Teaching Big Data Analytics Skills with Intelligent Workflow Systems

Yolanda Gil

Information Sciences Institute and Department of Computer Science
University of Southern California
4676 Admiralty Way
Marina del Rey CA 90292
gil@isi.edu

Abstract

We have designed an open and modular course for data science and big data analytics using a workflow paradigm that allows students to easily experience big data through a sophisticated yet easy to use instrument that is an intelligent workflow system. A key aspect of this work is the use of semantic workflows to capture and reuse end-to-end analytic methods that experts would use to analyze big data, and the use of an intelligent workflow system to elaborate the workflow and manage its execution and resulting datasets. Through the exposure of big data analytics in a workflow framework, students will be able to get first-hand experiences with a breadth of big data topics, including multi-step data analytic and statistical methods, software reuse and composition, parallel distributed programming, high-end computing. In addition, students learn about a range of topics in AI, including semantic representations and ontologies, machine learning, natural language processing, and image analysis.

Introduction

Big data analytics has emerged as a widely desirable skill in many areas. Although courses are now available on a variety of aspects of big data, there is a lack of a broad and accessible course doe non-CS majors that enables them to learn about big data analytics in practice. Students with limited or no programming skills that are interested in data science include most students in science, engineering and the humanities. As a result, acquiring practical data analytics skills is out of reach for many students and professionals, posing severe limitations to our ability as a society to take advantage of our vast digital data resources.

We are developing educational materials for data science and big data analytics to provide broad and practical training in data analytics in the context of real-world and science-grade datasets and data analytics methods. We capture common analytic methods as computational workflows that are used by students for practice with real-world datasets within pre-defined lesson units. The workflows include semantic constraints that the system uses to assist the users to set up parameters and validate the workflows. Several key features of the educational materials that we are developing include:

1. Expose students to well-understood end-to-end data analysis processes that have proven successful in several challenging domains and represent the state-of-the-art
2. Allow students to easily experiment with different combinations of data analysis processes, represented as workflows of computations that they can easily reconfigure and that the underlying system can easily manage and execute
3. Provide students with a lesson units of structured lessons to analyze real-world and scientific data, posing significant challenges to the students over and above what is learned in textbooks
4. Guide users to achieve target performance levels in each lesson as they experiment with different algorithms by easily changing the workflow steps with a graphical editor
5. Teach students basic concepts concerning semantics, formats, and metadata for real-world datasets

This work supplements existing academic training materials in big data and data science, by providing an on-line workflow environment for practice and exploration that is accessible to non-programmers.

The course was pre-tested in the Summer of 2015 with four students, three are non-CS undergraduates and one is a high-school student. The students were able to follow the materials, and used basic programming skills from intro courses they had taken (one student learned R on her own) and developed new workflows for basic statistical analysis of data, for image processing (using the OpenCV open source package), and for basic social network analysis.

The course will be taught at USC in Spring 2016. It has been approved as a masters level course in the new Communications Informatics and Spatial Informatics programs. The students will be journalism and geography majors

Copyright © 2016, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

	Section	Lesson topics
1	Data	What is data and what is not data; time series data; network data; geospatial data; text data; labeled and annotated data; big data
2	Data analysis software	Software for data analysis; inputs and outputs of programs; program parameters; programming languages; programs as black boxes
3	Multi-step data analysis	Pre-processing and post-processing data; building workflows by composing programs; workflows for data analysis; workflow inputs and parameters; running a workflow
4	Data analysis tasks	What is a data analysis task; prediction; classification; clustering; pattern detection; anomaly detection
5	Data pre-processing	Data cleaning; quality control; data integration; feature selection
6	Data post-processing	Summarization; filtering; visualization
7	Analyzing different types of data	Analyzing time series data; analyzing networked data; analyzing geospatial data; analyzing text; analyzing images; analyzing video
8	Parallel computing	Cost of computation; parallel processing; multi-core computing; distributed computing; speedup with parallel computing; dependencies across computations; limits of parallel speedup; execution failures and recovery; reduction
9	Semantic metadata	What is metadata; basic metadata vs semantic metadata; metadata about data collection; metadata about data processing; metadata for search and retrieval; metadata standards; domain metadata and ontologies
10	Provenance	What is provenance; provenance concerning data; provenance concerning agents; provenance concerning processes; provenance models; provenance standards
11	Semantic workflows	What is a semantic workflow; validating data analysis methods; automatically generating metadata; tracking provenance; publishing workflows; finding workflows
12	Visualization	Time series visualizations; geospatial visualizations; multi-dimensional spaces
13	Data stewardship	Data sharing; data identifiers; licenses for data; data citation and attribution
14	Data formats and standards	Data formats; data standards; data services; ontologies; linked open data

