

HIPÓTESIS BAYESIANA EN MODELOS DE COMPLETITUD

Marcela Ridaó ⁽¹⁾ Jorge H. Doorn ^{(1) (2)}

(1) INTIA, Universidad Nacional del Centro de la Provincia de Buenos Aires, Argentina

(2) Universidad Nacional de La Matanza, Argentina

e-mail: {mridao, jdoorn}@exa.unicen.edu.ar

ABSTRACT

La Ingeniería de Requisitos (IR) provee métodos, técnicas y herramientas para ayudar a los ingenieros a elicitar y especificar requisitos, asegurando el máximo de calidad y completitud. Sin embargo, el problema de la completitud es una amenaza constante a la calidad de los requisitos. La completitud es una meta inalcanzable y estimar el grado de completitud logrado en cualquier momento del proyecto es muy difícil. Esta situación no es única en el proceso global de desarrollo de software. Ocurre algo muy similar, por ejemplo, en las pruebas o inspecciones de software. Y este problema se produce también en otras áreas del conocimiento. En este artículo se presenta un proyecto que pretende analizar datos de captura y recaptura en el proceso de Ingeniería de Requisitos utilizando un nuevo punto de vista. Se considerará como posible el acoplamiento entre los diferentes factores que inciden en la probabilidad de captura.

INTRODUCCIÓN

El objetivo de la Ingeniería de Requisitos es sistematizar el proceso de definición de requisitos [1] [2] junto con la creación de un compromiso entre los clientes-usuarios y los desarrolladores, ya que ambos deben participar en dicho proceso [3]. La validación de los requisitos se ha convertido en una tarea compleja, principalmente debido al tipo de modelos usados para su representación, que requieren en muchos casos clientes-usuarios con habilidades especiales para entenderlos. El uso de representaciones basadas en lenguaje natural ayuda a la validación de los requisitos, que mejora notablemente cuando las representaciones son expresadas usando el vocabulario de los clientes-usuarios [4]. Una de las dificultades de las representaciones basadas en lenguaje natural es la ambigüedad [5] [6], un inconveniente que podría reducirse parcialmente con la elaboración de un glosario de la aplicación. Varias experiencias han mostrado que un glosario del vocabulario de los clientes-usuarios es, en sí mismo, una fuente de información para elicitar información del Universo de Discurso (UdeD) [7] [8] [9] [10].

El uso de los modelos del Léxico Extendido del Lenguaje (LEL) y de los Escenarios para la elicitación de requisitos y su utilización a través de todo el proceso de desarrollo de software facilita la validación con el cliente/usuario [11]. El propósito de estos modelos consiste en primero elicitar conocimiento del UdeD, y luego el conjunto de los requisitos del sistema de software a ser desarrollado. Las palabras o frases más relevantes o peculiares del UdeD son incluidas en el LEL. Los escenarios son usados para entender el UdeD primero, y luego para entender el problema y su funcionalidad. Cada escenario describe una situación específica del UdeD enfocándose en su comportamiento.

Estos modelos son sometidos a la verificación de su consistencia interna y luego validados con la colaboración de los clientes-usuarios. Durante la verificación y validación, la completitud es un aspecto fundamental. Tanto las inspecciones de LEL y Escenarios [12] como la Validación consideran la completitud como uno de sus objetivos principales. Sin embargo, esto está lejos de ser suficiente en cuanto a completitud se refiere.

El problema de la completitud en la Ingeniería de Software en general y en la Ingeniería de Requisitos en particular, es similar a problemas que se producen en otras áreas del conocimiento. Otis [13] introdujo un método para estimar el tamaño de una población cerrada de animales basándose en los datos de captura y recaptura de especímenes. Este método ha sido utilizado exitosamente en el área de inspecciones de software por varios autores [14] [15] [16] [17] [18] [19].

En este trabajo, se propone mejorar las estimaciones ya realizadas [20][21] del tamaño máximo de los modelos de LEL y Escenarios, mediante la aplicación de un método cuantitativo basado en las técnicas de captura/recaptura ya aplicadas en la estimación del número total de defectos en inspecciones [14] [15] [22].

ESTIMACIÓN DEL TAMAÑO DE POBLACIONES CERRADAS

Los métodos de estimación de poblaciones cerradas de cualquier naturaleza, animales, errores en el software y en este caso términos de LEL o escenarios, se basan en la idea de capturar más de una vez especímenes de una población cuyo tamaño se desconoce. Inicialmente, este método fue aplicado a poblaciones de animales salvajes [13]. La estrategia involucra la captura y liberación de especímenes en varias sesiones de captura. Si los mismos especímenes son capturados una y otra vez, puede concluirse que la población es básicamente el conjunto capturado. Cuando muy pocos especímenes son recapturados, en cambio, se puede concluir que la población es muy grande.

