Análisis de la Calidad de Datos en Experimentos en Ingeniería de Software

Carolina Valverde, Adriana Marotta y Diego Vallespir

Facultad de Ingeniería, Universidad de la República
{mvalverde,amarotta,dvallesp}@fing.edu.uy

Resumen Este trabajo presenta un enfoque sistemático, disciplinado y estructurado para identificar y medir los posibles problemas de calidad en los datos recolectados en experimentos en Ingeniería de Software, mediante la aplicación de la disciplina Calidad de Datos. También se presenta la aplicación de este enfoque en un experimento realizado en la Universidad de la República en Uruguay. Los resultados del análisis de la calidad indican claramente que los investigadores que realizan experimentos en ingeniería de software deben analizar la calidad de los datos del experimento antes de realizar los análisis estadísticos del mismo. La aplicación del enfoque mostró ser útil y presenta diferencias con propuestas anteriores para el estudio de la calidad de datos en experimentos controlados en ingeniería de software.

1. Introducción

La Ingeniería de Software Empírica busca, a partir de la experimentación, conocer si ciertos supuestos sobre el desarrollo de software son reales. Durante el proceso de experimentación se genera gran cantidad de datos. Sobre estos se realizan análisis estadísticos y estudios comparativos. Por último, se establecen resultados y conclusiones que surgen del propio análisis realizado [8].

Uno de los experimentos que sentó las bases de cómo experimentar en ingeniería de software, fue el realizado por Basili y Selbi [2]. Desde hace pocos años la comunidad de ingeniería de software empírica ha investigado distintos aspectos relativos a los experimentos controlados. Algunos ejemplos son: la forma de reportar experimentos [7], la forma de diseñarlos y conducirlos [8,17], y cómo realizar meta-análisis [13].

Los datos recolectados durante la ejecución de un experimento pueden ser de mala calidad. Esto provoca que los resultados del mismo sean cuestionables. Incluso, si la calidad de los datos está comprometida en un alto porcentaje de los datos, o con un problema "grande" en un pequeño porcentaje de los mismos, los resultados del experimento pueden ser incorrectos [1].

Sin embargo, en la literatura poco se menciona acerca de la importancia de la calidad de los datos recolectados en un experimento y menos aún en lo que refiere particularmente a experimentación en ingeniería de software [1,10,11]. Además, en los casos que se realiza un análisis de la calidad de los datos de un experimento, tanto como forma de evaluar la misma como para realizar una limpieza, este

se basa casi exclusivamente en la detección de *outliers* [11]. Como resultado de un extenso estudio bibliográfico, Bachman menciona que solo unas pocas publicaciones plantean el tema de calidad de datos, y que ningún estudio ha verificado la completitud o el grado de veracidad en los datos utilizados, y menos aún explorado los posibles efectos sobre los resultados de los experimentos [1]. Liebchen y Shepperd establecen que los investigadores en ingeniería de software empírica utilizan datos en los cuales basan sus investigaciones pero que, en la gran mayoría de los casos, no se estima ni analiza el efecto que podría causar en los resultados si los mismos fueran de mala calidad [11].

Por otro lado, se debe tener en cuenta que la Calidad de Datos es un área de investigación en sí misma, en la cual se ha generado un gran volumen de trabajo (sobre todo en los últimos años) enfocado principalmente a: definir los distintos aspectos de la calidad de los datos [3,9,12,14], y proponer técnicas, métodos y metodologías para la medición y para el tratamiento de la calidad de los datos [3,9,16].

En este artículo presentamos un análisis de la calidad de los datos recolectados por los sujetos que participaron de un experimento controlado en ingeniería de software. Adoptamos un enfoque sistemático, disciplinado y estructurado para estudiar los datos de dicho experimento de manera de encontrar problemas en la calidad de los mismos. Para ello utilizamos la propuesta de Batini y Scannapieco, que proviene de la disciplina Calidad de Datos [3]. Nuestro trabajo utiliza e instancia dicha propuesta para que pueda ser aplicada en los datos del experimento; pudiendo resultar repetible en estudios similares.