Table 1. Major topics in the proposed course on data science for non-programmers.

respectively. The syllabus of the course is available from [Gil 2015]. The paper presents our approach to teach data science, gives an overview of the curriculum, and describes how semantic workflows are used to teach core concepts in the course complemented by practice with the WINGS intelligent workflow system.

Data Science for Non-Programmers

Our focus is on students that do not take programming classes. Our goal is that they learn basic concepts of data science, so they can understand how to pursue data-driven research projects in their area and be in a better position to collaborate with computer scientists in such projects.

Existing courses on data science typically require programming skills. As an example, Coursera's "Introduction to Data Science" [Coursera 2015] requires two college-level courses in programming. Even when targeted to non-programmers, data science curricula focus on teaching programming. For example, Columbia University's Journalism School offers a set of courses to introduce students to data practices [Columbia 2015] that starts out teaching basic programming skills.

Although it is always beneficial to learn programming, not every student is inclined to invest the time and effort to do so. A course that enables them to learn basic concepts of data science will be more approachable and still useful. In the spirit of computational thinking [Wing 2006], our focus has been to design a curriculum that teaches computing concepts above the level of particular programming languages and implementations.

Another observation about data science curricula is that they tend to focus on databases and machine learning, with little attention to parallel and distributed computing. Although database technologies and machine learning algorithms are important, it is also important to include other concepts such as scalability through parallelism and distributed computation, as the motivation to learn about data science is often the pursuit of big data analytics and that requires understanding how to scale up computation.

Table 1 presents the major sections and topics of our course for data science. All the topics can be introduced without requiring programming skills.

The course also has more emphasis on metadata and semantics than are usually included in data science courses. There is also more emphasis on end-to-end methods for