Los modelos estadísticos iniciales de captura y recaptura asumían que cada animal salvaje tenía la misma probabilidad de ser atrapado y que cada procedimiento de captura tenía la misma probabilidad de capturar especímenes. Luego, se introdujeron nuevos modelos considerando diferencias entre los animales y otros factores cualitativos. El modelo original es actualmente identificado como M_0 . Los modelos M_t , M_b y M_h introducen diferentes fuentes de variación en las probabilidades. El modelo M_t considera que las probabilidades de captura pueden variar en las diferentes cacerías debido al clima, la ubicación de las trampas, o aún el procedimiento de captura. El modelo M_b considera que las probabilidades de captura pueden variar por cambios en el comportamiento causados por capturas previas. Finalmente, M_h considera que las probabilidades de captura pueden variar debido a la heterogeneidad entre los animales individuales.

La combinación de más de una fuente de variación de probabilidad de captura dio lugar a nuevos modelos, como M_{tb} , M_{th} , M_{bh} y M_{tbb} . Cada uno de estos modelos considera la independencia entre los diferentes factores cualitativos. En otras palabras, la probabilidad combinada es determinada por la regla del producto del teorema de Bayes [23].

RESULTADOS ALCANZADOS

En experimentos previos, nueve grupos, utilizando como técnica de elicitación la lectura de documentos, procedieron a la confección del Léxico Extendido del Lenguaje. Posteriormente, a partir de este modelo, cada uno de los grupos construyó un modelo de escenarios.

El caso de estudio utilizado fue un sistema de administración de planes de ahorro para la adquisición de vehículos OKm [24] [25]. Funciona a través de grupos de personas que pagan cuotas mensuales para adquirir un automóvil. Estas personas participan en reuniones mensuales en las cuales una unidad es adjudicada por sorteo o licitación a uno de los participantes. Además, el sistema incluye el arbitraje en los casos de renuncia o muerte de los participantes, falta de pago de las cuotas mensuales, seguros y contratos con los fabricantes, entre otros.

Los datos obtenidos fueron analizados empleando el método propuesto por Wohlin y Runeson [22]. El fundamento de este método reside en que un ajuste de los datos experimentales con una curva teórica simple como puede ser una exponencial decreciente empotra dentro de los parámetros de la curva las variaciones de probabilidades del modelo M_{th} sin necesidad de suponer ninguna distribución de probabilidades en particular.

El análisis de los datos de ambos experimentos confirmó casi totalmente lo predicho por [22] en el caso de inspecciones, donde se supone que al aumentar el número de inspectores, la cantidad de defectos estimada será igual a la cantidad real existente. En este sentido, se considera más apropiado enunciar que cuando el número de observaciones independientes crece, es razonable suponer que la diferencia entre la cantidad estimada y la cantidad efectivamente obtenida se reduce [20][21]. Se observó también que al aumentar el número de captores aumenta la precisión obtenida al ajustar los datos con una curva, tanto en el caso de la estimación del número de símbolos del LEL, como para escenarios.

En las experiencias realizadas se puede ver que el factor t (la diferencia entre los ingenieros de requisitos) influye de un modo muy similar a cómo lo hace en las inspecciones de software [20] [21]. En ambos casos, tiene una incidencia mayor que en el caso de animales salvajes. En la figura 1a, puede verse el número de símbolos del LEL obtenidos en un experimento en el cual nueve grupos, utilizando la misma técnica de elicitación, procedieron a la confección del Léxico Extendido del Lenguaje para un mismo UdeD. En la figura 1b, puede verse el número de escenarios obtenidos por los mismos grupos a partir de los Léxicos respectivos.

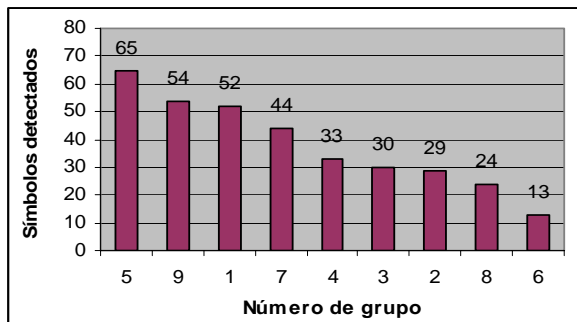


Figura 1a. Número de símbolos del LEL elicitados por diferentes ingenieros de requisitos.

