An ontology and frequency-based approach to recommend activities in scientific workflows

ABSTRACT

Nowadays there are several systems to help scientists in their daily activities. One type of these systems are the Scientific Workflow Management Systems which helps scientists to model, execute, store and share their experiments. In order to provide a more efficient and helpful experience for the user, these systems typically contains some kind of recommender system. This paper presents a novel hybrid approach to recommend activities in scientific workflows is based on activities frequency and domain ontology. Moreover, the recommendation problem was treated as a classification problem and as a regression problem. The validation considered the use of real workflows from the myExperiment repository and the results obtained by the proposed approach were compared with the ones obtained by the most used approaches. The results showed the proposed approach overcomes the traditional ones in all the evaluation metrics considered.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Recommender Systems; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

Keywords

Activities recommendation; Ontology based recommendation; Scientific Workflows; Artificial Intelligence; Recommender systems

1. INTRODUCTION

Nowadays, Scientific Workflow Management Systems (SWMS) are being increasingly adopted as a tool for helping scientists in the modeling, execution, storing, and sharing of their experiments. Scientific workflows are representations of structured processes, built manually, semi-automatically or automatically in order to solve scientific problems. The building blocks of these workflows are called activities, which can be: i) source code; ii) services; and iii) *finalized workflows* [28]. More then just helping scientists in the creation and execution of their experiments, these systems also stimulate the reuse of existing activities.

In the majority of the SWMS, activities are represented graphically as icons and the system has drag and drop functions. Thus, anyone can build computational experiments dragging icons and filling input parameters. Most of these systems provide sets of basic activities that can be used in different domains, for example, an activity which calculates the average value of a dataset is applicable in biology, physics, astronomy, and other areas. But there is a pre-condition for creating workflows: to know what are the available activities.

Currently, there are a large number of activities available in repositories such as *myExperiment* which stores more than 2,500 workflows [6] and *BioCatalogue* that provides more than 2,464 services [3]. This great amount of activities and the low reuse of most of the activities and workflows [28] motivated the development of techniques to recommend activities to scientists during the workflows composition.

Recommender systems aims to suggest items (products, books, movies, activities, etc.) that will be useful for the users. In the scientific experiment context, recommender system allow the scientists to take advantage of the activities reuse in scientific workflows without the need of knowing a great number of activities, avoiding the creation of activities similar to the ones already available.

This paper presents a hybrid approach to recommend scientific workflows activities based on frequency and the use of a domain ontology (*knowledge base*. Moreover, the recommendation problem was treated as both: a classification problem and a regression problem. Different classifiers were tests, such as Support Vector Machine (SVM), Naive Bayes (NB), K-Nearest-Neighbour (KNN), Classification and Regression Trees (CART), and Multi Layer Perceptron Neural Network (MLP). And the regression was obtained by different function generators, such as Support Vector Regression (SVR), CART, Neural Network, Multivariate Adaptive Regression Splines (MARS), and Binomial Regression (RB). These different methods were evaluated using real workflow data from myExperiments and the results were compared

SAC'16, April 4-8, 2016, Pisa, Italy

http://dx.doi.org/xx.xxxx/xxxxxxx.xxxxxxx

with the most used approaches in the related literature.

The rest of this paper is organized as follows. Section 2 presents the concepts used in this work. Section 3 summarizes related work. Section 4 presents the methodology used. Section 5 contains a brief description of the classifiers and regression algorithms used as the basis for the proposed approach. Section 6 presents and discusses the results. Finally, Section 7 contains the conclusions and future work.

2. BASIC CONCEPTS

2.1 Recommender systems

Recommender systems aims to recommend items that are interesting to users. Given a set C of all users, a set of all items that can be recommended S, the function u which assigns the utility of the item s to the user c, $u : C \times S \rightarrow R$ where R is a fully sorted set. For each user $c \in C$ the recommender wants to choose $s^{'} \in S$ which maximizes the utility function:

$$\forall c \in C, \quad s_c^{'} = \arg\max_{s \in S} u(c, s) \tag{1}$$

In recommender systems, the utility function u is not defined for the entire space $C \times S$, thus, the recommender systems must extrapolate the known space [1].

In order to solve this problem, different techniques were proposed to recommend items. Paiva et al. [9] classified them in six groups. The first, *Content-based*, recommends items similar to others previously selected by the user. Its limitations are: i) limited analysis of the item content that will be recommended (usually there are no semantic description of the item); ii) super-specialization: it occurs when the users receive recommendations too similar to their choices; and iii) new users must evaluate a minimum number of items before the system can recommend items for them.

