

SoftLearn: Soft computing para minería de procesos en e-learning

Manuel Mucientes, Manuel Lama, Juan C. Vidal

Centro de Investigación en Tecnoloxías da Información (CITIUS)
Universidade de Santiago de Compostela (España)
{manuel.mucientes, manuel.lama, juan.vidal}@usc.es

Resumen. La finalidad del proyecto SoftLearn consiste en la aplicación de técnicas de soft computing, concretamente algoritmos evolutivos y reglas borrosas, así como la hibridación de estas técnicas con las tradicionales en minería de procesos, para lograr extraer flujos de trabajo completos, precisos, adaptativos y jerárquicos que representan las unidades docentes en e-learning. A la conclusión del proyecto el profesorado dispondrá de un conjunto de herramientas con las que podrá obtener, analizar y evaluar automáticamente el flujo de trabajo IMS LD que ha seguido implícitamente en la realización del curso. Estas herramientas serán una ayuda inestimable para que el profesor pueda entender qué ha ocurrido durante la realización de la unidad docente y valore qué decisiones debe tomar para mejorar y adaptar el aprendizaje a las características de los estudiantes.

1 Motivación

En la red de excelencia europea STELLARnet (www.stellarnet.eu), que agrupa a gran parte de las instituciones punteras en la investigación y desarrollo en tecnologías de e-learning, se han identificado tres grandes retos investigadores en esta área. Uno de estos tres retos es la orquestación del aprendizaje docente (*orchestrating learning*), en el que se incluyen problemas de investigación aún no resueltos, como el diseño de cursos colaborativos que sean más efectivos desde un punto de vista pedagógico o la evaluación de las actividades educativas desde el punto de vista de la consecución de los objetivos del curso.

El proyecto SoftLearn, iniciado en Enero de 2012, se enmarca dentro del reto de la orquestación del aprendizaje docente y pretende avanzar en el estado del arte en las herramientas que permiten a los profesores entender y evaluar el flujo de actividades que se realizan en un curso. Para alcanzar este objetivo en SoftLearn se propone el uso de técnicas de *soft computing*, concretamente algoritmos evolutivos y reglas borrosas, que se hibridarán con técnicas tradicionales de *minería de procesos*, ya que se considera que la orquestación del aprendizaje es un proceso. Así, considerando únicamente los registros de las actividades realizadas por los estudiantes a través de una plataforma de e-learning, SoftLearn se centrará en determinar automáticamente la coordinación de las actividades docentes del curso (*flujo de aprendizaje*), quiénes y en qué condiciones las han ejecutado (*roles*), y qué recursos docentes se usaron en cada actividad (*contenidos y servicios*).