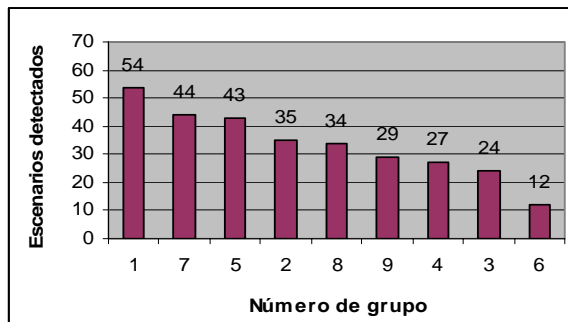


Figura 1b. Número de escenarios obtenidos por diferentes ingenieros de requisitos.

Sin embargo, un hecho notoriamente más importante se hace evidente: las probabilidades de los ingenieros de requisitos y las probabilidades de los símbolos / escenarios están lejos de ser factores independientes. Esto no fue reportado durante las experiencias con Captura y Recaptura en inspecciones de software. En otras palabras, un modelo M_{th} no puede ser aplicado basándose en la idea de independencia entre los factores. Este fenómeno puede ser fácilmente entendido teniendo en cuenta que diferentes personas tienen diferentes habilidades. En este caso puede verse, por ejemplo, que un grupo que fue muy bueno en la detección de símbolos, no lo fue tanto para la elaboración de escenarios. Entonces, parece ser que no existe algo como la probabilidad de un símbolo o escenario de ser detectado por un ingeniero de requisitos ni la probabilidad de un ingeniero de requisitos de detectar símbolos o escenarios. Lo que realmente existe es la probabilidad de un cierto ingeniero de requisitos de detectar un símbolo o un escenario determinado.

RESULTADOS ESPERADOS

La propuesta de este trabajo es considerar la combinación de las tres hipótesis: probabilidad de los elementos de ser detectados, habilidad de los captores y acostumbramiento, para estimar el número de elementos de una población cerrada. Este estudio se realizará en el marco de la estrategia de obtención de Requisitos antes presentada. Por lo tanto, se analizará la completitud de los modelos de LEL y Escenarios.

En la aplicación de los métodos de captura y recaptura a inspecciones de software [14] [15] [16] [17] [18] [19] los errores asumieron el rol de los animales salvajes y los inspectores, el rol de los cazadores. Es evidente que la variación de las probabilidades de captura debido a cambios en el comportamiento de los errores causados por más de una inspección es nula. Entonces, los modelos M_b , M_{bh} , M_{tb} y M_{tbh} deberían ser descartados, usando en cambio los modelos M_o , M_h , M_t y M_{th} . Esto es tan obvio que los autores antes mencionados [14] [15] [16] no mencionaron el factor comportamiento en la variación de la probabilidad.

Sin embargo, cuando las técnicas de captura y recaptura son aplicadas en el dominio de IR, el factor comportamiento debería ser analizado puesto que no está claro si debería ser descartado o no. Realmente, esto depende de la fuente de información utilizada durante la elicitación de conocimiento y de la técnica aplicada. Por ejemplo, si más de un ingeniero de requisitos lee el mismo documento para detectar símbolos del LEL, no hay forma de que los símbolos se vean influenciados por las lecturas previas. En cambio, si varios ingenieros de requisitos entrevistan al mismo cliente-usuario durante la actividad de creación del LEL, es posible que la persona

entrevistada (que contiene en su forma de hablar la información acerca de los símbolos del LEL) pueda resultar influenciada por entrevistas previas. Esto implica que el entrevistado tiende a cambiar después de la entrevista inicial. Puede sentirse incómodo por ser entrevistado una y otra vez sobre el mismo tema, o bien puede sentirse más involucrado con el proyecto de software, proveyendo entonces más y más información útil en entrevistas posteriores.

Entonces, se propone analizar nuevamente los datos obtenidos en los experimentos descritos en la sección anterior, tratando de obtener un modelo que represente mejor los datos experimentales, poniendo énfasis en el acoplamiento de los factores t (diferente probabilidad de captura para los ingenieros de requisitos) y h (diferente probabilidad de los símbolos o escenarios de ser capturados). Para este primer caso de estudio, no se considerará el factor b (diferente probabilidad de captura debido al acostumbramiento), ya que la técnica de elicitación utilizada ha sido la lectura de documentos, en cuyo caso este factor carece de sentido.