The second, *Collaborative Filter* recommends items that have been selected by *similar* users. Its limitations are: i) new user problem (there are no useful information available to identify the users similar); ii) new items are only recommended when they were evaluated by users; iii) sparse data: few users often evaluate many items and most of the users evaluate few items, therefore, the matrix (users $\times$ items) is sparse. Thus, rare items (which were evaluated by few) are unlikely to be recommended.

The third, *Hybrid approach*, combines features of existing techniques trying to minimize their limitations.

The fourth, *Community-based*, is based on information from the user social network (community). The recommendation is carried out according to the preferences of the users's friends and colleagues rather than preference of unknowns. It is a type of specialization of collaborative filter inheriting its features.

The fifth, *Demographic*, uses attributes such as region, age, and language for the recommendation. It was created to try to minimize the sparsity problem and is a specialization of the collaborative filter, and assumes that users with the same demographic characteristics may be considered similar.

The sixth, *Knowledge-based*, recommends items according to the application domain. The similarity is calculated considering semantic characteristics of the items. Its main limitation is the need for semantic descriptions (using, for example, ontologies) about the domain, items, and users.

2.2 Scientific Workflow Management Systems

Scientific Workflow Management Systems are software infrastructures which allow the construction, execution, reuse, and provenance storage of scientific experiments represented as workflows [21]. Workflows allow the modeling and the computational execution of scientific problems by combining data and operations on data in configurable structures composed of activities [12].

There are different paradigms to model scientific workflows (called *Models of Computation* - [MoC]), which represents how data is exchanged between activities and the types of operations on data available and/or allowed. This paper discusses two MoCs: *dataflow* and *control flow*. The first, most widely used in scientific workflows, performs transformations on the data, provides data visualizations, and prepares simulations. The second, most used in business process workflows, emphasizes events, flowcharts and sequence of activities to be developed [20]. It is common to find workflow management systems that combine these two paradigms.

In scientific workflows, typically, there are an intense use of *dataflow* elements and few *control flow* elements [20]. The workflows elements can be be classified according to their structures, they are called *subworkflows* when composed of several chained activities and encapsulated [22], called *activity*, if they correspond to a single activity [11], and *Shim*, if they act as adapters/connectors between two activities that are syntactically incompatible [19].

During the construction and execution of workflows, some management systems allow the capture of provenance data [33]. According to Lim et al. [18] such provenance can be classified in two types: i) *provenance prospective*, which can be captured during the workflow construction and models the specification of a workflow; and ii) *retrospective provenance*, which models the execution of workflows, i.e., what tasks were performed, and what transformations on the data occurred. This information is captured in runtime.

The workflow construction (or composition) corresponds to the inclusion of activities, the connection between these activities, the link between the input data and activities, as well as, the filling of some activities' parameters.

To assist in this construction, some composition approaches were proposed. The automatic composition approaches define the problem to be modeled and the management system automatically creates an workflow which "solves" the input problem, connecting the activities automatically for the user. This approach is recommended for users who do not know specific details of the process and/or are not concerned about how the workflow will solve the input problem. The approach based on the recommendation of activities is used during the composition of workflows and the management system suggests to the user some activities that can be useful for the workflow under construction. This suggestion is usually based on similarity measures, for example, seeking

for activities in similar workflows, or seeking for activities used by users with the same profile of the current user. The recommendation technique is suitable for more expert users who want to participate in the workflow construction.

2.3 Recommendation of activities in workflows

There are two main tasks that must be considered by recommender systems in SWMS. The first is to maximize the utility function described in the equation (1), in other words, recommend items that meet the users' needs. The second address domain-specific problems, in the context of this paper, recommending activities for scientific workflows. There are constrains about the connections between activities' input and output (only compatible data types can be connected), semantic dependency between activities, and some control flow restrictions (some activities must be executed before others).

The dependence between activities' input and output requires the compatibility between the data types: the data type from the output of the previous activity must be compatible with the input of the activity to be recommended, for example, if activity a produces as output a number, only activities which receives a number as input parameters should be recommend to follow the activity a in the workflow.

The connection of two activities considering only the compatibility of the data types of outputs and inputs does not guarantee the workflow will run correctly or the user's problem will be solved. This happens due to the possible of semantic mismatch between activities. For example, if activity a produces as output a number corresponding to a temperature and activity b receives as input a number corresponding to a gene id, the data type (number) is the same, but, semantically, these two activities are not compatible.