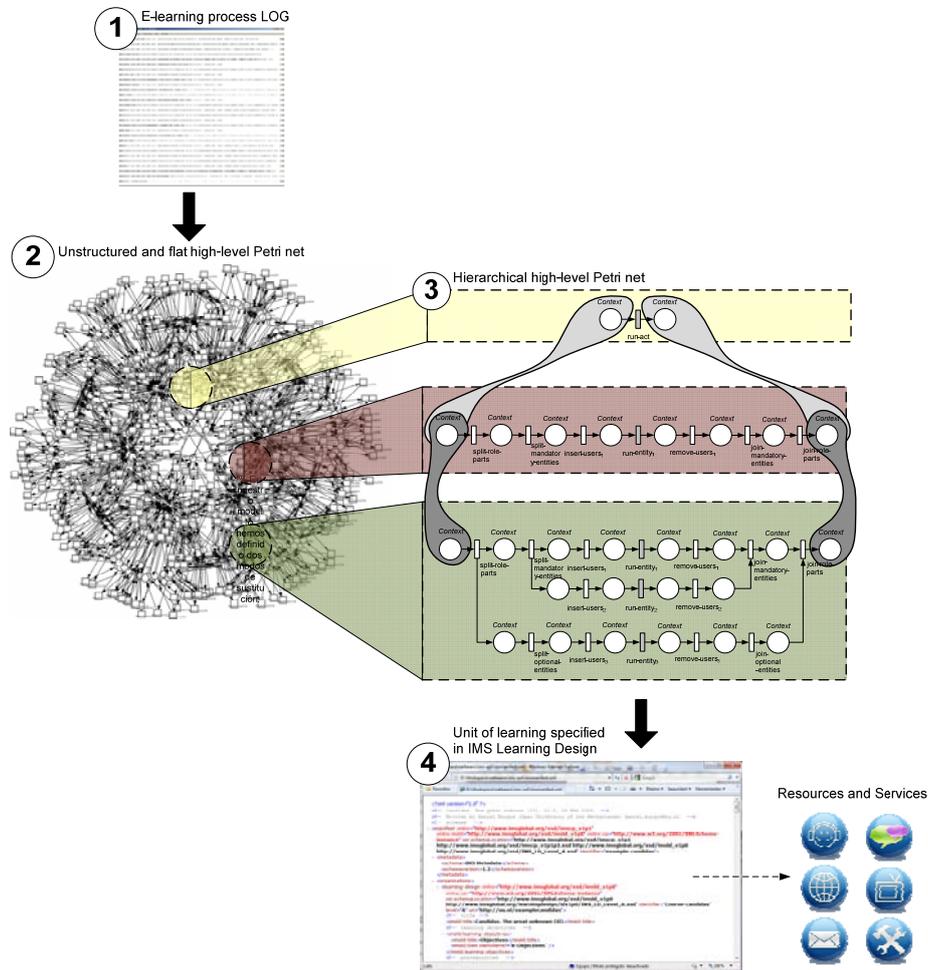


Fig 1. Minería de procesos para la generación de flujos de trabajo completos, precisos y jerárquicos que modelan unidades docentes basadas en la especificación IMS LD.

La minería de procesos [1] es el conjunto de técnicas y procedimientos que tratan de reconstruir de forma automática el modelo observado en los ficheros de registros. Estas técnicas han sido ampliamente aplicadas en el ámbito industrial y de servicios [2,5], pero no así en el campo del e-learning, en general, y en el diseño del flujo de las actividades de un curso, en particular. En SoftLearn este diseño se representa a través de IMS Learning Design (IMS LD) [2], un estándar *de facto* para el diseño del flujo de actividades educativas (es decir, la orquestación del aprendizaje docente), que se modela formalmente a través de redes de Petri jerárquicas [3,4]. Dado que IMS LD impone restricciones al flujo de trabajo (jerarquización y condiciones sobre las actividades), para reducir la complejidad de los algoritmos de generación y evaluación automática del flujo de trabajo, en SoftLearn se propone dividir el problema en dos

partes: primero, se obtiene la red de Petri aplanada con la que se representa la coordinación de las actividades docentes, y después se creará la red de Petri jerárquica que refleja el esquema de ejecución de la especificación IMS LD.

Teniendo esto en cuenta, la Fig. 1 ilustra la estrategia que se seguirá en SoftLearn para generar y evaluar automáticamente los flujos de trabajo basados en IMS LD. Cada uno de los elementos IMS LD (4 en Fig. 1) se representa mediante una red de Petri que coordina la ejecución de los subelementos de los que se compone, los cuales se sustituyen a su vez por una red de Petri. De esta forma, se irá creando la red de Petri jerárquica (3 en Fig. 1) que modela formalmente los flujos docentes IMS LD y que tiene asociada una red de Petri aplanada (2 en Fig. 1) cuya ejecución proporciona el mismo resultado y que refleja el comportamiento de los profesores y estudiantes que ha sido recogido en los registros de actividades.

El resto del artículo se estructura de la siguiente forma: en la sección 2 se describe el estado actual de las técnicas de minería de procesos en relación a su aplicación al problema de generación y evaluación de flujos de trabajo en e-Learning; y en la sección 3 se describen los resultados esperados del proyecto.