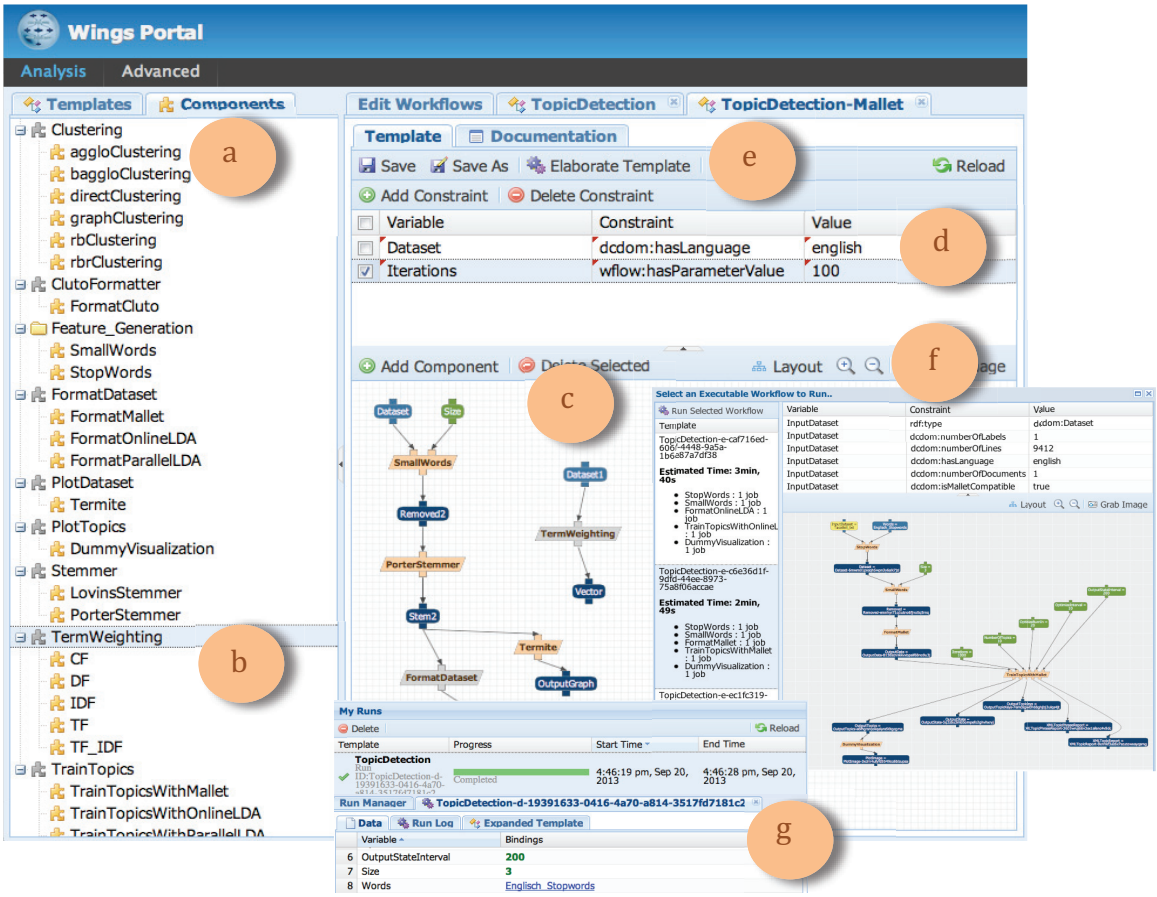

Figure 1: WINGS user interface for composing and validating workflows: (a) a library of components is available to the user including component classes organized in a hierarchy, (b) a component or a component class can be selected and dragged and dropped into the canvas, (c) once dropped the component can be connected to express dataflow, (d) the user can add constraints about the datasets or parameters of the workflow, (e) the user can ask the system to validate the workflow by reasoning about the semantic constraints, (f) the user can ask the system to expand the constraints of the workflow, even to expand it to add additional needed parameters and other information. The examples are for text analytics (from [Hauder et al 2011a; Hauder et al 2011b]).

data analysis, which include data pre-processing, data post-processing, and visualization.

Learning these concepts must be supplemented with practice. But how can students with no programming skills be able to see programs in action? A major component of the course is the use of an intelligent workflow system that enables students to practice complex data analysis concepts.

Semantic Workflows

For this work we use the WINGS semantic workflow system [WINGS 2015]. WINGS is an intelligent workflow system that can assist users, and therefore students, to create valid workflows [Gil et al 2011a] and automate the elaboration of high-level workflows [Gil et al 2011b]. Users find pre-defined workflows and workflow components that they can reuse and extend to create their own workflows. As users select and configure workflows to be executed, WINGS ensures that workflows are correctly composed by checking that the data is consistent with the semantic constraints defined for the workflow and its components. Users can track execution progress and view results.

Figure 1 shows a snapshot of the existing WINGS user interface for composing and validating workflows, in this case to create applications for text analytics [Hauder et al 2011a; Hauder et al 2011b]. WINGS can represent high-level workflow templates that have classes of components as steps (e.g., "TermWeighting"). WINGS workflow templates can also express compactly processing of multiple objects even before the collections are selected [Gil et al 2009]. WINGS can validate the workflows that are created by the user by reasoning about the semantic constraints that have been defined for the workflow, its components, and all the associated input and output data. WINGS can execute workflows in several scalable distributed execution engines [Gil et al 2013]. A high-level introduction to