La Figura 1 presenta el trabajo completo que fue realizado respecto a la calidad de los datos. Luego de ejecutado el experimento controlado se identificaron los posibles problemas de calidad sobre los datos. Para cada problema, se identificaron y ejecutaron métricas específicas cuyos resultados fueron registrados. Luego se realizó la limpieza y migración de los datos. Este artículo presenta los problemas de calidad, las métricas definidas y la medición realizada en la base de datos del experimento, dejando fuera las actividades de limpieza de datos.

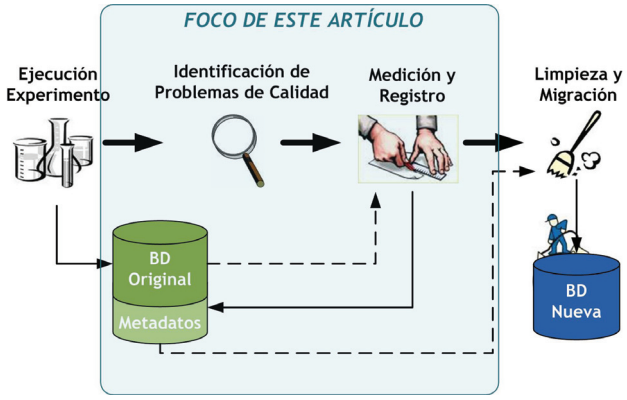

Figura 1. Etapas del estudio realizado.

Este artículo está organizado de la siguiente manera. La sección 2 presenta las dimensiones y factores de la calidad de datos que son utilizados en este trabajo. El experimento controlado se presenta en la sección 3. La sección 4 presenta los problemas de calidad detectados. Los resultados se presentan en la sección 5 y las conclusiones en la sección 6.

2. Calidad de Datos

Los datos constituyen un recurso muy valioso para las organizaciones al ser utilizados principalmente para la toma de decisiones. La mala calidad de los mismos influye de manera significativa y profunda en la efectividad y eficiencia de las organizaciones así como en todo el negocio [3]. Cada día se hace más notoria la importancia y necesidad en distintos contextos de un nivel de calidad adecuado para los datos.

Existen distintos aspectos que componen la calidad de datos. Estos se conocen como dimensiones de calidad. En los trabajos del área de Calidad de Datos existe un núcleo de dimensiones que es compartido por la mayoría de las propuestas [9,12,14,16]. Nuestro trabajo se basa en la propuesta de Batini y Scannapieco [3], que consensua las distintas dimensiones que se han propuesto.

En este trabajo utilizamos una abstracción de la calidad de datos [6], donde además de las dimensiones se definen otros conceptos para la clasificación y el manejo de la misma. Estos conceptos son el de factor, métrica y método de medición. Una dimensión de calidad captura una faceta (a alto nivel) de la calidad de los datos. Por otra parte, un factor de calidad representa un aspecto particular de una dimensión de calidad. Una dimensión puede ser entendida como un agrupamiento de factores que tienen el mismo propósito de calidad. Una métrica es un instrumento que define la forma de medir un factor de calidad. Un mismo factor de calidad puede medirse con diferentes métricas. A su vez, un método de medición es un proceso que implementa una métrica. Se pueden utilizar distintos métodos de medición para una misma métrica.

Normalmente, los datos se encuentran en algún repositorio de datos. En nuestro caso los datos se encuentran en una base de datos relacional y por eso es de especial interés considerar las mediciones de la calidad de los datos en ese tipo de repositorio.