2 Minería de Procesos

Un proceso se define como un conjunto de actividades o tareas cuya ejecución coordinada produce un determinado servicio o producto [5]. El control de esa ejecución se modela a través de lo que se conoce como flujo de trabajo, en el que intervienen un conjunto de usuarios y servicios que son los encargados de realizar las actividades que forman parte del mismo. La minería de procesos consiste en la extracción de conocimiento relacionado con un proceso a partir de los registros de seguimiento (o ficheros de eventos) capturados por un sistema de información [1] En esta área de investigación se distinguen los siguientes problemas:

1. *Evaluación del proceso*, en el que se comprueba si las actividades que están incluidas en los registros se corresponden con aquéllas que componen el flujo de trabajo que ha sido ejecutado. A partir de esta comprobación se identifican caminos de control que no han sido ejecutados y se propone una extensión y/o modificación del flujo de trabajo original.
2. *Generación del proceso*, en el que se construye automáticamente un modelo del proceso cuya ejecución reproduce el comportamiento observado en los registros del sistema. Para ello es necesario determinar las dependencias entre las actividades que intervienen en el proceso y generar un flujo de trabajo basado en un modelo formal de concurrencia (como las redes de Petri [6]).

Los dos aspectos fundamentales que es necesario tener en cuenta al aplicar minería de procesos a un registro son la *completitud* y la *precisión* del flujo de trabajo obtenido. Un flujo de trabajo es completo si es capaz de reproducir todos los caminos de control del fichero de eventos. Además, se dice que el flujo es preciso si no puede reproducir más caminos de control que los que hay en el registro. Las mayores dificultades que deben abordar las técnicas en minería de procesos para obtener flujos de trabajo completos y precisos son debidas a: la duplicidad de actividades (múltiples transiciones con la misma etiqueta); las actividades ocultas (no aparecen en el registro pero deben

ser incluidas para enrutamiento); y la sensibilidad al ruido (comportamiento poco frecuente en el registro).

Algunas propuestas de generación de flujos de trabajo a partir de registros son [7,8,9], pero en todas ellas los resultados obtenidos presentan importantes limitaciones y no resuelven los problemas ya apuntados. Una de las técnicas de referencia en el ámbito de generación de procesos es la propuesta en [10,11], donde se sigue una aproximación heurística basada en una métrica de frecuencia que aborda los problemas de completitud y ruido. Este algoritmo ha sido mejorado [12], pero todavía no es suficientemente robusto al ruido y no es capaz de capturar actividades ocultas. Por otra parte, en [13] se aborda el descubrimiento de flujos de trabajo jerárquicos aplicando técnicas de minería de datos (se utiliza un algoritmo de clustering para agrupar las actividades). Sin embargo, en esta aproximación requieren que la paralelización y la sincronización estén balanceadas. En cuanto a las técnicas empleadas en la evaluación de flujos de trabajo (una vez que han sido obtenidos), son las propias de la minería de datos (Aalst, 2007), utilizando fundamentalmente algoritmos de clasificación convencionales (árboles de decisión, etc.).

Por otra parte, la generación de flujos de trabajo mediante técnicas de soft computing es reciente. Los primeros trabajos [14,15,16] combinan algoritmos genéticos y redes de Petri, pero no obtienen el flujo de trabajo a partir de un fichero de eventos. El trabajo más significativo en el que se ha abordado este problema es [17]. En este artículo se solucionan algunos de los problemas de aproximaciones tradicionales en minería de procesos y, además, se hace un análisis en profundidad de las métricas para determinar la calidad de los flujos de trabajo obtenidos. Sin embargo, no se aborda ningún tipo de jerarquización del flujo de trabajo, y tampoco se obtienen flujos de trabajo adaptativos (con condiciones de ejecución en algunas de sus actividades).

La jerarquización de flujos de trabajo mediante soft computing es un campo poco explorado, siendo [18] el trabajo más significativo. Sin embargo, debido a la necesidad de incrementar la interpretabilidad de los flujos de trabajo obtenidos, este es un campo prometedor. En SoftLearn, debido a las características de la especificación IMS LD es imprescindible abordar la jerarquización para poder modelar correctamente la unidad docente.

