

Herramienta de análisis y minado de workflows en ejercicios de laparoscopia

David Nieves , Cèsar Ferri , José Hernández-Orallo , Carlos Monserrat , and María José Ramírez-Quintana

Departamento de Sistemas Informáticos y Computación
Universitat Politècnica de València, Spain
{*daniecor, cferri, jorallo, cmonserr, mramirez*}@dsic.upv.es

1 Introducción

Los avances tecnológicos en aprendizaje automático y visión por ordenador han brindado la oportunidad de desarrollar sistemas de supervisión automática que puedan ayudar a los cirujanos noveles a mejorar sus destrezas mediante la detección y corrección temprana de errores durante sus primeros pasos en la cirugía laparoscópica [16] [15]. En muchos casos, los motivos que se encuentra detrás de una mala intervención quirúrgica se hallan en una adquisición deficiente de estas destrezas, lo que puede acarrear postoperatorios más largos o, en los casos más extremos, la muerte del paciente [3] [11]. Antes de que el sistema pueda supervisar, y para que esta supervisión sea eficaz, el supervisor automático debe de aprender todas las formas correctas en las que la rutina de cirugía se puede desarrollar. Descartada la programación manual, debido al coste de tiempo y esfuerzo que supone, la opción más eficiente según esas restricciones es que el sistema aprenda partir de ejemplos demostrativos realizados por cirujanos expertos. El coste de grabar estas demostraciones, por parte de los expertos, es exigente para ellos, debido alto nivel de pericia que requieren y el tiempo que demandan. Por tanto, se estima que sean muy pocos los ejemplos con los que el sistema debe de generalizar y aprender las formas correctas (i.e., workflows) en las que un ejercicio de laparoscopia se debe de realizar. Esto implica conocer el conjunto de actividades fundamentales que se requieren, así como determinar el orden o dependencias que puedan existir entre ellas. En definitiva, extraer la esencia de la tarea que subyace dentro de las demostraciones que se proporcionan al sistema.

La dificultad inherente de los ejercicios de laparoscopia provoca que en muchas ocasiones se realicen más actividades quirúrgicas de las estrictamente necesarias [5]. Por otra parte, los cirujanos más experimentados suelen desarrollar una manera propia o “estilo” cuando realizan ciertas rutinas de cirugía (e.g., sutura), lo que puede dificultar la generalización cuando se disponen de muy pocos ejemplos con los que aprender la forma de la tarea [2]. Esto lleva a que la agregación completa de todos los ejemplos, sin ningún tipo de filtrado de información, tienda rápidamente a una representación conceptual de la tarea, o *workflow*, que no proporcione ninguna abstracción significativa, y que sea cuasi imposible de interpretar visualmente. En este trabajo, por tanto, presentamos una herramienta desarrollada en R, que permite la extracción de workflows subyacentes en conjuntos de ejemplos demostrativos de ejercicios de laparoscopia. Esta herramienta esta compuesta por un método novedoso de minado de workflows y una aplicación interactiva con la que poder probarlo con ejemplos reales de laparoscopia.

2 Origen de datos

En la actualidad, no existen muchos datasets públicos de ámbito médico que estén especializados en ejercicios de laparoscopia. Dentro de los pocos que hay disponibles, posiblemente el más popular y completo sea el dataset JIGSAWS [7] (JHU-ISI Gesture and Skill Assessment Working Set), el cual ha sido registrado a través del sistema de cirugía robótica *da Vinci*¹. Originalmente concebido para el desarrollo de aplicaciones centradas en el análisis de gestos quirúrgicos y evaluación automática de destrezas, hoy en día se ha convertido en un *benchmark* público para el estado del arte actual en métodos de reconocimiento de actividades quirúrgicas en laparoscopia [2]. Los tipos de ejercicios de entrenamiento que el dataset contiene son los siguientes:

- **Sutura:** El cirujano debe de realizar cuatro pasadas de sutura sobre un tejido sintético con una incisión marcada. Sosteniendo la aguja con una pinza, se inserta la aguja a un lado de la incisión indicada en