En la figura 2, se representa el problema en el contexto del modelo del LEL del caso de estudio.

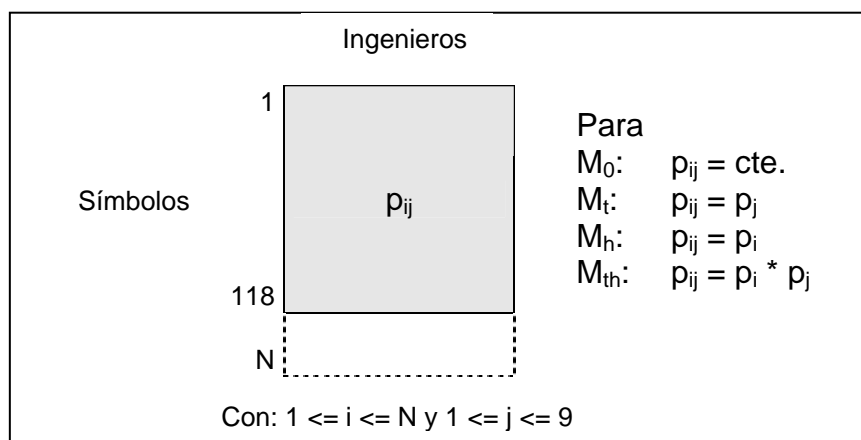


Figura 2. Probabilidad de captura según el modelo

El subíndice j se utiliza para representar los 9 grupos de ingenieros de requisitos y el subíndice i, para el número de símbolos. El área gris representa los símbolos detectados por los 9 grupos, mientras que el área punteada, indica los símbolos que aún no han sido detectados.

Los modelos utilizados en inspecciones de software, asumen diferentes posturas en relación con la probabilidad de captura p_{ij} . M_0 considera esta probabilidad constante; M_t sólo considera variación en la probabilidad debido a diferentes habilidades de los captores, con lo cual p_{ij} será igual a p_j ; M_h sólo considera variación en la probabilidad debido a la heterogeneidad de los símbolos, con lo que p_{ij} será igual a p_i ; y por último, M_{th} considera la variación de los dos factores, pero de forma independiente, con lo cual p_{ij} es el producto de p_i y p_j .

Sin embargo, ninguno de estos modelos ha podido representar los datos experimentales adecuadamente. Parece ser que no existe la probabilidad de un símbolo de ser detectado (p_i) ni la probabilidad de un ingeniero de requisitos de detectar símbolos (p_j), sino la probabilidad de cierto ingeniero de requisitos de detectar un símbolo determinado. Por lo tanto, para el cálculo de p_{ij} se considerará el acoplamiento de los factores t y h.

En futuros proyectos, se pretende extender el análisis a casos de estudio en los cuales tenga sentido considerar el factor b acoplado a los otros dos factores. Es decir, se aplicará una técnica de elicitación en la cual el comportamiento de la fuente de información pueda verse afectado por los procedimientos de captura, como podría suceder con el método de entrevistas.

CONCLUSIONES

La validación de los requisitos del software es una tarea difícil; no tanto sobre los requisitos efectivamente entendidos y modelados, como sobre aquellos que permanecen ocultos. Estos aparecerán en el medio del proceso de desarrollo de software con un notorio poder de disturbio, o

serán descubiertos cuando el software sea puesto en servicio. La funcionalidad del software no será lo que los clientes-usuarios deseaban.

La estrategia de captura y recaptura ayuda a reducir la cantidad de requisitos ocultos dando a los involucrados una idea acerca de cuántos requisitos permanecen sin modelar y tal vez ayuden en el desarrollo de mejores heurísticas para todo el proceso de Ingeniería de Requisitos.

Se ha planeado repetir el análisis de los datos disponibles de captura y recaptura en el proceso de Ingeniería de Requisitos, considerando el acoplamiento entre los diferentes factores cualitativos que pueden variar la probabilidad de captura.