2.1 Minería de Procesos en E-Learning

La aplicación de la minería de procesos en el ámbito educativo es una línea de investigación emergente que centra su atención en modelar, generar y evaluar los procesos que los estudiantes y profesores llevan a cabo durante el desarrollo de una unidad o módulo docente. En este contexto la granularidad de lo que se entiende por proceso abarca desde el comportamiento de los estudiantes cuando realizan una actividad educativa hasta el flujo de trabajo que coordina la ejecución de todas las actividades de una unidad docente. Actualmente, la minería de procesos está siendo usada para extraer el comportamiento de los estudiantes a la hora de navegar por las preguntas de un test [19], obtener las actividades realizadas por los estudiantes que crean un documento de texto de forma colaborativa [20], o identificar el flujo de trabajo que sigue un grupo de estudiantes cuando colaboran entre sí a través de una herramienta de comunicación determinada [21]. Sin embargo, no existe ningún trabajo que aborde la

generación y/o evaluación automática del flujo de trabajo asociado a las actividades educativas modeladas a través de la especificación IMS LD. Esto se debe a:

1. Existen muy pocas aproximaciones que desde la perspectiva de los flujos de trabajo modelen el esquema de control definido en la especificación IMS LD [22,23, 24]. De estas aproximaciones únicamente OPENET LD [25,4] representa formalmente dicho esquema de control, para lo cual usa redes de Petri jerárquicas. Esta representación formal es imprescindible para poder aplicar las técnicas de minería de procesos, dado que a través de ella se determina si las soluciones obtenidas verifican las propiedades de control que son deseables en los flujos de trabajo.
2. Los algoritmos de minería de procesos propuestos en la literatura no están pensados para tratar con la complejidad de los flujos de trabajo asociados a unidades docentes IMS LD, en los cuales (i) existe una estructura jerárquica de control que es un reflejo directo de la estructura de los componentes de IMS LD; y (ii) aparecen condiciones o reglas que restringen las actividades docentes (es decir, los caminos de control) que deberán ser realizadas por los estudiantes y profesores.

En el proyecto SoftLearn se utilizará OPENET LD [4], un motor de ejecución de unidades docentes IMS LD que a su vez está basado en un motor de redes de Petri [26] y que genera un registro de eventos con las actividades realizadas en el transcurso de la unidad docente. Las características de OPENET LD dan viabilidad al proyecto, dado que permiten establecer un criterio claro de validación de los algoritmos de minería de procesos: estos algoritmos deberán obtener la red de Petri jerárquica que sigue OPENET LD al coordinar la realización de la unidad docente por parte de los estudiantes y profesores.

3 Resultados Esperados

El objetivo de SoftLearn es la aplicación de técnicas de soft computing para extraer flujos de trabajo completos, precisos, adaptativos y jerárquicos que modelen las unidades docentes en e-learning. Asimismo, se propondrán algoritmos para el análisis y evaluación de dichos flujos de trabajo, así como para la recomendación de usuarios a las actividades educativas. Más concretamente, los resultados que se esperan son los siguientes:

- Generación automática del flujo de trabajo de una unidad docente: se hibridarán técnicas básicas de minería de procesos con algoritmos evolutivos para extraer las redes de Petri que modelan las unidades docentes. En la actualidad se han implementado una serie de algoritmos que extraen con éxito el flujo de aprendizaje a partir de los registros de actividad generados por OPENET LD. La Fig. 2 muestra la red de Petri obtenida para un curso dado.
- Jerarquización de flujos de trabajo asociados a unidades docentes IMS LD: partiendo de redes de Petri planas, se obtendrán redes de Petri jerárquicas que modelen formalmente el flujo de actividades educativas de acuerdo con IMS LD.
- Análisis y evaluación de flujos de trabajo jerárquicos: se usarán algoritmos de clasificación basados en la hibridación de técnicas de soft computing (algoritmos evolutivos y lógica borrosa) para analizar y evaluar los flujos de trabajo jerárquicos.