¹<https://www.davincisurgery.com/da-vinci-surgery/da-vinci-surgical-system/>

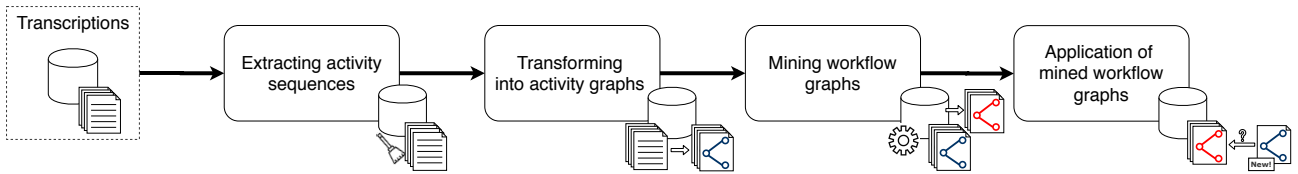


Figure 1: Proceso de minado de workflows a partir de las transcripciones de ejercicios de laparoscopia.

el tejido y se saca por el otro extremo con el otro instrumental, devolviéndose la aguja posteriormente a la primera pinza para realizar la siguiente pasada. Existe un total de 39 ejemplos registrados de este ejercicio.

- **Anudado:** Esta tarea consiste en realizar dos nudos sencillos alrededor de un tubo de goma usando las dos pinzas. Se disponen de 36 ejemplos de esta tarea.
- **Paso de aguja:** El ejercicio consiste en pasar la aguja de sutura a través de cuatro aros pequeños metálicos, colocados a cierta altura, de derecha a izquierda ayudándose con las dos pinzas. En total, el dataset consta de 28 ejemplos para esta tarea.

El dataset cuenta con 8 cirujanos, identificados con letras de la *A* a la *H*, con diferentes niveles de experiencia en cirugía robótica y medida en horas de simulador (i.e., E-expert (>100hrs), I-intermediate (10-100hrs), Novice (<10hrs)). Cada cirujano ha registrado un máximo de 5 ejemplos o “trials” de cada ejercicio, aunque el número de ejemplos está desbalanceado debido a que no todos los cirujanos participaron en todos los ejercicios y ni todos ellos registraron un total de 5 intentos. Cada trial se identificará por el identificador del cirujano junto con el número del intento que hizo (e.g., B004). El dataset ofrece 3 recursos de datos por cada trial: las cinemáticas que genera el instrumental durante el trial, el vídeo del interior del simulador laparoscópico durante el ejercicio y la transcripción de los gestos quirúrgicos que se han realizado. Estas transcripciones son anotadas por cirujanos expertos tras observar el trial, usando para ello un vocabulario de actividades quirúrgicas adaptado por los autores del dataset. Una transcripción constará de la secuencia de actividades que se ha hecho, acompañada cada una por el instante de vídeo en el que se inicia y finaliza. Los trials están puntuados usando una medida que indica la calidad y habilidad técnica (i.e., Global Rating Score o *GRS*) demostrada por el cirujano durante el ejercicio y el resultado obtenido finalmente.

En este trabajo usaremos las transcripciones que proporciona JIGSAWS para aplicar nuestro proceso de minado. Estas transcripciones pueden asimilarse a los ficheros de registro de eventos (i.e., ficheros logs) que son empleados en otros trabajos de minado de workflows disponibles en la literatura [6] [8] [1]. Para nuestro trabajo hemos extendido la información que ya proporcionaba el dataset añadiendo la división por cuartiles en función de la puntuación *GRS*. Esto está motivado porque, entre otros objetivos, en nuestro estudio actual nos interesa determinar el criterio de selección de ejemplos “ricos” que nos permita identificar los diferentes estilos, en forma de workflows, para cada ejercicio y que la generalización sobre ejemplos no vistos sea la mayor posible, esto es, el mayor número de ejemplos representados por estos workflows.