Besides the syntactic (data type) and semantic compatibilities, it is necessary to connect the activities in the correct order. Unless in movies or books recommender systems, in the activities recommender systems the order of the items is important. For example, given two activities over a database: the first update a register, and the second queries the database (querying the updated register). The order in which these activities will be executed will alter the results produced.

These characteristics led to the development of specific approaches to recommend activities in scientific workflows. These approaches will be described in the next section 3.

3. RELATED WORK

The related literature presents some approaches to recommend activities in scientific workflows. One of the most common is the use of the frequency of the occurrence of a pair of activities [27, 29, 4, 10, 37, 36]. Other approaches area based on data provenance [24, 25, 8, 16, 13, 32]. Compatibility between workflow activities' inputs and outputs is the base of some related work [34, 2, 35]. Other works use semantic annotation for helping the recommendation [7, 38].

Some recent approaches uses the reliability between users and services to improve the recommendation process [31]. Association rules, such as *Apriori* algorithm are also used to identify the frequent itemsets [26, 30] and, thus, the items (activities or services) that will be recommended.

The present work aims to use artificial intelligence techniques to take advantage of the main characteristics of the approaches proposed in the literature.

4. MATERIAL AND METHODS

The data used in the test ad validation of the approach was obtained from the myExperiment [23] repository and contains 72 bioinformatics workflows. The data was organized as a matrix M, where lines represent workflows and columns represent all the activities available. The element $M_{i,j} = 0$ means that the workflow represented by the line i does not contains the activity j, and the element $M_{i,j} = 1$ indicates that the workflow from line i contains the activity j.

To test the recommendation techniques, the dataset was divided into two sets: 90% of the data for training set and 10% for the testing set.

In order to treat the recommendation problem as a classification or regression problem the dataset was transformed. Since a binary classifier expects to receive a set of positive and negative examples in order to learn how to classify, the following steps were performed. All the dataset will receive a new column (called *class*) to indicate if the respective line corresponds to a correct (real) workflow or not. Initially, all the lines will receive the value *TRUE*, i.e., they correspond to correct workflows. Each of the lines (workflows) will become 60 lines: one line will stay intact and, for the others, one specific activity will be removed (choose randomly for each workflow), and one activity will be added (one different activity for each one of the 59 lines remaining), the added activity will be one of the 59 most frequent activities in the original dataset that were not present in the current workflow. These new 59 lines will have the value *FALSE* in the column that represents the class. Thus, for a given workflow, only one representation will be considered a positive instance an the other 59 will be considered negative. The value 59 was chosen based on the distribution of the frequencies of the activities in the dataset.

In order to deal with the unbalanced training dataset, a oversampling technique was performed. Therefore, for each original line from the dataset, 59 new lines were created as negative instance (as presented) and 59 copies of the positive instance were created.

Given a workflow without one of its activities, the classifiers produces binary results indicating which should be the corresponding workflow(s) (from which it is possible to identify the recommended activity). On the other hand, the results of the regression are numbers and its values can be used to rank the recommend activities or a threshold can be used to verify if an activity should or not be recommend.

Since recommender systems expects to receive a sorted list of recommended activities, whenever two activities receives the same rank, the following tiebreaker is used: the activities most frequently used in the original dataset come first. If two activities still have the same rank, then they are sorted alphabetically in the result.

Two metrics were used to evaluate the approach and compare it with related one: *S@k*, which measures the percentage of correct items among the k fist positions of the recommender system resulting list of activities, and *Mean Reciprocal Rank* (MRR), which measures the average position of the correct item in the resulting list.

5. CLASSIFICATION AND REGRESSION

This section summarizes the main concept used by each of the classifiers and function regression algorithms used.

The KNN classifier starts with a training set of labeled elements and a unlabeled set of elements that will be classified. For each instance of the unlabeled elements the KNN finds the k nearest neighbors of the training set, the instance is classified in the class which has the most close neighbors [17].

The Classification And Regression Tree (CART) use a binary tree structure to learn and make decisions. This tree is organized as a root node, a number of decision nodes, and leaf nodes. Each element to be classified starts the decision-making flow through the root of the tree, each decision node performs a logical test based on some attribute. When the element reaches a leaf it will receive the respective classification. The construction of the tree depends on the algorithm that is being used and the applied division criterion. The CART regressor uses the residual mean deviance $\sum_{i=1}^{n}(y_i - f(x_i))^2$ instead of the Gini index (or information gain) and each leaf node will be filled with the average value of the training examples represented by the leaf. Other aspects such as construction, pruning and training are identical to traditional tree classifiers [5].