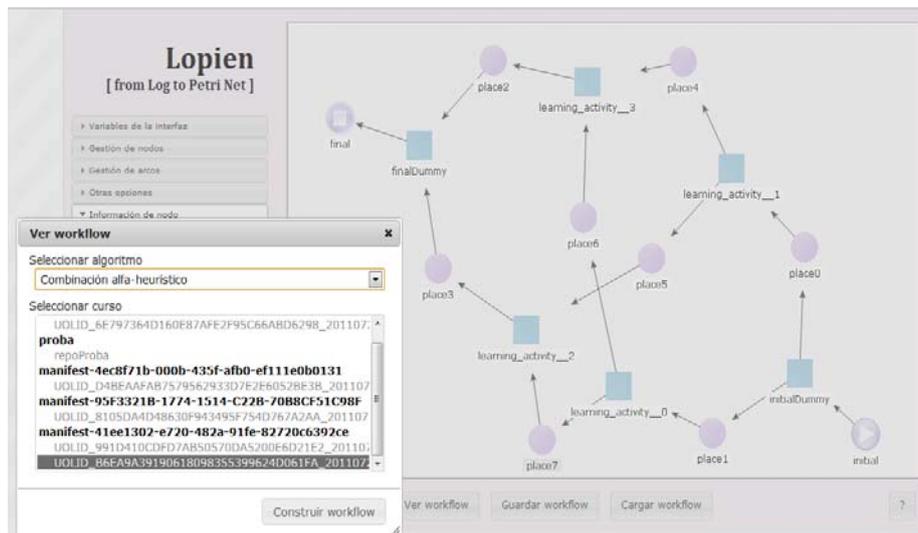


Fig. 2. Red de Petri generada a partir de los registros de una unidad de aprendizaje realizada en OPENET LD [4]

- Recomendación de usuarios relevantes para la realización de las actividades educativas: se desarrollarán algoritmos de recomendación basados en la hibridación de técnicas de soft computing y de inteligencia colectiva.

Los algoritmos desarrollados en el proyecto SoftLearn son componentes de gran utilidad dentro de un sistema de gestión de cursos, al permitir al profesor realizar un análisis de la unidad docente de forma totalmente automática. Esto facilita, por tanto, la modificación del flujo seguido en la unidad docente y permite adaptar el curso a las características de los alumnos. Estos algoritmos, además, se codificarán en sendos módulos software que se integrarán como plugins a varias de las plataformas de e-learning más usadas en la actualidad como Moodle o Claroline.

Agradecimientos. Los autores desean agradecer al Ministerio de Economía y Competitividad por el soporte financiero a través del proyecto SoftLearn (TIN2011-22935). Manuel Mucientes está financiado por el programa *Ramón y Cajal* del Ministerio de Economía y Competitividad.

Referencias

1. Aalst, W. van der, Dongen, B. van, Herbst, J., Maruster, L., Schimm, G., Weijters, A. (2003): Workflow mining: A Survey of issues and approaches. *Data and Knowledge Engineering*, 47(2):237–267.
2. Koper, R., Tattersall, C. (eds.) (2005): *Learning Design – A handbook on modeling and delivering networked education and training*. Springer-Verlag, Berlin.