3 Proceso de minado de workflows

El proceso que seguimos para la extracción de workflows puede observarse en la Figura 1. Partimos siempre de un conjunto de transcripciones de un ejercicio concreto, por ejemplo, sutura. Aplicamos un preprocesado sobre cada transcripción para extraer las secuencias de actividades empleadas durante esa ejecución. Remarcar que no existe solapamiento temporal entre estas actividades, ni tampoco hay registro de eventos que no estén en el vocabulario antes indicado. Debido a que los trials pueden discrepar en la actividad de inicio y de final, añadimos dos actividades sintéticas en los extremos de cada secuencia (i.e., vértices *S* y *F*). Este paso facilitará la alineación posterior de los ejemplos durante el proceso de minado. Seguido de la extracción de estas secuencias, las transformamos a una representación en forma de grafo (i.e., *activity graph*), donde cada actividad corresponderá con un vértice etiquetado con el identificador de la actividad, y cada relación de precedencia entre actividades se transformará en una arista dirigida con la dirección de la actividad precedente a la consecuente. Remarcar que una actividad repetida dentro de la secuencia tendrá correspondencia con un único vértice del grafo resultante. Por contra, las transiciones repetidas entre dos actividades se sumarán y asignarán como peso a la arista dirigida que las une.

Finalmente, los grafos son introducidos en nuestro proceso de minado de workflows. Este proceso se basa en un algoritmo iterativo que está inspirado en el funcionamiento de sistemas lógico-inductivos *top-down* basados en cobertura de ejemplos [9]. Por este motivo, y a diferencia de otros métodos que obtienen una única representación de un procedimiento médico para todos los ejemplos dados [4] [17], nuestro método de minado es tolerante a

las diversidad de maneras en las que un cirujano puede realizar un ejercicio determinado. De este modo, durante el proceso podemos sintetizar y extraer diversos workflows, pudiendo cada uno de ellos representar a un subconjunto de ejemplos de entrada que expresen una forma determinada de llevar a cabo el ejercicio de laparoscopia. A continuación, describimos los pasos principales de nuestro proceso de minado:

1. **Obtención del grafo agregado:** Primero, los grafos de entrada son agregados en un único grafo al que denominamos G^+ . La agregación conlleva a que se sumen los pesos de aquellas aristas que sean comunes en los ejemplos.
2. **Selección del umbral:** Para separar el workflow subyacente de las actividades y transiciones superfluas buscaremos un valor de umbral, relativo al peso de las aristas, que permita hacer ese filtrado. Esto significa que, dado ese valor de umbral, solo permanecerán en el grafo resultante aquellas aristas que tengan un peso igual o superior a dicho valor. De lo anterior se puede deducir que la elección del umbral no es trivial. Un umbral demasiado alto puede “romper” el grafo y, por tanto, dar con un workflow inservible para representar la tarea. Por el contrario, un umbral demasiado bajo puede propiciar que se mantengan transiciones y actividades intrascendentes para aprender la esencia de la tarea. Nuestro proceso tiene especial cuidado de lo anterior y calcula automáticamente el umbral óptimo para lograr un balance entre ambos casos. Empezando con un umbral igual al peso máximo que está presente en el grafo, descendemos éste hasta obtener un valor de umbral que devuelve un grafo válido ².
3. **Aplicación del umbral:** Una vez calculado el umbral óptimo, lo aplicamos sobre las aristas del grafo agregado. Se eliminarán las aristas que no sobrepasen dicho umbral. En caso de que un vértice quede completamente desconectado del grafo, éste quedará descartado también. Esto llevará a descartar actividades que no son necesarias para aprender dicha forma de realizar la tarea. El grafo resultante de la aplicación del umbral será lo que consideraremos workflow, el cuál identificaremos con una letra mayúscula asignada en orden ascendente (A, B, C, \dots). En [8] puede encontrarse un criterio de filtrado similar al que proponemos nosotros, pero más laxo a la hora de descartar actividades del workflow resultante. Lo anterior hace que obtengan representaciones menos compactas que las obtenidas por nuestro método.
4. **Cobertura de ejemplos:** En este último paso comprobamos en qué ejemplos se encuentra presente el workflow hallado. En caso de que el workflow esté contenido en un ejemplo, dicho ejemplo será categorizado con el identificador del workflow y se retirará del conjunto de ejemplos del proceso. Si ningún ejemplo ha sido cubierto, se volverá al Paso 2 bajando el umbral al siguiente peso del grafo agregado. Una vez comprobada la cobertura del workflow con todos los ejemplos de la iteración, el proceso vuelve al Paso 1 con solo aquellos ejemplos que no fueron cubiertos.