Classifiers based on Bayesian methods use training data to calculate the observed probability in each class based on the values of their features. These classifiers present better results when applied to problems where the information of various attributes may be considered simultaneously to estimate the output probabilities. The Naive Bayes algorithm is an example of these classifiers, it is an application of Bayes' theorem adapted to classification, taking some naive preconditions on data such as: i) independence among the characteristics; and ii) that all the features are equally important. In the real world, these preconditions are flawed, even thus, the algorithm presents satisfactory performance [17].

Neural networks (NNs) are inspired by the human brain, its neurons, and their connections. They goal is to model a relationship of weighted inputs and outputs that are defined by multiple processing nodes, which are responsible for calculating the sum of weighted inputs and transfers them to the activation function that determines whether a signal will be sent (or not) to the following neuron or the output of the network [15]. When NNs are used as classifiers, the response of the output layer is the instance classification, when they are used as regressors. The network weights can be used to produced the resulting value.

Support Vector Machines (SVM) are techniques that can be used to classify data using a hyperplane. The goal is to choose the optimum separation hyperplane in order to separate individuals from different classes. It corresponds to an optimization problem that can be formulated in the following way [15]:

$$\max_{\alpha} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j d_i d_j k(x_i, x_j) \tag{2}$$

with the following constrains

$$\sum_{i=1}^{N} \alpha_i d_i = 0 \tag{3}$$

$$0 \leq \alpha_i \leq C \tag{4}$$

$$C > 0 \tag{5}$$

where α are the Lagrange multipliers, d are the expected outputs, x is the input dataset, K is a kernel function, and C is given positive constant.

The SVR is an adaption of the SVM for regression, in this work we used the ϵ-*SVR*. It has an opposite goal when compared with the classifier: while the latter tries to maximize the separation (of instances belonging to different classes), the first aims to approximate the elements, with a given tolerance for mistakes. It can be formulated as an optimization problem in the following way [15]:

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_i^* - \alpha_j \alpha_j^* K(x_i, x_j) - \sum_{i=1}^{N} \alpha_i \alpha_i^* d_i \tag{6}$$

with the following constrains:

$$-C \leq \alpha_i - \alpha^* \leq C \tag{7}$$

$$\sum_{i=1}^{N} \alpha_i - \alpha_i^* = 0 \tag{8}$$

where α, α^* are the Lagrange multipliers, d are the expected outputs, x is the input dataset, K is a kernel function, and C is given positive constant.

The *Multivariate Adaptive Regression Splines*(MARS) is a non-parametric regression technique that can be seen as a generalization of the linear regression [14]. It uses linear segments of functions, and have the following structure:

$$(x-t)_+ = \begin{cases} x - t, \text{se } x > t \\ 0, \text{cc} \end{cases} \tag{9}$$

$$(t-x)_+ = \begin{cases} t - x, \text{se } x < t \\ 0, cc \end{cases} \tag{10}$$

The idea is to construct pairs of mirrored functions for each independent variable X_j with a node corresponding to each value $x_{i,j}$ of the respective variable. MARS regression model has the following function:

$$f(X) = \beta_0 + \sum_{m=1}^{M} \beta_m h_m(X) \tag{11}$$

where $h_m(X)$ is a function or the product of two or more functions; the β_m coefficients are estimated by the minimization of the residual squared mean deviance.

The Binomial Regression can be modeled as a *generalized linear model*, and is composed of three components: i) the

distribution of the dependent variable (in this case, binomial); ii) the linear predictor $\alpha + \beta X = \frac{p}{1-p}$; and iii) the *link* function, which relates the mean of the distribution with the linear predictor, in our case it is: $g(\mu) = \log(\frac{p}{1-p})$ [14].

The dependent variable Y, in the binomial regression, follows a normal distribution and the link function and the predictor are showed in equation (14).

$$g(\mu) = \log_e \left(\frac{\pi}{1-\pi} \right) \quad (12)$$

$$g(\mu) = \beta_0 + \sum_{j=1}^{p} \beta_j X_j \quad (13)$$

$$\log_e \left(\frac{\pi}{1-\pi} \right) = \beta_0 + \sum_{j=1}^{p} \beta_j X_j \quad (14)$$

where $\pi = \mu$ is the mean of Y, p corresponds to the data dimension, β are the regression coefficients and they are estimated using the maximum likelihood, and X is the input dataset.