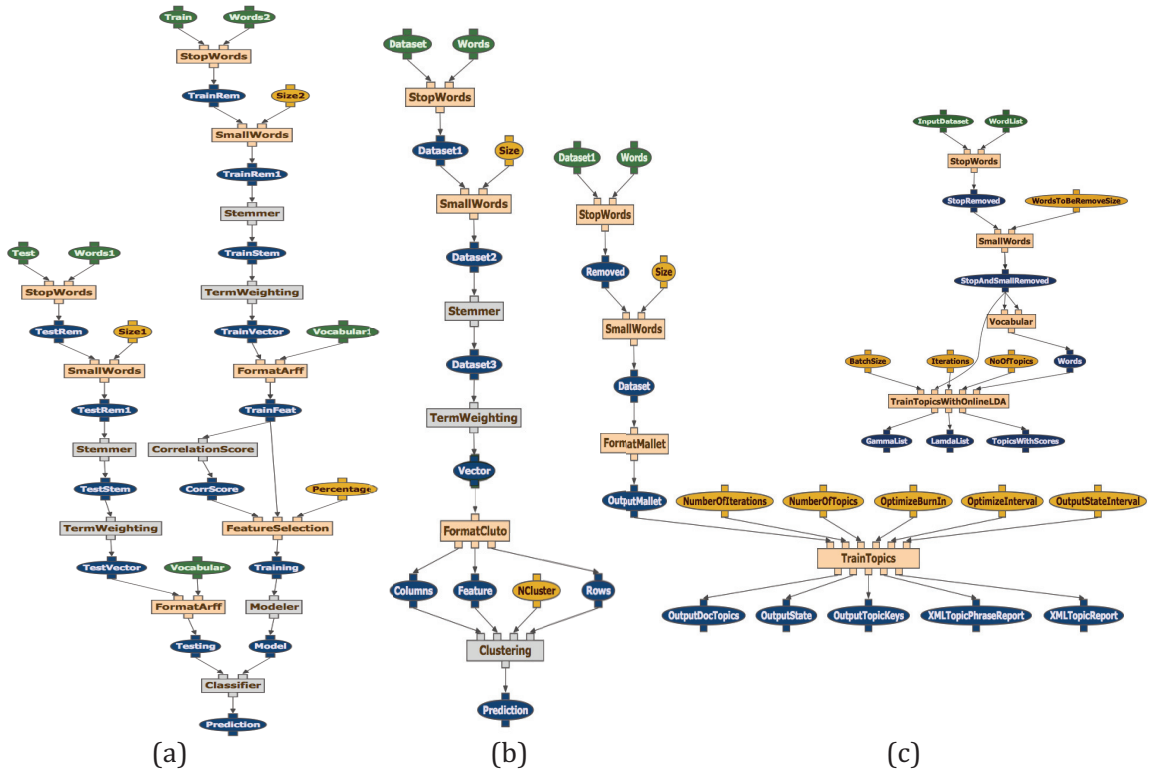

Figure 2: Workflows created by text analytics experts: (a) supervised classification of documents, (b) unsupervised clustering of documents, (c) detection of trending topics in document collections (from [Hauder et al 2011a]).

WINGS can be found in [Gil et al 2011a], a formal description of the workflow representation language and reasoning algorithms is given in [Gil et al 2011b].

Using Workflows to Teach Big Data Analytics

This section illustrates how the topics in the course are illustrated using the WINGS intelligent workflow framework.

A major benefit of using WINGS in the course is the ability for students to access expert-grade data analytic methods that are ready to be run with real-world data. Many courses on different levels of statistics, machine learning and data mining are offered in most universities and train students on the relative merits of different algorithms and statistical techniques. But a course on machine learning or data mining is not sufficient to prepare students to do data analytics in practice. First, designing an appropriate end-to-end process to prepare and analyze the data plays a much more influential role than using a novel classifier or statistical model. For example, in many cases, the prediction accuracy of text classifiers on one dataset can differ 5-10% depending on how the unstructured texts are converted to feature vectors. In contrast, once the data is preprocessed the difference between which classifier (such as support vector machines or Naïve Bayes) is applied on the same feature vector is only 0.5-5%. Second, state-of-the-art data analytics often involves multi-step methods with sophisticated statistical techniques such as cross-validation and combinations of algorithms such as ensemble methods. Such methods are challenging to set up and run and few users would have requisite experience and infrastructure to experiment with them. Finally, key data analytic skills can only be learned by experiencing the performance of different methods with real data, by understanding different data preparation strategies, and by exploring the relative merits of different algorithmic choices and their effect in the overall performance.

Figure 2 illustrates several workflows that capture expert-level methods for text analysis, taken from [Hauder et al 2011a]. These workflows were used by high-school students with no programming skills to analyze Web documents [Hauder et al 2011b]. We have had many users of WINGS who did not have programming skills and were able to run complex data analytics and understand the method captured in the workflow.

Students will work with multiple domains and use workflows that capture end-to-end expert-level analytic meth-

ods. Many topics in the course require exposing the student to sophisticated methods that work best with different kinds of data, notably 3, 4, and 7. Other topics are illustrated through the ability of workflows to show how data is prepared and pre-processed (topics 5 and 6).