Con el conjunto de workflows que extraemos podemos identificar nuevos ejemplos no vistos por el sistema en caso de que estos estén presentes en ellos, es decir, comprobar la generalización de los workflows minados. En esta fase de aplicación, y dado un ejemplo en forma de grafo, comprobaremos de nuevo si el workflow está contenido dentro del ejemplo. El tal caso, ejemplo será categorizado con el identificador del workflow.

4 Implementación de la herramienta

Tanto para implementar el método de minado, como para desarrollar una aplicación web interactiva ³ para poder estudiar su comportamiento con las tareas del dataset JIGSAWS, hemos usado el lenguaje de programación R [14]. Para la extracción y manipulación de las secuencias de gestos quirúrgicos usamos el paquete `dplyr`⁴. Para su transformación a grafos dirigidos e implementación de las operaciones del proceso de minado hemos usado el paquete `igraph`⁵. En el caso de la aplicación web desarrollada, hemos usado `Shiny` ⁶, junto con los paquetes `shinydashboard` ⁷ y `visnetwork` ⁸, para mejorar así el aspecto visual de la aplicación en general, y añadir funcionalidades de interacción relativa a los grafos. Finalmente, para las estadísticas y gráficos de la evolución del proceso de minado hemos usado los paquetes `DT` ⁹ y `ggplot2` ¹⁰.

La aplicación web implementa en su “backend” el método de minado aquí descrito, añadiéndole la interactividad que proporciona Shiny para que pueda ser usado sobre los ejemplos que ofrece JIGSAWS. Por ejemplo,

²Consideramos que un grafo es válido si todo nodo que es alcanzado desde el vértice de inicio S , alcanza, a su vez, el vértice final F . De esta forma aseguramos que la representación de la tarea es completa de principio a final y que no quedan actividades “colgantes” en el workflow (i.e., vértices que tienen una arista de entrada pero no de salida o viceversa)

³<https://safe-tools.dsic.upv.es/shiny/SurgicalWorkflowMining/>

⁴<https://cran.r-project.org/web/packages/dplyr/index.html>

⁵<http://igraph.org/r/>

⁶<https://shiny.rstudio.com/>

⁷<https://rstudio.github.io/shinydashboard/>

⁸<https://datastorm-open.github.io/visNetwork/>

⁹<https://rstudio.github.io/DT/>

¹⁰<https://ggplot2.tidyverse.org/>

una vez el usuario selecciona alguno de los ejercicios de laparoscopia del dataset, la aplicación le permite ver su descripción detallada, explorar la secuencia original de actividades y el *activity graph* para cada uno de sus *trials*, y finalmente probar el proceso de minado que proponemos escogiendo el subconjunto que se desee de estos trials. Una vez se lanza el proceso de minado, la aplicación muestra los workflows obtenidos y diferentes estadísticas al respecto. Para cada workflow se puede visualizar su representación como *activity graph*, junto con el grafo agregado G^+ a partir del que se obtuvo. De esta forma se puede apreciar la síntesis de información y analizar el comportamiento del proceso para los ejemplos dados. Por otra parte, se muestra información relativa a la cobertura de ejemplos tanto en la fase de minado como en la aplicación de los workflows sobre el resto de ejemplos no vistos. Por último, se muestra la evolución del filtrado de aristas y vértices a lo largo del proceso, pudiéndose comparar, a nivel cuantitativo, la diferencia que hay entre el workflow minado con respecto al grafo agregado de la iteración.

