Genomic Variant Classifier Tool
Grau Garcia, Isel Del Carmen; Garcia Lorenzo, Maria M.; Sengupta, Dipankar; Farid, Dr Dewan; Manderick, Bernard; Nowe, Ann; Daneels, Dorien; Bonduelle, Maryse; Croes, Didier; Van Dooren, Sonia

Published in:
Lecture Notes in Networks and Systems

Publication date:
2016

Document Version:
Proof

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
Genomic Variant Classifier Tool

Isel Grau  
Department of Computer Science  
Universidad Central “Marta Abreu” de Las Villas  
Santa Clara, Cuba  
COMO, Artificial Intelligence Laboratory  
Vrije Universiteit Brussel  
Brussels, Belgium  
igrauger@uab.cu

Dipankar Sengupta, Dewan Md. Farid,  
Bernard Manderick, Ann Nowe  
COMO, Artificial Intelligence Laboratory  
Vrije Universiteit Brussel  
Brussels, Belgium

Maria M. Garcia Lorenzo  
Department of Computer Science  
Universidad Central “Marta Abreu” de Las Villas  
Santa Clara, Cuba  

Dorien Daneels, Maryse Bonduelle, Didier Croes,  
Sonia Van Dooren  
Centre for Medical Genetics, Reproduction and Genetics,  
Reproduction Genetics and Regenerative Medicine,  
Vrije Universiteit Brussel, UZ Brussel  
Brussels, Belgium

Abstract—The exome or genome based high throughput screening techniques are becoming a definitive criterion in the conventional clinical analysis of the genetic diseases. However, pathogenic classification of an identified variant, is still a manual and time consuming process for clinical geneticists. Thus, to facilitate the variant classification process, we have developed GeVaCT, a Java based tool that implements a classification approach based on the literature review of cardiac arrhythmia syndromes. Furthermore, the adoption of this automated knowledge engineer by the clinical geneticists will aid to build a knowledge base for the evolution of the variant classification process by use of novel machine learning approaches.

Keywords — variant classification; automated knowledge engineer; genomic variant; cardiac arrhythmia syndromes.

I. INTRODUCTION

With the emergence of new screening techniques, targeted or whole exome and genome screening are becoming standard diagnostic norms in clinical settings to identify the variants for a genetic disease [1]. However, the development of Bioinformatics solutions for pathogenic classification of the variants is still an open problem and henceforth, the existing process can be considered as ponderous for geneticists.

The literature cites for relations amid features of the genomic variants and their respective pathogenic classification [2-4], however, the analysis of these features is still a manual and time consuming process. On the other side, this Bioinformatics problem can be considered as a supervised classification problem, where the use of interpretable machine learning techniques (such as Decision Trees or Fuzzy Cognitive Maps) can lead to an accurate prediction model along with the discovery of causal relations among the variant features and the pathogenic class labels. Regrettably, the literature cites lack of knowledge bases relating the genomic variants to their respective pathogenic class labels.

In this demo paper, we describe GeVaCT (Genomic Variant Classifier Tool), a Java based tool for pathogenic classification of genomic single nucleotide and short insertion/deletion variants. GeVaCT addresses two main concerns in the domain of variant classification: firstly we aim to design and implement a default classification approach, based on a literature review of cardiac arrhythmia syndromes and existing knowledge from clinical geneticists; on the other side, the use of the proposed tool generates a knowledge base of variants and their computed classification in a transparent manner, acquiring the available knowledge as an automated knowledge engineer.

II. DEFAULT CLASSIFICATION ALGORITHM

The default classification algorithm designed for GeVaCT is mainly based on a published variant classification schema for cardiac arrhythmia syndromes. This approach is supported by a yield of DNA testing over a time span of 15 years (1996-2011), between probands with isolated/familial cases, and also between probands with or without clear disease-specific clinical characteristics [2]. Based on these studies, there are two approaches proposed: one to classify missense variants and another to classify nonsense & frameshift variants. However, both approaches have a certain scoring criteria in common, which was taken into account during the implementation, to have a minimal computational time. The current version of GeVaCT only supports classification of variant files annotated by Alamut Batch, with a future plan to support the inputs from other annotation software tools.

The proposed algorithm is implemented in two phases: pre-processing and classification. In the pre-processing phase, the annotated tab-delimited variant file (.vcf.ann extension) from the Alamut batch, is refined based on the gene list for a disease of interest. This first step allows to reduce the number of variants for the analysis. Secondly, filters are applied to look for variants that have been already reported in the Human Genome Mutation Database [5] and in ClinVar [6], or those which have been previously detected and classified in an internal patient population. And lastly, the variants are filtered...
based on their location in the genome and their coding effect, followed by the check for minor allele frequency of the variant in a control population [7].

Thereafter, in the classification phase, the filtered variants are separately classified as missense or nonsense and frameshift variants. For missense variants the classification is based on the values of following parameters: amino acid substitution and its impact on protein function [8-9], biochemical variation [10], conservation [11], frequency of variant alleles in a control population [12], effects on splicing [13], family and phenotype information and functional analysis. Whereas for the nonsense and frameshift variants, the classification is based on: effects on splicing, frequency of variant alleles in a control population, family and phenotype information and functional analysis. In the current methodology some of the steps require the intervention of the expert; however, this process has been designed in a user’s reciprocate fashion, to facilitate and accelerate the computation of the result. For each computed parameter in the classification process, a score is assigned to the variant, which is subsequently aggregated. Decisively, based on the cumulative score each variant is thereby classified into one of the following five classes: Non-Pathogenic, Unlikely Pathogenic, Unclear, Likely Pathogenic, or Pathogenic [14].

The validation of the tool is based on the testing of 130 cardiac arrhythmia syndrome patients, available at UZ Brussel. The results of the variant classification made by the tool were validated by manual curation, performed by the clinical geneticist.

III. KNOWLEDGE BASES GENERATION

One of the first challenges on building a knowledge-based system is the acquisition of expert and literature knowledge. Although for certain application domains, there are plenty of data available, regrettably, this is not the case of genomic variant classification. Within artificial intelligence, the automated knowledge engineering tools aim to acquire the knowledge from different sources and then to organize them in an effective representation [15].

GeVaCT works also as an automated knowledge engineer in a totally transparent process for the end user (i.e. clinical geneticist). The class computed for each variant is recorded and the tool is capable of saving this information in an Attribute Relation File Format (.arff) file. This extension is widely use in machine learning platforms such as WEKA [16], facilitating the future analysis of the data. It is also possible to save the knowledge base in tab-delimited files.

Conclusively, based on the validation process described, we should obtain 130 knowledge bases. This will relate the genomic variants described by several features with their obtained classification generated by the default classification approach. The formulation of this problem as a supervised classification problem and its corresponding treatment with machine learning techniques could help the experts in the classification process by the elimination of manual intervention. Furthermore, since only approximately 10 of 150 variant features are currently taken into account, the use of interpretable machine learning techniques could help on gaining insights in the causal relations between other features and the pathogenic classification.

FINAL REMARKS

In this demo paper, we report a Java based tool called GeVaCT, developed for classification of genomic variants. The tool has been tested on a dataset of 130 cardiac arrhythmia syndrome patients, available at UZ Brussel. The contribution of GeVaCT is twofold: we design and implement a classification algorithm for labelling genomic variants, automating a manual and time consuming process for clinical geneticists; and at the same time, we build a knowledge base of variants and their computed classification in a transparent manner, as an automated knowledge engineer. Future work includes the extension of the tool for supporting the inputs from other annotation software tools. Conclusively, the GeVaCT tool is available in the public domain as a downloadable executable file [17].

REFERENCES