Las mediciones en una base de datos relacional se pueden realizar a varios niveles: celda, tupla, tabla, e incluso a nivel de la base de datos entera. En definitiva, se pueden considerar distintos niveles de granularidad para evaluar la calidad de los datos. Por esto se definen funciones de agregación, las cuales calculan un valor de calidad para un conjunto de datos a partir de valores de calidad medidos para cada elemento de ese conjunto, es decir permiten pasar de un nivel de granularidad de datos a otro, obteniendo la calidad resumida para ese nuevo nivel. Por ejemplo, es posible obtener una medida de calidad de una tupla a partir de las medidas de calidad de cada una de sus celdas.

A continuación se presentan las dimensiones y factores de calidad utilizadas en este trabajo. De la propuesta de Batini y Scannapieco [3], no se considera la

dimensión frescura relacionada con el tiempo y la vigencia de los datos, ya que entendemos no tiene aplicabilidad en nuestro caso. Esto se debe a que los datos del experimento se consideran "frescos" (son eternamente vigentes).

Dimensión: Exactitud

La exactitud indica que tan precisos, válidos y libres de problemas están los datos. Establece si existe una correcta y precisa asociación entre los estados del sistema de información y los objetos del mundo real.

Existen tres factores de exactitud: exactitud semántica, exactitud sintáctica y precisión. La exactitud semántica se refiere a la cercanía que existe entre un valor v y un valor real v'. Interesa medir que tan bien se encuentran representados los estados del mundo real en el sistema de información.

La exactitud sintáctica se refiere a la cercanía entre un valor v y los elementos de un dominio D. Interesa saber si v corresponde a algún valor válido de D, sin importar si ese valor corresponde a uno del mundo real.

La precisión, por otra parte, se refiere al nivel de detalle de los datos.

Dimensión: Completitud

La completitud indica si el sistema de información contiene todos los datos de interés, y si los mismos cuentan con el alcance y profundidad que sea requerido. Establece la capacidad del sistema de información de representar todos los estados significativos de una realidad dada.

Existen dos factores de la completitud: cobertura y densidad. La cobertura se refiere a la porción de datos de la realidad que se encuentran contenidos en el sistema de información. La densidad se refiere a la cantidad de información contenida y faltante acerca de las entidades del sistema de información.

En un modelo relacional la completitud (en particular, la densidad) se caracteriza principalmente por los valores nulos, cuyo significado a pesar de ser variado, es importante conocer. Un nulo puede indicar que dicho valor no existe, que existe pero no se conoce, o que no se sabe si existe en el mundo real.

Dimensión: Consistencia

Esta dimensión hace referencia al cumplimiento de las reglas semánticas que son definidas sobre los datos. La inconsistencia de los datos se hace presente cuando existe más de un estado del sistema de información asociado al mismo objeto de la realidad, y hay contradicciones entre dichos estados.

Las restricciones de integridad, por otra parte, definen propiedades que deben cumplirse por todas las instancias de un esquema relacional. Se distinguen tres tipos de restricciones de integridad, las cuales se corresponden con los factores de esta dimensión.

Las restricciones de dominio, se refieren a la satisfacción de reglas sobre el contenido de los atributos de una relación.

Las restricciones intra-relación, se refieren a la satisfacción de reglas sobre uno o varios atributos de una relación. Las restricciones inter-relación, se refieren a la satisfacción de reglas sobre atributos de distintas relaciones.

Dimensión: Unicidad

La unicidad indica el nivel de duplicación de los datos. La duplicación ocurre cuando un objeto del mundo real se encuentra representado más de una vez

en los datos, esto es, varias tuplas representan exactamente el mismo objeto. Distinguimos entonces dos factores de la dimensión Unicidad: Duplicación (la misma entidad aparece repetida de manera exacta) y Contradicción (la misma entidad aparece repetida con contradicciones).

3. Experimento para Evaluar Técnicas de Pruebas

Un tipo de experimento controlado usado habitualmente en ingeniería de software es el experimento con sujetos humanos [8]. En estos experimentos gran parte de los datos que se recolectan durante el mismo es generada por humanos. Por ende, la calidad de estos datos siempre debe estar en duda.