5 Resultados preliminares

En una primera exploración de los resultados obtenidos para el ejercicio de sutura usando la herramienta desarrollada, hemos observado que, al contrario de lo que se podría pensar, los *trials* registrados por los expertos tienen puntuaciones relativamente bajas. Si comparamos los resultados obtenidos a partir de los ejemplos mejor puntuados (cuartil $Q1$), con los obtenidos a partir de los ejemplos de expertos, se observa que estos últimos contienen más transiciones y actividades. Lo anterior repercute también en que los workflows minados a partir de los expertos son menos compactos que los obtenidos usando los mejor puntuados. Si analizamos la cobertura de los workflows de unos y otros, podemos apreciar que los expertos cubren menos ejemplos no visto con los workflows más compactos. Esto nos lleva a pensar que los ejemplos de los expertos estén afirmativamente sesgados por un estilo particular a la hora de realizar la sutura que reduzca su generalización sobre el resto de formas, tal y como hacen constar en [2]. A pesar de ello, hemos observado que nuestro criterio de filtrado reduce considerablemente el exceso de transiciones aunque los ejemplos contengan muchas que sean innecesarias (obsérvese para ello las diferencias entre los primeros workflow con respecto a sus G^+ en los histogramas comparativos de las iteraciones del proceso). Si comparamos los resultados de los ejemplos mejor puntuados y de los expertos nuevamente a este respecto, podemos ver que, a pesar de que los G^+ de los expertos son más grandes, los primeros workflows que se obtienen en cada casos (expertos y mejor puntuados) son muy similares en tamaño. Por último, se ha observado que el proceso de minado tiende a obtener un último workflow que recoge toda aquella información poco frecuente y, presumiblemente innecesaria, que esta presente en los ejemplos.

6 Conclusiones y trabajo futuro

En este trabajo hemos presentado una herramienta para minar workflows subyacentes a partir de un conjunto de ejemplos en forma de secuencias de actividades que representan ejercicios de laparoscopia. La herramienta, compuesta por el método novedoso de minado de workflows y una aplicación web interactiva para poder probarlo sobre datos reales extraídos del dataset JIGSAWS, se ha mostrado eficaz abstrayendo múltiples workflows a partir de los ejemplos que se introducen, independientemente del “ruido” que provocan las actividades y transiciones innecesarias que contienen.

Actualmente tenemos varias líneas de continuación de lo presentado en este trabajo. Pretendemos integrar la aplicación que aquí presentamos con otros desarrollos previos que hemos realizado [12] [10], con el objetivo de obtener el módulo de aprendizaje del futuro sistema de supervisión de ejercicios de laparoscopia. De esta manera, un cirujano podrá emplear esta herramienta sobre los ejemplos que registre y pueda gestionar así el conocimiento que adquiere el sistema de forma rápida y cómoda. También estamos interesados en la validación de los workflows que podemos obtener dado un dominio de aplicación. Contemplamos automatizar esta validación usando las descripciones textuales que se proporcionan a los estudiantes. Esta información es rica en aspectos como condiciones y restricciones relativas a la tarea que pueden ser costosas de registrar a partir de las demostraciones directas al sistema (e.g., evitar el contacto con el instrumental con cierto elemento del set del ejercicio). Por último, pensamos adaptar y extender nuestra herramienta de minado de workflows a otros dominios de supervisión y asistencia, como puede ser la elaboración de recetas de cocina [13].

Agradecimientos

Este trabajo ha sido parcialmente financiado por los fondos europeos FEDER y por el MINECO (TIN2014-61716-EXP y TIN 2015-69175-C4-1-R), y por la Generalitat Valenciana (PROMETEOII/2015/013). David Nieves está financiado por el MINECO a través de una beca FPI (BES-2016-078863).