Another major benefit of workflows is to illustrate how programming can be accomplished through software reuse and composition. More and more every day, many programming tasks involve reuse of libraries or software packages. Workflows emphasize this paradigm and allow students to learn a component-based approach to programming, where the components are already pre-defined and the user's task is to compose them through data flow and control flow connectors. These concepts are easy for non-programmers to understand, and allow them to develop pretty complex applications without formal programming training. In addition, workflows illustrate the concept of heterogeneous software applications, where different components may be implemented in different languages and used together in a workflow. Students can learn how to create software components from software packages available in the Web, so they can be empowered to build workflows for new domains.

The workflows in Figures 1-2 can be used to illustrate these concepts. These workflows include components from a variety of software packages and libraries, including generic packages such as MATLAB and R, machine learning packages such as Weka and Cluto, and text analysis packages such as Mallet. The components can be implemented in a variety of programming languages, including Java, C++, and Python. Workflows allow students with no programming background to understand and practice software reuse and composition.

Another key didactic use of workflows is to illustrate parallel distributed programming concepts. A major challenge in teaching data analytics is the difficulty for students to learn in practical ways about parallel programming. Significant differences among different algorithms and methods may be only apparent when using data at larger scales.

Figure 3 shows a workflow for text analytics that contrasts the processing of a collection (left) with the processing of a single document (right). WINGS includes workflow constructs that manage parallelization starting from abstract specifications of document and algorithm collections [Gil et al 2009]. We illustrate those constructs and their mapping into a Hadoop (MapReduce) infrastructure for workflow execution.

Metadata tracking is another set of topics where workflow systems can be helpful. Students should learn to manage metadata in analytic processes, as metadata is crucial to provide context to datasets. Metadata includes key information about: 1) the origins of raw data, such as the instruments used to collect it or the investigators involved;

Figure 3: A workflow that illustrates parallel processing: it processes a collection of training data (left) to build a classifier, then it uses it to classify test data (right). The documents that form the training dataset are preprocessed in parallel, indicated by the stacks of nodes.

2) the types and relationships across datasets, such as the temporal extent of each of several datasets; 3) the statistical properties of datasets, such as error rates and other quality indicators; and 4) the processes applied to derive new data from raw data sources. Metadata is often poorly managed and laborious to integrate into analysis steps, and so it is often lost through analytic processes. Key metadata is often archived locally by key investigators but not moved along with the data throughout the analytic process steps. Metadata for analytic results is tracked manually and seldom published. Provenance is a kind of metadata that workflow systems track automatically. A workflow captures the process that was used to create the workflow results, which represent the provenance of those results. Workflow systems can naturally generate provenance metadata [Kim et al 2007]. WINGS uses the W3C PROV standard for provenance on the Web (http://www.w3.org/2011/prov/). Many workflow systems can publish provenance records for workflow executions [Moreau and Ludaescher 2007]. WINGS is unique in that it can export these provenance records as well as their associated workflow templates, all as web objects that are openly accessible and use linked open data principles [Garijo and Gil 2011]. We have developed lesson units devot-

ed to metadata tracking through a workflow system, which does it automatically. This allows students to appreciate the importance of ontologies to describe datasets, and the role of Semantic Web technologies in providing standard representations and tools to manage metadata. Students can learn that process metadata is key for documenting results, so that they can be interpreted appropriately, searched based on what processes were used to generate them, and so that they can be understood and used by others. We have developed lesson units devoted to the importance of metadata and provenance in data analytics, particularly in a scientific context but also in commercial contexts where accountability is important.

Finally, workflow systems can help students learn about the value of visualizations. Visualization courses often focus on how to process big data at scale to generate visualizations that have illustrative purposes. However, the utility of visualizations to a user is often neglected. Visualizations need to be put in the context of understanding a dataset and the results of an analytic method once the data is processed. In this regard, workflows offer an integrated view of analytics and visualizations, since they include both data analytics and visualization steps.

Workflows with data visualizations steps allow students to understand how data visualizations make a difference in understanding a problem. In addition, visualizations are used to show intermediate results of the workflows, which helps convey the importance of pre-processing and other data preparation steps in the overall analytic method.