Durante 2008 y 2009 en la Facultad de Ingeniería, Universidad de la República en Uruguay realizamos un experimento controlado para evaluar la efectividad y costo de distintas técnicas de pruebas unitarias [15]. Los sujetos que participaron del experimento debían probar distintos programas con distintas técnicas de pruebas en busca de defectos. Un defecto es una anomalía en el código fuente. Cuando un sujeto encontraba un defecto debía registrar los datos del mismo. Los sujetos también registraban la clasificación de cada defecto según dos taxonomías: ODC [5] y una propuesta por Beizer [4]. Para cada uno de los defectos los sujetos debían registrar, entre otros, los siguientes datos: nombre de archivo y número de línea de código donde se encuentra el defecto, clasificación del defecto en ODC y Beizer, tiempo que le llevó detectar el defecto y descripción del defecto.

Los sujetos no solamente registraban defectos. En una herramienta web para registro de defectos y tiempos los sujetos registraban los siguientes datos: fecha y hora de comienzo y finalización, tiempo de diseño de casos de prueba y de ejecución de la experiencia, y los datos mencionados sobre los defectos encontrados.

La herramienta web fue construida a medida para la recolección de datos del experimento. Esta utiliza una base de datos centralizada donde se guardan todos los datos que registran los sujetos durante el experimento. En este artículo presentamos el análisis de la calidad de los datos que están almacenados en dicha base.

4. Problemas de Calidad en los Datos de un Experimento

En esta sección se presentan los problemas de calidad que identificamos. Llamamos *problemas de calidad* a los tipos de problemas dentro de un factor y dimensión de calidad. Los problemas son específicos para el experimento que analizamos pero generalizables a otros experimentos similares.

El primer paso hacia la identificación de los problemas de calidad, fue definir en conjunto con el equipo que condujo el experimento controlado (consumidores de los datos) cuáles eran los datos que tenían mayor impacto en los análisis estadísticos. Nuestro foco estuvo entonces en estudiar la calidad de esos datos. A continuación, se procedió a analizar las dimensiones y factores de calidad propuestos por Batini y Scannapieco [3] aplicándolos a dicho conjunto de datos.

Otra actividad importante realizada para identificar los problemas fue la exploración de la herramienta web de registro de tiempos y defectos y un análisis de la estructura de la base de datos que contiene los datos que registraron los sujetos. De esta manera, fue posible identificar qué problemas de calidad podrían presentarse en la base bajo estudio, y qué métricas era necesario definir para poder detectar su presencia o ausencia.

La Figura 2 muestra la relación que existe entre dimensiones, factores, problemas y métricas, y cómo se aplican estos conceptos en un caso particular. Mientras que la definición de dimensión y factor de calidad son generales, los problemas y métricas son específicos y definidos para nuestro experimento, aunque pueden ser utilizadas en experimentos similares. Es por ello que los problemas de calidad aquí presentados podrían ser reutilizados en otros experimentos controlados similares.

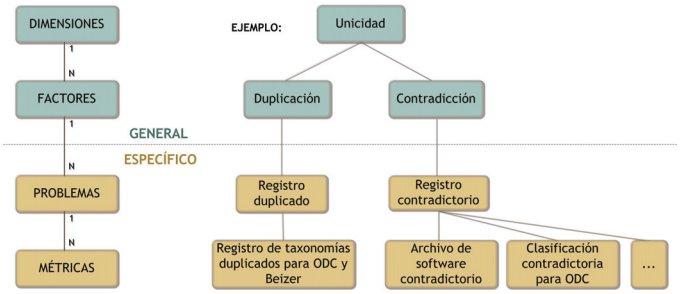

Figura 2. Relación entre Dimensiones, Factores, Problemas y Métricas.

Como se mencionó anteriormente, las dimensiones de calidad de datos que se miden son: Exactitud, Completitud, Consistencia y Unicidad. En el Cuadro 1 se muestran todos los problemas identificados para el experimento, por cada factor y cada dimensión de calidad.

