstart in Frequent Structure Mining Can Make a Difference". In: *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '04. Seattle, WA, USA: ACM, 2004, pp. 647–652. ISBN: 1-58113-888-1.

[22] Carl Edward Rasmussen. "The Infinite Gaussian Mixture Model". In: *Advances in Neural Information Processing Systems 12*. Ed. by S. A. Solla, T. K. Leen, and K. Müller. MIT Press, 2000, pp. 554–560.

[23] David De Roure, Carole A. Goble, and Robert Stevens. "The design and realisation of the myExperiment Virtual Research Environment for social sharing of workflows". In: *Future Generation Computer Systems* 25.5 (2009), pp. 561–567.

[24] Johannes Starlinger, Sarah Cohen-Boulakia, and Ulf Leser. "(Re)Use in Public Scientific Workflow Repositories". In: *Scientific and Statistical Database Management*. Ed. by Anastasia Ailamaki and Shawn Bowers. Vol. 7338. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 361–378.

[25] Sinead Williamson et al. "The IBP Compound Dirichlet Process and Its Application to Focused Topic Modeling". In: *Proceedings of the 27th International Conference on International Conference on Machine Learning*. ICML'10. Haifa, Israel: Omnipress, 2010, pp. 1151–1158.

[26] Xifeng Yan and Jiawei Han. "gSpan: graph-based substructure pattern mining". In: *Proceedings of 2002 IEEE International Conference on Data Mining*. 2002, pp. 721–724.