6. RESULTS

Table 1 presents the results obtained by the different approaches proposed in this paper and some from the related literature.

The classification techniques CART, KNN, NNET; the random recommendation, and the use of the Apriori algorithm (proposed in [26, 30]) presented a very low performance for the dataset used. The three classifiers failed to converge, consequently did not make a good ranking. Regarding the other two techniques the poor performance of the random recommendation is due to the existence of many activities in the dataset (280), therefore the probability of randomically chooses the correct one is small, the Apriori does not considered the following key factors: order of activities, inputs and outputs of these activities, neither the semantics getting a poor performance (in this relative small and sparse dataset).

The NNET classifiers and the regression using Binomial, CART, MARS, and NNET presented better results for the metrics S@5 and S@10, suggesting the correct activities in among the first recommended ones. They also present better MRR results in comparison with the prior presented techniques. But, the results that worth to mention are the ones obtained by SVM classifier and regressor SVM, with the metric S@5 three times higher than the previous one $S@5 = 0.428$, and $S@10 \geq 0.714$. The metric MRR also showed higher results by this technique: $MRR_{Class} = 0.2958$ and $MRR_{Reg} = 0.3149$.

Among the classical techniques from the related literature, considering data: without provenance information; without information about the authors and reliability; and without prior semantic annotations [27, 29, 4, 10, 37, 36]. The best results were achieved combining the input and output constrains (I/O) with the frequency of the activities (considering their orders).

As presented, we also create an ontology and annotated the activities. The use of these information combined with the input and output constrains and the frequency of consecutive activities (*Freq. I/O Onto.*) achieved the best results for S@5 (0.571) ad MRR (0.3174). This approach did not achieved the best results only for S@10 (which best result was achieved by the SVM regressor).

Table 1: Recommender techniques results

Technique	S@5	S@10	S@100	S@280	**MRR**
Classifiers					
CART	0.000	0.000	0.000	1	0.0101
KNN	0.000	0.000	0.143	1	0.0102
NAIVE	0.000	0.000	0.000	1	0.0101
NNET	0.143	0.143	0.143	1	0.1524
SVM	0.428	0.714	1.000	1	0.2958
Regressors					
Binomial	0.000	0.285	0.571	1	0.277
CART	0.000	0.285	0.428	1	0.0391
MARS	0.000	0.285	0.428	1	0.0254
NNET	0.143	0.143	0.143	1	0.1524
SVR	0.428	0.857	1.000	1	0.3149
Other					
Random	0.000	0.000	0.000	1	0.0097
Apriori	0.000	0.000	0.143	1	0.0102
I/O	0.000	0.428	1.000	1	0.0562
Freq. I/O	0.428	0.714	1.000	1	0.2936
Freq. I/O Onto.	0.571	0.714	1.000	1	0.3174

Analyzing the results in details it was possible to identify the two main situations in each the use of ontology was very helpful in the improvement of the results. The first occurs when the activity to be recommended is the first of the workflow, therefore there is no previous activity to verify the frequency of the pair (previous and current) of activities. The second occurs when two activities have the same rank (considering input and outputs or frequency) but the ontology can identify which are the most promising activity according to each ontological annotation.

7. CONCLUSIONS

This paper presented a hybrid technique to recommend activities in scientific workflows based on frequency, input and output, and ontologies. Moreover it treated the recommendation problem to a classification an regression problem and tested these different approaches for recommendation of activities in scientific workflows using real data obtained in the myExperiment repository.

The best algorithms to accomplish this task were SVM and the combination the frequency and ontology-based one. Their performance was similar. The disadvantage of SVM is the very high training time to adjust parameters while the disadvantage of approach based on frequency and ontology is the need to know the application domain to build an ontology and annotate the workflows.

As future work the authors intend to use other variations of SVM, such as v-SVR, ϵ-SVR, and v-SVM. Multiclass classification and a mixed of classifier (ensemble method) will also be tested.

8. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Toward the next

generation of recommender systems: a survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749, 2005.

[2] N. Y. Ayadi and Z. Lacroix. Resolving Scientific Service Interoperability With Schema Mapping. In *2007 IEEE 7th International Symposium on BioInformatics and BioEngineering*, pages 448–455. IEEE, Oct. 2007.