Learning AI Topics through Data Science

The students are exposed to several major topics in AI through the materials in the class. Specifically, they learn basic knowledge about these key topics:

- *Semantic representations and ontologies:* The datasets included in the workflow system have types, which are organized in a class hierarchy and where each type can have properties. This enables students to learn about how to create ontologies to describe domain information.
- *Constraint reasoning:* The ontological properties of the data are used to define metadata of datasets. The workflow system uses that metadata in logical rules to assert new metadata of workflow results. These are logical if-then rules that the students can define to constrain how the workflow processes the data.
- *Machine learning:* The students learn to run classifiers and clustering methods. They see examples of training data, their features and their labels. They can try different classifiers and compare their performance. They also learn to run clustering algorithms, and can set up the target number of clusters to different values to understand how the algorithm parameter values affect the results.
- *Image processing:* There are workflows where different image processing algorithms are applied to images with different characteristics (black and white or color, blurry or crisp, outdoors or indoors, faces or landscapes, etc.) Students can see how different transformations to an original image can affect the ability of a segmentation algorithm to detect certain object boundaries.
- *Natural language:* The workflows for text processing include steps for pre-processing text, formulating features out of words, stemming, and common statistics such as TF-IDF. They also learn to handle common Web data, such as HTML documents and tweets. Students can experiment with the workflows using documents of different characteristics (short or long, news or blogs, etc).
- *Network analysis:* There are workflows for network analysis that detect cliques, detect the most influential nodes, and find communication paths.

The course was pre-tested in the Summer of 2015 with four students, three are non-CS undergraduates and one is a high-school student. We illustrate some of the materials created by the students doing tasks equivalent to homework assignments.

Figure 4 shows an example of a logic constraint created by one of the students. It applies to a workflow component that generates a histogram. It is expressed as an if-then rule that sets the number of bins depending on the dataset size (input_values), which is a semantic property (metadata) of the input dataset. This constraint represents Sturges' rule, which sets the number of bins to the logarithm of the number of data points plus one.

Figure 5 shows an example of a workflow created by a student using OpenCV components. The student was interested to see if smoothing would affect the quality of the segmentation algorithm, and how different thresholds would affect the performance.

An important observation is that the context of data science gives the students a concrete basis to understand the AI concepts. Because they can experiment with the data, they can try different AI algorithms and techniques in machine learning, natural language, and image processing. Because they have to describe the data, they create logic representations of the data characteristics so the workflows can reason with them.

Related Work

Workflow systems such as Pegasus, Taverna, Kepler, and others [Deelman et al 2005; Hull et al 2006; Ludäscher et al 2006] provide sample workflows in their introductory materials and software downloads, but do not provide structured lessons. VisTrails has been used to teach a range of visualization topics [Silva et al 2011].

```
[Set_bins_parameter:
 (?c rdf:type pcdom:histogram1Class)
 (?c pc:hasInput ?idv)
 (?idv pc:hasArgumentID 'input_values')
 (?c pc:hasInput ?ipv)
 (?ipv pc:hasArgumentID 'number_of_bins')
 (?idv dcdom:hasDataPoints ?num)
 log(?num '2 ?out)
 addOne(?out ?bins)
 -> (?ipv pc:hasValue ?bins)]
```

Figure 4: A (simplified) semantic constraint created by a student to set up a value for an input parameter of a workflow component.

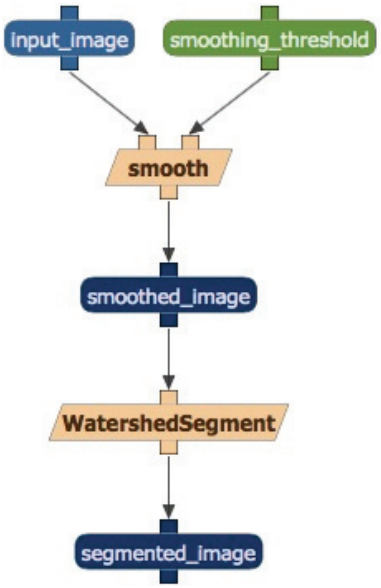

Figure 5: A workflow created by a student to explore how smoothing may affect the quality of segmentation.

Educational tools that use scientific datasets in a few domains have been investigated [Dooley et al 2006; Sendlinger et al 2008], but they do not embark in teaching big data analytics topics.

Conclusions

We have developed a course for non-programmers to learn about data science, and in particular concepts of parallel and distributed computing. The course allows the students to practice by using semantic workflows. The workflows capture complex multi-step data analysis methods, which include semantic constraints about their use. An intelligent workflow system, WINGS, uses those constraints to validate the workflows and assist the students to set up the analyses properly.

Acknowledgments

We gratefully acknowledge support from the National Science Foundation (NSF) with award ACI-1355475.

