Repositorios, **memoria dinámica** del desarrollo

Tres expertos extranjeros estuvieron el 30 de agosto del 2017 en la Universidad de los Andes presentado una técnica útil en ingeniería de *software*, poco conocida en el país. Aquí el resumen de sus intervenciones.

Cuando se diseña e implementa una aplicación de *software* siempre quedan rastros del proceso de los desarrolladores, de los requerimientos del cliente, de la creación del código y de lo sucedido hasta llegar a la versión liberada a los usuarios. Si se aprovecha esta información histórica, extrayéndola mediante minería de repositorios de *software*, se ahorra mucho tiempo en la producción de nuevas aplicaciones.

La minería de repositorios de *software* es minería de datos aplicada a tareas de ingeniería de *software*, en la que se emplean técnicas de diferentes áreas y disciplinas de la informática. Se ha estudiado desde hace 15 años, pero en Colombia es poco conocida y en nuestra academia solo la trabajan el profesor Jairo Hernán Aponte, de la Universidad Nacional, y el profesor Mario Linares, del Departamento de Ingeniería de Sistemas y Computación (DISC), en Los Andes.

Para difundir sus usos y posibilidades entre ingenieros de sistemas, en empresas y academia, el profesor Linares organizó el 3.er Foro en Ingeniería de *Software*: "Minería de repositorios al servicio del desarrollo de *software*" y participó con una conferencia sobre el uso de minería de repositorios en el desarrollo de aplicaciones Android (ver pág. 45). Massimiliano Di Penta, de la Universidad de Sannio (Italia), Sonia Haiduc, de la Florida State University, y Christopher Vendome, de College of William and Mary (Estados Unidos), mostraron un panorama general y hablaron de los problemas que esta técnica resuelve.

Fuente: presentación de Massimiliano Di Penta.

Mario Linares explicó a la revista Foros ISIS cómo funciona esta minería y resumió el contenido de las conferencias de los invitados. Dijo que todo el historial del proceso de desarrollo se puede alojar en una suerte de memoria. La información (es decir, código, documentos, texto, archivos) se registra de manera explícita en los repositorios, si se siguen buenas prácticas. Pero como no siempre es así, queda embebida o implícita en los artefactos y es factible obtenerla con minería de repositorios de *software*.

Consideraciones de uso

Massimiliano Di Penta mostró una perspectiva general sobre cómo minar repositorios. Habló de los beneficios de las técnicas disponibles diseñadas para llevar a cabo tareas particulares, qué extraen y qué se utiliza. Por ejemplo, si se quiere encontrar al mejor desarrollador de la empresa para detectar un error en una aplicación, un algoritmo de minería asigna al experto que en otras ocasiones ha tenido éxito en esa labor.

También se pueden predecir defectos en un *software* en proceso: es posible determinar el porcentaje de probabilidades de fallas de una porción del código o detectar errores por acciones del desarrollador. En este caso, el algoritmo de minería, es capaz de revelar los problemas que se presentaron en el pasado por una acción concreta y genera los reportes del caso.

Sonia Haiduc, de la Florida State University, Mario Linares, del Departamento de Ingeniería de Sistemas y Computación, Massimiliano Di Penta, de la Universidad de Sannio (Italia), y Christopher Vendome, de College of William and Mary (Estados Unidos).

El profesor Di Penta recomendó trabajar con cuidado para no tomar decisiones equivocadas: hay que ser precavido en el análisis de las dificultades, en la escogencia del algoritmo, en su uso bajo un dominio y con unos datos en particular y en el estudio de los resultados. Por ello, es necesario confrontar el proceso automático para verificar la precisión de los resultados generados.

Cuándo usarla

Mario Linares explicó que hay repositorios para diferentes actividades. Con los históricos se es posible analizar la evolución de los artefactos; los hay que rastrean problemas (*issue tracker*), registran *bugs* (errores) o solicitudes de cambio. Hay otros que guardan el código fuente, de donde se puede sacar el lenguaje de las aplicaciones y la información de usuario de un programa.

Para decidirse a emplear las técnicas o herramientas de minería es importante constatar que, efectivamente, con un algoritmo, los datos resultantes sí ayudarán a identificar una necesidad o una tarea que se deba implementar; también verificar que la información sí se puede extraer y que el volumen de datos lo amerita, pues no tiene sentido minar diez documentos. Además, es necesario determinar el retorno de la inversión e identificar si la compañía tiene los recursos y el tiempo para entrenar un equipo, implementar el proceso, continuarlo y mantenerlo.

Recuperación de texto

A veces se debe buscar en repositorios históricos de comunicaciones, pues la información sobre las decisiones de diseño se encuentra en las conversaciones entre los desarrolladores, en sus correos o chats. Estos datos pueden ser útiles para redocumentación, modernización de sistemas existentes o en la creación de nuevos productos de *software*.

Cuando se quiere revisar ese historial, se emplea la minería de repositorios con técnicas generadas en otras disciplinas, conocidas como procesamiento de lenguaje natural y recuperación de la información. A estas se refirió Sonia Haiduc en su conferencia "El uso de recuperación de texto y procesamiento de lenguaje natural en ingeniería de *software*", pues no solo los documentos del desarrollo son texto: como el código es un subconjunto del lenguaje natural se puede tratar como texto. El propósito es emplear esos insumos de forma automática en las tareas del desarrollo de *software*.

La profesora Haiduc mostró los modelos más utilizados: Vector Space Model (VSM), Latent Semantic Indexing (LSI), Latent Dirichlet Allocation (LDA), Okapi BM25 and BM25F, Language Models.

Atención a las licencias libres

Las herramientas disponibles para minería de repositorios de *software* también se utilizan para evitar problemas de propiedad intelectual cuando se usa código *open source*. A pesar de que ciertos programas y librerías son de acceso libre, su uso, distribución y modificación están protegidos por una gran variedad de licencias que pueden ser de dos tipos: restrictivas y permisivas. Conocer y respetar sus especificaciones en el momento de descargarlos y aprovecharlos evita demandas. Este fue el tema de la conferencia de Christopher Vendome, experto en licenciamiento y más concretamente en el del uso de técnicas de minería de repositorios aplicadas a prevenir inconvenientes con los permisos de programas gratuitos.

Los algoritmos diseñados para tal fin ubican y caracterizan las licencias del *software open source* con que se planea trabajar, y establecen si estas son compatibles con las de otras librerías o si alguna de las de un código seleccionado viola las exigencias de otra. ■

Beneficios: Reducción de los tiempos de trabajo. Algo que tomaba 10 horas, se puede demorar 15 minutos, cuando el algoritmo ya está implementado. El provecho económico depende de cada contexto, pero también lo hay.

En el mercado: Hay herramientas comerciales y *open source*, *frameworks* y librerías en diferentes lenguajes de programación, con *suites* y algoritmos implementados. Algunos de los libres son: Weka, de la Universidad de Waikato (Nueva Zelandia); Rapid Minner, SciPy y Panda. Repositorios como Github y Stack Overflow también proveen APIs para acceso a los datos alojados en el repositorio. Estas herramientas en general son para minería de datos, pero pueden ser personalizadas para construir sistemas de minería de repositorios de *software*.