[3] J. Bhagat, F. Tanoh, E. Nzuobontane, T. Laurent, J. Orlowski, M. Roos, K. Wolstencroft, S. Aleksejevs, R. Stevens, S. Pettifer, R. Lopez, and C. Goble. Biocatalogue: a universal catalogue of web services for the life sciences, June 2014.

[4] B. Cao, J. Yin, S. Deng, D. Wang, and Z. Wu. Graph-based workflow recommendation: on improving business process modeling. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, CIKM '12, pages 1527–1531. ACM, 2012.

[5] S. B. Connor. *Wiley Encyclopedia of Statistics in Quality and Reliability*, chapter Perfect Sampling. Wiley, 2007.

[6] C. G. David De Roure. myexperiment, junho 2014.

[7] D. de Oliveira, L. Cunha, L. Tomaz, V. Pereira, and M. Mattoso. Using Ontologies to Support Deep Water Oil Exploration Scientific Workflows. In *2009 Congress on Services - I*, pages 364–367. IEEE, July 2009.

[8] F. de Oliveira, L. Murta, C. Werner, and M. Mattoso. Using provenance to improve workflow design. In J. Freire, D. Koop, and L. Moreau, editors, *Provenance and Annotation of Data and Processes*, volume 5272 of *Lecture Notes in Computer Science*, pages 136–143. Springer Berlin Heidelberg, 2008.

[9] F. A. P. de Paiva, J. A. F. Costa, and C. R. M. Silva. A Hierarchical Architecture for Ontology-Based Recommender Systems. In *2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence*, pages 362–367. IEEE, Sept. 2013.

[10] C. Diamantini, D. Potena, and E. Storti. Mining Usage Patterns from a Repository of Scientific Workflows. In *Proceedings of the 27th Annual {ACM} Symposium on Applied Computing*, SAC '12, pages 152–157. ACM, 2012.

[11] D. Garijo, P. Alper, K. Belhajjame, O. Corcho, Y. Gil, and C. Goble. Common motifs in scientific workflows: An empirical analysis. In *2012 IEEE 8th International Conference on E-Science*, pages 1–8. IEEE, Oct. 2012.

[12] D. Garijo, P. Alper, K. Belhajjame, O. Corcho, Y. Gil, and C. Goble. Common motifs in scientific workflows: An empirical analysis. *Future Generation Computer Systems*, 36:338–351, July 2014.

[13] D. Garijo, O. Corcho, and Y. Gil. Detecting Common Scientific Workflow Fragments Using Templates and Execution Provenance. In *Proceedings of the Seventh International Conference on Knowledge Capture*, K-CAP '13, pages 33–40, New York, NY, USA, 2013. ACM.

[14] T. J. Hastie, R. J. Tibshirani, and J. H. Friedman. *The elements of statistical learning : data mining, inference, and prediction*. Springer series in statistics. Springer, New York, 2009.

[15] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 3nd edition, 2007.

[16] D. Koop, C. E. Scheidegger, S. P. Callahan, J. Freire, and C. T. Silva. VisComplete: automating suggestions for visualization pipelines. *IEEE transactions on visualization and computer graphics*, 14(6):1691–8, Jan. 2008.

[17] B. Lantz. *Machine Learning with R*. Packt Publishing, Birmingham, 1nd edition, 2013.

[18] C. Lim, S. Lu, A. Chebotko, and F. Fotouhi. Prospective and Retrospective Provenance Collection in Scientific Workflow Environments. In *2010 IEEE International Conference on Services Computing*, SCC '10, pages 449–456. IEEE, July 2010.

[19] C. Lin, S. Lu, X. Fei, D. Pai, and J. Hua. A Task Abstraction and Mapping Approach to the Shimming Problem in Scientific Workflows. In *2009 IEEE International Conference on Services Computing*, SCC '09, pages 284–291. IEEE Computer Society, 2009.

[20] B. Ludascher, I. Altintas, C. Berkley, D. Higgins, E. Jaeger, M. Jones, E. A. Lee, J. Tao, and Y. Zhao. Scientific workflow management and the kepler system. *Concurrency and Computation: Practice and Experience*, 18(10):1039–1065, 2006.

[21] T. McPhillips, S. Bowers, D. Zinn, and B. Ludäscher. Scientific workflow design for mere mortals. *Future Generation Computer Systems*, 25(5):541–551, May 2009.