References

Columbia 2015]. http://www.journalism.columbia.edu/page/1058-the-lede-program-an-introduction-to-data-practices/906

Coursera 2015. https://www.coursera.org/course/datasci.

Deelman, E., Singh, G., Su, M. H., Blythe, J., Gil, Y., et al. "Pegasus: a Framework for Mapping Complex Scientific Workflows onto Distributed Systems." Scientific Programming Journal, vol. 13, 2005.

R. Dooley, K. Milfeld, C. Guiang, S. Pamidighantam and G. Allen. "From Proposal to Production: Lessons Learned Developing the Computational Chemistry Grid Cyberinfrastructure," 4(2), 2006.

Daniel Garijo and Yolanda Gil. "A New Approach for Publishing Workflows: Abstractions, Standards, and Linked Data." Proceedings of the 6th Workshop on Workflows in Support of Large-Scale Science (WORKS-11), held in conjunction with SC-11, 2011.

Gil, Yolanda. Syllabus for "Introduction to Computational Thinking and Data Science". (2015): figshare. Available from http://dx.doi.org/10.6084/m9.figshare.1614949 Retrieved 00:35, Dec 02, 2015 (GMT)

Gil, Y., Groth, P., Ratnakar, V., and C. Fritz. "Expressive Reusable Workflow Templates." Proceedings of the IEEE e-Science Conference, Oxford, UK, 2009.

Gil, Y.; Ratnakar, V.; Kim, J.; Gonzalez-Calero, P. A.; Groth, P.; Moody, J.; and Deelman, E. "Wings: Intelligent Workflow-Based Design of Computational Experiments." IEEE Intelligent Systems, 26(1). 2011.

Gil, Y.; Gonzalez-Calero, P. A.; Kim, J.; Moody, J.; and Ratnakar, V. "A Semantic Framework for Automatic Generation of Computational Workflows Using Distributed Data and Component Catalogs." Journal of Experimental and Theoretical Artificial Intelligence, 23(4), 2011.

"Time-Bound Analytic Tasks on Large Datasets through Dynamic Configuration of Workflows", Gil, Y., Ratnakar, V., et al. Proceedings of the 8th Workshop on Workflows in Support of Large-Scale Science (WORKS-13), held in conjunction with SC-13, 2013.

Hauder, M., Gil, Y. and Liu, Y. "A Framework for Efficient Text Analytics through Automatic Configuration and Customization of Scientific Workflows". Proceedings of

the Seventh IEEE International Conference on e-Science, Stockholm, Sweden, December 5-8, 2011.

Hauder, M.; Gil, Y.; Sethi, R.; Liu, Y.; and Jo, H. "Making Data Analysis Expertise Broadly Accessible through Workflows." Proceedings of the Sixth Workshop on Workflows in Support of Large-Scale Science (WORKS'11), held in conjunction with SC 2011, Seattle, WA, 2011.

Hull D, Wolstencroft K, Stevens R, Goble C, Pocock M, Li P, Oinn T. "Taverna: A Tool for Building and Running Workflows of Services", Nucleic Acids Research, Vol 34, 2006.

Kim, J.; Deelman, E.; Gil, Y.; Mehta, G.; and Ratnakar, V. "Provenance Trails in the Wings/Pegasus Workflow System." Concurrency and Computation: Practice and Experience, Special Issue on the First Provenance Challenge, 20(5). 2008.

Ludäscher, B. et al. "Scientific workflow management and the Kepler system." Concurrency and Computation: Practice and Experience. 18, 2006.

Moreau, L. and B. Ludäscher. Concurrency and Computation: Practice and Experience, Special Issue on the First Provenance Challenge, 20(5). 2008.

Sendlinger, S.C.; DeCoste, D.J.; Dunning, T.H.; Dummitt, D.A.; Jakobsson, E.; Mattson, D.R.; Wiziecki, E.N. "Transforming Chemistry Education through Computational Science," Computing in Science & Engineering, 2008.

C. Silva, E. Anderson, E. Santos, and J. Freire. "Using VisTrails and Provenance for Teaching Scientific Visualization." Computer Graphics Forum, 30(1), 2011.

Jeannette Wing. "Computational Thinking". Communications of the ACM vol. 49, no. 3, March 2006.

WINGS 2015. http://www.wings-workflows.org