Cuadro 1. Problemas de Calidad en los Datos.

Dimensión	**Factor**	**Problema de calidad**	**Id.**
Exactitud	Exactitud sintáctica	Valor fuera de rango	P1
		Estandarización	P2
	Exactitud semántica	Registro inexistente	P3
		Defecto mal registrado	P4
		Valor fuera de referencial	P5
Completitud	Densidad	Valor nulo	P6
		Clasificación de defecto	P7
Consistencia	Integridad intra-relación	Reglas de integridad intra-relación	P8
		Valor único	P9
	Integridad referencial	Referencia inválida	P10
Unicidad	Duplicación	Registro duplicado	P11
	Contradicción	Registro contradictorio	P12

Para todos los casos, la unidad de medida del resultado es booleana: se mide si el objeto contiene o no un problema. Los problemas se miden, en la mayoría de los casos y salvo que se indique lo contrario, mediante la definición y ejecución de consultas SQL.

Por motivos de espacio presentamos únicamente la discusión de uno de los problemas detectados.

Registro duplicado.

Se identifica este problema cuando existen dos o más registros que aparecen repetidos de manera exacta. Existen dos situaciones:

- Cuando contienen el mismo valor en la clave y demás atributos (o en su defecto valores nulos). Este caso se contempla con controles del SGBD.
- A pesar de contener distinta clave primaria, hacen referencia al mismo objeto de la realidad y contienen los mismos datos en los campos que se definan. Para este caso se verifica que no existan registros repetidos (según el criterio definido) en la base bajo estudio.

La causa de este problema se puede deber a una equivocación por parte del Verificador (sujeto del experimento en nuestro caso) al registrar varias veces el mismo defecto, o un error de la herramienta que ocasione se almacenen registros repetidos en la base. De no considerar este problema, la cantidad de defectos total que se obtenga no reflejará la realidad, ya que se contarán varios registros que hacen referencia al mismo defecto.

En el caso particular de la métrica "Registro de taxonomías duplicados para ODC y Beizer", el criterio de duplicación definido consiste en identificar los registros de defectos que contienen (para las taxonomías de ODC y Beizer) la misma clasificación dentro de la misma categoría para un mismo sujeto.

5. Resultados y Discusión

Se identificaron 12 problemas de calidad (que fueron presentados en el Cuadro 1), y se definieron un total de 24 métricas para medir estos problemas aplicados sobre objetos (celdas y/o tuplas) de la base.

Del total de métricas definidas el 70 % se midió de forma automática, el 10 % de forma manual y para el 20 % restante no se realizó la medición. La automatización consistió en ejecutar sentencias SQL y, para un caso particular, se incluyeron algoritmos programados en Java. Las mediciones manuales corresponden a verificaciones, realizadas manualmente, contra otras fuentes de datos no persistidas. Por ejemplo, consultas a responsables del experimento. Diversos motivos imposibilitaron (hasta el momento) ejecutar la medición de algunas de las métricas definidas. Por ejemplo, para medir la correctitud semántica sobre los registros de defectos se deben verificar manualmente los 1009 registros de defectos existentes, lo cual no se realizó aún por razones del esfuerzo asociado a dicha verificación.

Al medir encontramos que 14 de las métricas dieron como resultado al menos una tupla o celda con la presencia (potencial) de un problema en los datos. El

Cuadro 2 presenta dichas métricas, el objeto medido en la base de datos y el porcentaje de objetos (potencialmente) con problemas de calidad (porcentaje de objetos que tiene problemas de calidad de la totalidad de objetos medidos con dicha métrica). Los nombres de los objetos son nemotécnicos y permite conocer qué fue medido. A modo de ejemplo, el objeto "*Registro_Defecto.tiempo_detección*", indica cuánto tiempo le llevó a un sujeto detectar cierto defecto.