[22] C. B. Medeiros, J. Perez-Alcazar, L. Digiampietri, G. Z. Pastorello Jr., A. Santanche, R. S. Torres, E. Madeira, and E. Bacarin. {WOODSS} and the Web: Annotating and Reusing Scientific Workflows. *{SIGMOD} Rec.*, 34(3):18–23, 2005.

[23] C. G. Roure. myexperiment, 2015.

[24] Q. Shao, M. Kinsy, and Y. Chen. Storing and Discovering Critical Workflows from Log in Scientific Exploration. In *2007 IEEE Congress on Services (Services 2007)*, pages 209–212. IEEE, July 2007.

[25] Q. Shao, P. Sun, and Y. Chen. Efficiently discovering critical workflows in scientific explorations. *Future Generation Computer Systems*, 25(5):577–585, May 2009.

[26] W. Tan, J. Zhang, R. Madduri, I. Foster, D. De Roure, and C. Goble. Providing Map and GPS Assistance to Service Composition in Bioinformatics. In *2011 IEEE International Conference on Services Computing*, pages 632–639. IEEE, July 2011.

[27] A. Telea and J. J. van Wijk. vission: An object oriented dataflow system for simulation and visualization. In *PROCEEDINGS OF IEEE VISSYM*, pages 95–104, 1999.

[28] F. Wang, H. Deng, L. Guo, and K. Ji. A Survey on Scientific-Workflow Techniques for E-science in Astronomy. In *2010 International Forum on Information Technology and Applications*, volume 1, pages 417–420. IEEE, July 2010.

[29] J. Wang, Y. Han, S. Yan, W. Chen, and G. Ji. Vinca4science: A personal workflow system for e-science. In *Internet Computing in Science and Engineering, 2008. ICICSE '08. International Conference on*, pages 444–451, 2008.

[30] Y. Wang, J. Cao, and M. Li. Change Sequence Mining in Context-Aware Scientific Workflow. In *2009 IEEE International Symposium on Parallel and Distributed Processing with Applications*, pages 635–640. IEEE, 2009.

[31] J. Yao, W. Tan, S. Nepal, S. Chen, J. Zhang, D. De Roure, and C. Goble. Reputationnet: A reputation engine to enhance servicemap by recommending trusted services. In *Services Computing (SCC), 2012 IEEE Ninth International Conference on*, pages 454–461, 2012.

[32] P. Yeo and S. S. R. Abidi. Dataflow Oriented Similarity Matching for Scientific Workflows. In *2013 IEEE International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum*, pages 2091–2100. IEEE, May 2013.

[33] R. Zeng, X. He, and W. van der Aalst. A Method to Mine Workflows from Provenance for Assisting Scientific Workflow Composition. In *2011 IEEE World Congress on Services*, pages 169–175. IEEE, July 2011.

[34] J. Zhang. Ontology-driven composition and validation of scientific grid workflows in kepler: a case study of hyperspectral image processing. In *Proceedings of the Fifth International Conference on Grid and Cooperative Computing Workshops*, GCCW '06, pages 282–289. IEEE Computer Society, 2006.

[35] J. Zhang. Ontology-Driven Composition and Validation of Scientific Grid Workflows in Kepler: a Case Study of Hyperspectral Image Processing. In *2006 Fifth International Conference on Grid and Cooperative Computing Workshops*, pages 282–289. IEEE, 2006.

[36] J. Zhang, C. Lee, S. Xiao, P. Votava, T. J. Lee, R. Nemani, and I. Foster. A Community-Driven Workflow Recommendations and Reuse Infrastructure. In *2014 IEEE 8th International Symposium on Service Oriented System Engineering*, pages 162–172. IEEE, Apr. 2014.

[37] J. Zhang, W. Tan, J. Alexander, I. Foster, and R. Madduri. Recommend-As-You-Go: A Novel Approach Supporting Services-Oriented Scientific Workflow Reuse. In *2011 IEEE International Conference on Services Computing*, pages 48–55. IEEE, July 2011.

[38] L. Zhang, Y. Y. Y. Y. Wang, P. Xuan, A. Duvall, J. Lowe, A. Subramanian, P. K. Srimani, F. Luo, and Y. Duan. Sesame: A new bioinformatics semantic workflow design system. In *2013 IEEE International Conference on Bioinformatics and Biomedicine*, pages 504–508. IEEE, Dec. 2013.