3. Vidal, J.C., Lama, M., Sánchez, E., Bugarín, A. (2008): Application of Petri nets on the execution of IMS learning design documents. Proceedings of the 3rd European Conference on Technology Enhanced Learning (EC-TEL 08). Times of convergence: Technologies across learning contexts, Lecture Notes in Computer Science, vol. 5192, Maastricht, Netherlands, 461–466.
4. Vidal, J.C., Lama, M., Bugarín, A. (2012): Petri net-based engine for adaptive learning. Expert Systems with Applications. Aceptado.
5. Aalst, van der W., Weijters, A., Maruster, L. (2004): Workflow mining: discovering process models from event logs. IEEE Transactions on Knowledge Data Engineering, 16(9):1128–1142.
6. Murata, T. (1989): Petri nets: Properties, analysis, and applications. Proceedings of the IEEE, 77(4):541–580.
7. Cook, J., Du, Z., Liu, C., Wolf, A. (2004): Discovering models of behavior for concurrent workflows. Computers in Industry, 53(3):297-319.
8. Herbst, J. (2000): Dealing with concurrency in workflow induction. Proceedings of the 7th European Concurrent Engineering Conference, Leicester (ECEC 2000), Reino Unido, 169–174.
9. Herbst, J., Karagiannis, D. (2004): Workflow mining with InWoLvE. Computers in Industry 53(3), 245-264.
10. Aalst, W. van der, Reijers, H., Weijters, A., Dongen, B. van, Medeiros, A., Song, M., Verbeek, H. (2007): Business process mining: An industrial application. Information Systems, 32:713-732.
11. Weijters, A., Aalst, W. van der (2003): Rediscovering workflow models from event-based data using little thumb. Integrated Computer-aided Engineering, 10(2), 151–162.
12. Wen, L., Wang, J., Sun, J. (2006): Detecting implicit dependencies between tasks from event logs. Lecture Notes in Computer Science, 3841, 591–603.
13. Schimm, G. (2004): Mining exact models of concurrent workflows. Computers in Industry, 53(3), 265-281.
14. Malpathak, S., Saitou, K., Qvam, H. (2002): Robust design of flexible manufacturing systems using colored Petri net and genetic algorithm. Journal of Intelligent Manufacturing, 13(5):339-351.
15. Mauch, H. (2003): Evolving Petri nets with a genetic algorithm. Proceedings 2003 International Conference on Genetic and Evolutionary Computation Conference (GECCO 2003), Lecture Notes in Computer Science, vol. 2724, 1810–1811.
16. Nummela, J., Julstrom, B. (2005): Evolving Petri nets to represent metabolic pathways. Genetic and Evolutionary Computation Conference (GECCO 2005), ACM, 2133-2139.
17. Medeiros, A., Weijters, A., Aalst, W. van der (2007): Genetic process mining: an experimental evaluation. Data Mining and Knowledge Discovery, 14:245–304.
18. Moore, J., Hahn, L. (2004): An improved grammatical evolution strategy for hierarchical Petri net modeling of complex genetic algorithms. Applications of evolutionary computing, Evo Workshops 2004, Lecture Notes in Computer Science, vol. 3005, 63–72.
19. Pechenizkiy, M., Trčka, N., Vasilyeva, E., Aalst, W. van der, De Bra, P. (2009): Process mining online assessment data. Proceedings of the 2th International Conference on Educational Data Mining (EDM 2009), Córdoba, Spain, 41–50.

20. Reimman, P., Frerejean, J., Thompson, K. (2009): Using process mining to identify models of group decision making in chat data. Proceedings of the 9th International Conference on Computer Supported Collaborative Learning (CSCL 2009), Rhodes, Greece, 98–107.
21. Southavilay, V., Kalina Yacef, K., Calvo, R.A. (2009): WriteProc: A framework for exploring collaborative writing process. Proceedings of the 14th Australasian Document Computing Symposium (ADCS 09), Sydney, Australia, 129–136.
22. Karampiperis, P., Sampson, D. (2007): Towards a common graphical language for learning flows – Transforming BPEL to IMS Learning Design level A representations. Proceedings of the 7th IEEE International Conference on Advanced Learning Technologies (ICALT 2007), Niigata, Japón, 798–800.
23. Vantrois, T., Peter, Y. (2003): COW – A flexible platform for the enactment of learning scenarios. Proceedings of the 9th International Workshop on Groupware (CRIWG 2003), Groupware: Design, implementation, and use. Lecture Notes in Computer Science, vol. 2806, Grenoble, France, 168–182.
24. Molina, A.I., Jurado, F., de la Cruz, I., Redondo, M.A., Ortega, M. (2009): Tools to support the design, execution and visualization of instructional designs. Proceedings of the 6th International Conference on Cooperative Design, Visualization, and Engineering (CDVE 09). Lecture Notes in Computer Science, vol. 5738, Luxemburgo, Luxemburgo, 232–235.
25. Vidal, J.C., Lama, M., Sánchez, E., Bugarín, A., Novegil, A. (2009): OPENET LD: An ontology-based Petri net engine to execute IMS LD units of learning. Proceedings of the 9th International Conference on Advanced Learning Technologies (ICALT 09), Riga, Letonia, 499–503.
26. Vidal, J.C., Lama, M., Bugarín, A. (2010): OPENET – Ontology-based engine for high-level Petri nets. Expert Systems with Applications, 37(9): 6493–6509.