Este análisis de la calidad de los datos permitió, por ejemplo, detectar que la clasificación de defectos contuvo un alto porcentaje de problemas de calidad. Esto se ve reflejado en las mediciones relativas a la clasificación de defectos: un 38,4 % de los objetos medidos presentan el problema de calidad que refiere a nombres de las categorías fuera de un referencial, mientras que un 17,6 % contienen registros de taxonomías duplicados. Habiendo detectado esta situación todo lo concerniente a la clasificación de defectos fue eliminado del análisis estadístico de efectividad y costo de las técnicas [15].

Este estudio de la calidad de los datos también permite conocer para qué datos conviene realizar una limpieza antes de que sean utilizados en el análisis estadístico del experimento. De esta forma se pretende garantizar que el análisis del experimento se realice sobre datos válidos. Caso contrario los resultados, y las conclusiones del experimento pueden incluso no corresponderse con la realidad (y este es justamente el objetivo de los experimentos controlados, estudiar y conocer la realidad).

6. Conclusiones

El aporte principal de este trabajo es haber utilizado un enfoque sistemático, disciplinado y estructurado de la disciplina Calidad de Datos para identificar y medir los problemas de calidad en los datos que se pueden encontrar en experimentos en ingeniería de software. No encontramos en la literatura otros estudios o publicaciones que aborden esta problemática con un enfoque como el que aquí se presenta. Normalmente la literatura, como ya se mencionó, aborda casi exclusivamente la calidad de datos en experimentos como la forma de detectar *outliers*. Entendemos que nuestra propuesta es más amplia ya que considera diversos problemas de calidad (considerando dimensiones y factores de calidad) y no solamente el estudio de *outliers*. También entendemos que el estudio de *outliers* debe realizarse luego de una limpieza como la que aquí presentamos, ya que para identificarlos es razonable que la población sea la "correcta", es decir, trabajar sobre los datos ya analizados respecto a otras propiedades de calidad y, sobre todo, datos ya limpios respecto a esas propiedades.

Además, en este artículo, mostramos el uso de nuestra propuesta en un experimento controlado y los resultados de la medición de la calidad en los datos. En el análisis realizado encontramos que los datos de los experimentos (obviamente) contienen problemas y que en muchos casos se debe realizar una limpieza antes de realizar los cálculos estadísticos. Incluso, respecto a uno de los datos (tipos de defectos clasificados en dos taxonomías), descartamos su uso en el análisis estadístico debido a la mala calidad de esos datos.

Cuadro 2. Métricas aplicadas sobre objetos de la base con resultado de al menos una tupla o celda con presencia de problemas.

Id. Prob.	**Métrica**	**Objeto(s)**	**%Prob.**
P1	Valor fuera de rango en tiempos	Registro_Defecto.tiempo_deteccion	2,1 %
	Valor fuera de rango en líneas de código	Registro_Defecto.linea	0,8 %
		Registro_Defecto.linea_estructura	0,3 %
P3	Archivo de software inexistente	Archivo	2,1 %
P5	Nombre de la categoría fuera de referencial	Registro_Taxonomia	38,4 %
P6	Valor nulo en atributos sin definición (*not null*)	Experimento.tiempo_ejecucion	6,8 %
		Registro_Taxonomia.taxonomia_id	13,7 %
		Registro_Taxonomia.valor_categoria_id	0,02 %
		Experimento.tiempo_casos (técnica dinámica)	6,8 %
P7	Clasificación de defectos según ODC	Registro_Defecto (ODC)	0,3 %
P8	Regla de integridad en el tiempo de detección de defectos para técnicas estáticas	Registro_Defecto.tiempo_deteccion	8,9 %
	Regla de integridad en el tiempo de ejecución de casos	Experimento.tiempo_ejecucion	4,5 %
P10	Referencia en taxonomías y registro de defectos	Registro_Taxonomia.registro_id	4,9 %
	Referencia inválida en jerarquía de categorización para Beizer	Valor_Categoria.categoria_padre	10 %
P11	Registro de taxonomías duplicados para ODC y Beizer	Registro_Taxonomia (ODC)	17,6 %
P12	Archivo de software contradictorio	Archivo_Software	8,3 %
	Clasificación contradictoria para ODC	Registro_Taxonomia (ODC)	1,5 %
	Registro de taxonomía contradictorio para Beizer	Registro_Taxonomia (para Beizer)	0,7 %

