Contents lists available at ScienceDirect

Data in Brief

journal homepage: www.elsevier.com/locate/dib

Data Article

Two datasets of defect reports labeled by a crowd of annotators of unknown reliability

Jerónimo Hernández-González [a,*], Daniel Rodriguez [c], Iñaki Inza [a], Rachel Harrison [d], Jose A. Lozano [a,b]

[a] Department of Computer Science and Artificial Intelligence, University of the Basque Country UPV/EHU, Donostia, Spain
[b] Basque Center for Applied Mathematics BCAM, Bilbao, Spain
[c] Department of Computer Science, University of Alcala, Madrid, Spain
[d] Department of Computing, Oxford Brookes University, Oxford, UK

ARTICLE INFO

Article history:
Received 4 November 2017
Received in revised form
7 November 2017
Accepted 23 March 2018
Available online 28 March 2018

ABSTRACT

Classifying software defects according to any defined taxonomy is not straightforward. In order to be used for automatizing the classification of software defects, two sets of defect reports were collected from public issue tracking systems from two different real domains. Due to the lack of a domain expert, the collected defects were categorized by a set of annotators of unknown reliability according to their impact from IBM's orthogonal defect classification taxonomy. Both datasets are prepared to solve the defect classification problem by means of techniques of the learning from crowds paradigm (Hernández-González et al. [1]).

Two versions of both datasets are publicly shared. In the first version, the raw data is given: the text description of defects together with the category assigned by each annotator. In the second version, the text of each defect has been transformed to a descriptive vector using text-mining techniques.

DOI of original article: https://doi.org/10.1016/j.asoc.2017.10.047

* Corresponding author.
E-mail addresses: jeronimo.hernandez@ehu.eus (J. Hernández-González), daniel.rodriguezg@uah.es (D. Rodriguez), inaki.inza@ehu.eus (I. Inza), rachel.harrison@brookes.ac.uk (R. Harrison), ja.lozano@ehu.eus (J.A. Lozano).

https://doi.org/10.1016/j.dib.2018.03.109

Specifications Table

Subject area	*Computer Science*
More specific subject area	*Software Engineering; Machine Learning*
Type of data	*Text fields and multiple annotations of a discrete class variable (defect impact* of IBM's orthogonal defect classification, *ODC* [2]*).*
How data was acquired	*Gathered from public issue tracking systems for the defect descriptions. Manual annotation of each defect by different labelers.*
Data format	*Both raw and text-processed datasets.*
Experimental factors	*In the second version of the datasets, standard natural language processing techniques were used to extract a relevant set of variables from the text fields and transform the original database into a dataset which can be handled by machine learning techniques.*
Experimental features	*A description of the datasets and the agreement between the labels of the different annotators is provided.*
Data source location	http://compendium.open.ac.uk/bugzilla/ http://bugzilla.mozilla.org/
Data accessibility	*All the data is published together with this article.*
Related research article	*The data in this paper was used in [1].*

Value of the Data

- A large set of software defect reports is collected (and processed) from public repositories and adapted for the task of defect classification.
- Five labelers for each dataset give their annotations by means of the most convenient defect impact from the ODC taxonomy [2], according to their subjective point of view.
- As no ground truth is available, the evaluation of classification models learnt from this type of data is a challenge that has not been solved to date.
- The processing and extraction of meaningful information from the text fields that may boost the performance of the learning algorithms may be improved.

1. Data

In software engineering, having available the classification of the reported defects is useful, although this task is complex and time consuming. The datasets presented in this paper were prepared to learn to automatize the classification of software defects by means of the paradigm of (machine) learning from crowds. Thus, multiple annotators were asked to provide the most convenient category (label) for each defect report according to their subjective point of view. Note, therefore, that the labels provided by the annotators may be wrong. To enhance generalization, two datasets were prepared.

The first dataset is composed of reports collected from the issue tracking system (ITS) of the Compendium project, a software tool for mapping information, ideas and arguments. An ITS is typically used by software projects for reporting and tracking defects as well as proposing new functionalities. An ITS organizes the information through tickets, which maintain data such as an identifier, summary, description, opening/closing/modification dates, etc. The ITS used by the Compendium project is implemented in Bugzilla (https://www.bugzilla.org/) and collects support issues, feature requests and bug reports from the Compendium community. The collected dataset comprises 962 examples, all the entries available in the ITS in August 2014 (with the exception of some obvious spam). Only informative fields are taken into account: severity, summary and description. Severity is a 3-value variable (Bug, Support or Feature), and both summary and description are text fields. Five annotators with experience in computer science were asked to annotate the examples according to