REFERENCIAS

- [1] Kotonya, G., Sommerville, I. (1998). Requirements Engineering. Processes and Techniques. John Wiley & Sons.
- [2] Sommerville, I., Sawyer, P. (1997). Requirements Engineering. A good Practice Guide. John Wiley & Sons.
- [3] Maculay, L. (1993). Requirements Capture as a Cooperative Activity. Proceedings of IEEE International Symposium on Requirement Engineering. IEEE Computer Society Press, San Diego, Ca. 174-181.
- [4] Leite, J.C.S.P., Franco, A.P.M.(1990). O Uso de Hipertexto na Elicitação de Linguagens da Aplicação, Anais de IV Simpósio Brasileiro de Engenharia de Software, SBC, Brazil. 134-149.
- [5] Jackson, M. (1995). Software Requirements & Specifications. A lexicon of practice, principles and prejudices. Addison Wesley, ACM Press.
- [6] Berry, D.M., Kamsties, E. (2004) Ambiguity in Requirements Specification. In Leite, J.S.C.P., Doorn, J.H (Eds.) Perspectives on Software Requirements. Kluwer Academic Press. 7-44.
- [7] Ben Achour, C., Rolland, C., Maiden, N.A.M., Souveyet, C. (1999). Guiding Use Case Authoring: Results of an Empirical Study. Proceedings International Symposium On Requirements Engineering. IEEE Computer Society Press, Limerick-Ireland. 36-43.
- [8] Rolland, C., Ben Achour, C. (1998). Guiding the construction of textual use case specifications. Data & Knowledge Engineering, 25. 125-160.
- [9] Oberg, R., Probasco, L, Ericsson, M. (1998). Applying Requirements Management with Use Cases. Rational Software Corporation.
- [10] Regnell, B. (1999). Requirements Engineering with Use Cases – a Basis for Software Development, Doctoral Thesis, Department of Communication Systems, Lund University.
- [11] Leite, J.C.S.P., Hadad, G.D.S., Doorn, J.H., Kaplan, G.N. (2000). A Scenario Construction Process. Requirements Engineering Journal, 5, (1). 38-61.
- [12] Leite, J.C.S.P., Hadad, G.D.S., Doorn, J.H., Kaplan, G.N. (2005). Scenario Inspections. Requirement Engineering Journal, 10, (1). 1, 21.
- [13] Otis, D.L., Burnham, K.P. White G.C., Anderson D.R. (1978). Statistical inference from Capture on Closed Animal Populations. Wildlife Monograph, 62.
- [14] Briand, L.: El Emam, K., Freimut, B., Laitenberger, O. (2000). A Comprehensive Evaluation of Capture-Recapture Models for Estimating software Defects Contents. IEEE Transactions on Software Engineering, 26, (6). 518-540.
- [15] Biffi, S. (2000). Using Inspection data for Defect Estimation. IEEE Software special Issue on recent project estimation methods.
- [16] Biffi, S. (2003). Evaluating defect estimation models with major defects. The Journal of Systems and Software, 65. 13-29.
- [17] Kamel, A., Sorenson, P. (2003). The Application of Capture-Recapture Log-Linear Models to Software Inspections Data. Proceedings of ISESE'03 - International Symposium on Empirical Software Engineering, 213-222.
- [18] Thelin, T. (2004). Team-based Fault Content Estimation in the Software Inspection Process. Proceedings of ICSE'04 - 26th International Conference on Software Engineering, 263-272.
- [19] Petersson, H., Thelin, T., Runeson, P., Wohlin, C. (2003). Capture–recapture in software inspections after 10 years research—theory, evaluation and application. The Journal of Systems and Software, 72, 249–264.
- [20] Doorn, J., Ridao, M. (2003) Completitud de Glosarios: Un Estudio Experimental. Anais do WER'03, Workshop en Engenharia do Requisitos, Paracicaba-SP, Brasil, 317-328.
- [21] Ridao, M., Doorn, J. (2006). Estimación de Completitud en Modelos de Requisitos Basados en Lenguaje Natural. Anais do WER'06, Workshop en Engenharia do Requisitos, Rio de Janeiro, Brasil, 151-158.
- [22] Wohlin, C., Runeson, P. (1998). Defect content estimations from Review Data. Proceedings of the 20th International Conference on Software Engineering. 400-409.
- [23] Papoulis, A. (1985). Probability, Random Variables, and Stochastic Processes. Mc Graw Hill.
- [24] Mauco, V., Ridao M., del Fresno, M., Rivero, L., Doorn, J.H. (1997). Ingeniería de Requisitos, Proyecto: Sistema de Planes de Ahorro. Reporte técnico, ISISTAN, UNCPBA, Tandil, Argentina.
- [25] Rivero, L., Doorn, J., del Fresno, M., Mauco, V., Ridao, M., Leonardi, M.C.(1998). Una Estrategia de Análisis Orientada a Objetos basada en Escenarios: Aplicación en un Caso Real. Anais do WER'98, Workshop en Engenharia do Requisitos, Maringá, Brasil. 79-90.