Desde la perspectiva de la ingeniería de software empírica este artículo busca concientizar acerca de la importancia que tiene la disciplina de calidad de datos en los experimentos controlados. Desde la perspectiva de la calidad de datos, este trabajo muestra una aplicación de las técnicas de medición de calidad y de limpieza de datos a un dominio particular.

Referencias

1. Bachmann, A.J.E.: Why Should We Care about Data Quality in Software Engineering? Ph.D. thesis, University of Zurich (2010)
2. Basili, V.R., Selby, R.W.: Comparing the effectiveness of software testing strategies. IEEE Transactions on Software Engineering 13(12), 1278–1296 (Dec 1987)
3. Batini, C., Scannapieco, M.: Data Quality: Concepts, Methodologies and Techniques. Springer-Verlag Berlin Heidelberg (2006)
4. Beizer, B.: Software Testing Techniques, Second Edition. Van Nostrand Reinhold Co. (1990)
5. Chillarege, R.: Handbook of Software Reliability Engineering. IEEE Computer Society Press, McGraw-Hill Book Company (1996)
6. Etcheverry, L., Peralta, V., Bouzeghoub, M.: Qbox-foundation: a metadata platform for quality measurement. In: 4th Data and Knowledge Quality Workshop (2008)
7. Jedlitschka, A., Ciolkowski, M., Pfahl, D.: Reporting experiments in software engineering. In: Shull, F., Singer, J., Sjøberg, D.I.K. (eds.) Guide to Advanced Empirical Software Engineering, pp. 201–228. Springer London (2008)
8. Juristo, N., Moreno, A.M.: Basics of Software Engineering Experimentation. Kluwer Academic Publishers, 1st edn. (2001)
9. Lee, Y.W., Strong, D.M., Kahn, B.K., Wang, R.Y.: Aimq: a methodology for information quality assessment. Inf. Manage. 40, 133–146 (2002)
10. Liebchen, G.A.: Data Cleaning Techniques for Software Engineering Data Sets. Ph.D. thesis, Brunel University (2010)
11. Liebchen, G.A., Shepperd, M.: Data sets and data quality in software engineering. In: Proceedings of the 4th international workshop on Predictor models in software engineering. pp. 39–44. PROMISE '08, ACM, New York, NY, USA (2008), http://doi.acm.org/10.1145/1370788.1370799
12. Neely, M.P.: The product approach to data quality and fitness for use: A framework for analysis. In: Proceedings of the 10th International Conference on Information Quality MIT (2005)
13. Pickard, L.M., Kitchenham, B.A., Jones, P.W.: Combining empirical results in software engineering. Information and Software Technology 40(14), 811 – 821 (1998), http://www.sciencedirect.com/science/article/pii/S0950584998001013
14. Strong, D.M., Lee, Y.W., Wang, R.Y.: Data quality in context. Commun. ACM 40, 103–110 (1997)
15. Vallespir, D., Apa, C., De León, S., Robaina, R., Herbert, J.: Effectiveness of five verification techniques. In: Proceedings of the XXVIII International Conference of the Chilean Computer Society (2009)
16. Wang, R.Y., Reddy, M.P., Kon, H.B.: Toward quality data: an attribute-based approach. Decis. Support Syst. 13, 349–372 (1995)
17. Wohlin, C., Runeson, P., Höst, M., Ohlsson, M.C., Regnell, B., Wesslén, A.: Experimentation in software engineering: an introduction. Kluwer Academic Publishers, Norwell, MA, USA (2000)