References

- [1] R. Agrawal, D. Gunopulos, and F. Leymann. Mining process models from workflow logs. In *International Conference on Extending Database Technology*, pages 467–483. Springer, 1998.
- [2] N. Ahmidi, L. Tao, S. Sefati, Y. Gao, C. Lea, B. B. Haro, L. Zappella, S. Khudanpur, R. Vidal, and G. D. Hager. A dataset and benchmarks for segmentation and recognition of gestures in robotic surgery. *IEEE Transactions on Biomedical Engineering*, 64(9):2025–2041, 2017.
- [3] J. D. Birkmeyer, J. F. Finks, A. O’reilly, M. Oerline, A. M. Carlin, A. R. Nunn, J. Dimick, M. Banerjee, and N. J. Birkmeyer. Surgical skill and complication rates after bariatric surgery. *New England Journal of Medicine*, 369(15):1434–1442, 2013.
- [4] T. Blum, N. Padoy, H. Feußner, and N. Navab. Workflow mining for visualization and analysis of surgeries. *International journal of computer assisted radiology and surgery*, 3(5):379–386, 2008.
- [5] C. Cao, C. L. MacKenzie, and S. Payandeh. Task and motion analyses in endoscopic surgery. In *Proceedings ASME Dynamic Systems and Control Division*, pages 583–590. Citeseer, 1996.
- [6] E. Galbrun, P. Cellier, N. Tatti, A. Termier, and B. Crémilleux. Mining periodic patterns with a mdl criterion. *arXiv preprint arXiv:1807.01706*, 2018.
- [7] Y. Gao, S. S. Vedula, C. E. Reiley, N. Ahmidi, B. Varadarajan, H. C. Lin, L. Tao, L. Zappella, B. Béjar, D. D. Yuh, et al. Jhu-isi gesture and skill assessment working set (jigsaws): A surgical activity dataset for human motion modeling. In *MICCAI Workshop: M2CAI*, volume 3, page 3, 2014.
- [8] C. W. Günther and W. M. Van Der Aalst. Fuzzy mining—adaptive process simplification based on multi-perspective metrics. In *International conference on business process management*, pages 328–343. Springer, 2007.
- [9] N. Lavrac and S. Dzeroski. Inductive logic programming. In *WLP*, pages 146–160. Springer, 1994.
- [10] C. Monserrat, J. Hernández-Orallo, J. Dolz, M. Rupérez, and P. Flach. Knowledge acquisition by abduction for skills monitoring: Application to surgical skills. In *26th International Conference on Inductive Logic Programming, ILP2016*, 2016.
- [11] M. Nathan, J. M. Karamichalis, H. Liu, S. Emani, C. Baird, F. Pigula, S. Colan, R. R. Thiagarajan, E. A. Bacha, and P. del Nido. Surgical technical performance scores are predictors of late mortality and unplanned reinterventions in infants after cardiac surgery. *The Journal of thoracic and cardiovascular surgery*, 144(5):1095–1101, 2012.
- [12] D. Nieves, C. Ferri, J. Hernández-Orallo, and C. Monserrat. Low-level event detection system for minimally-invasive surgery training. In *Proceedings of the 4th international Workshop on Sensor-based Activity Recognition and Interaction*, page 5. ACM, 2017.
- [13] S. Stein and S. J. McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pages 729–738. ACM, 2013.
- [14] R. C. Team et al. R: A language and environment for statistical computing. 2013.
- [15] S. S. Vedula and G. D. Hager. Surgical data science: the new knowledge domain. *Innov Surg Sci*, 2(3):109–121, 2017.
- [16] S. S. Vedula, M. Ishii, and G. D. Hager. Objective assessment of surgical technical skill and competency in the operating room. *Annual review of biomedical engineering*, 19:301–325, 2017.
- [17] S. Yang, M. Zhou, S. Chen, X. Dong, O. Ahmed, R. S. Burd, and I. Marsic. Medical workflow modeling using alignment-guided state-splitting hmm. In *Healthcare Informatics (ICHI), 2017 IEEE International Conference on*, pages 144–153. IEEE, 2017.