İSTANBUL TECHNICAL UNIVERSITY
FACULTY OF COMPUTER AND INFORMATICS

VISUALIZATION OF MUTATIONS COLLECTED FROM PUBLIC DATABASES

Graduation Project Interim Report

Burak Bozdağ
150170110

Muhammed Furkan Kamer
150160013

Department: Computer Engineering
Division: Computer Engineering

Advisor: Asst. Prof. Dr. Mehmet Baysan

February 2021

Statement of Authenticity

I/we hereby declare that in this study

1. all the content influenced from external references are cited clearly and in detail,
2. and all the remaining sections, especially the theoretical studies and implemented software/hardware that constitute the fundamental essence of this study is originated by my/our individual authenticity

07.02.2021

Burak Bozdağ Muhammed Furkan Kamer

VISUALIZATION OF MUTATIONS COLLECTED FROM PUBLIC DATABASES

(SUMMARY)

Mutations collected from different genes can be useful to detect possible effects of nearby or similar genes on both humans and animals. This data is useful for diagnosis regarding diseases and can be used to distinguish them with their treatment. Researchers store this data in format called ‘VCF’.

Data stored in VCF file format includes but not limited to mutation data, base range, alleles to determine its effects. To make it easier for geneticists to observe these effects there are many annotation methods and applications that exist but their output as well as interface are not simple and output files are hard-to-read.

To overcome these issues a web application is designed. Web application includes a simple interface to upload VCF files. After it gets data as VCF file input. Parses these data and gathers information about these input from public databases and annotates. Information gathered from these databases is visualized on data tables and graphs. Application distinguished from other annotation platforms of its simplicity and its output format. Application annotates input info from info scanned from public databases, processes them and shows them in a table manner also includes models of genes with visualization of alternation of nucleotides. These models are interactive models that users can walk up and down to observe alternates. Thus, it simplifies to recognize changes in a broader manner. If the mentioned gene can’t be found about in public databases, nearby genes will be included and its effects estimated accordingly.

HERKESE AÇIK VERİTABANLARINDAN TOPLANAN MUTASYONLARIN GÖRÜNTÜLENMESİ

(ÖZET)

Farklı genlerden elde edilen mutasyon verileri hastalıklar üzerinde bu genler veya bu genlere konum olarak yakın genlerin etkilerini ölçmede kullanışlı bir araçtır. Bu veriler bilim insanları tarafın 'VCF' dosya uzantılı formatlarda saklanmaktadır.

VCF dosyalarında tutulan veriler aynı zaman mutasyon verileri dışında, genin bulunduğu baz aralığı, alel genler gibi birçok veriyi de içerir. Bu verilerden yola çıkılarak genlerin hastalıklar ve organizma üzerindeki etkileri çeşitli açıklamalarla bu verilerin zenginleştirilmesiyle elde edilir fakat bu yöntemlerin birçoğunda elde edilen çıktı insanların anlamasını güçleştiren unsurlar da içermektedir.

Bu güçlüklerin üstesinden gelmek için bir web uygulaması tasarladık. Bu web uygulaması verilen tutulduğu VCF dosyaların karşıya yüklenebileceği basit bir arayüz içermekte ve dosya bu arayüz aracılığıyla karşıya yüklendikten sonra gerekli açıklamalar insanların daha rahat bir şekilde görebileceği ve anlayabileceği formatlarda (tablo, grafik, modelleme) ile bu verilerin zengin halini göstermektedir. Bu kolay kullanışlılığı ile diğer uygulama ve yöntemlerden ayrılmakta ve bu alanlarda çalışan bilim insanlarının işini oldukça kolaylaştırmaktadır. İlgili dosyada bulunan mutasyon verileri ile bilgiler geniş bir veri tabanı süzgecinde elde edildiğinden daha geniş bir alana hitap etmektedir.

Contents

1 Introduction and Problem Definition....4
2 Literature Survey....6
3 Novel Aspects and Technological Contributions....7
4 System Requirements....8
    4.1 Use Cases / User Stories....8
5 Project Plan....9
    5.1 Project Resources....9
    5.2 Work Breakdown and Work Assignment....9
    5.3 Time Plan....10
6 Goals and Evaluation Criteria....11
7 References....12

1 Introduction and Problem Definition

The area of bioinformatics continues to develop and evolve over time. One of the main topics in this area is to annotate variants that occur in live cells. Some variants are natural but some variants can be unexpected and cause cells to become cancerous. Annotating variants help scientists about researching the behavior of genes which are changed by specific variants.

But there is another issue about annotation: The most popular tools that are developed for annotating variants give output in a way that requires detailed documentation in order to read and understand. This means that the output of the annotation process is not human-readable and hard to analyze. Also, we have not found any tool that provides annotation with more readable and understandable output in the literature.

In this project, our aim is to provide annotation results in a way that everyone can read and understand the output of this annotation process. We have seen that the literature needs a tool such as this in order to help bioinformatics develop and improve.

The main characteristics of our system to be implemented can be listed as:

- Variant annotation: As the name suggests, we will implement our feature that annotates given variants (mutations) which are in VCF file format.

- Human-readable output: Our system will give results in a way that everyone can read and understand clearly. The given information will still include technical details but users will understand given information easily when they search about unknown terms on the Internet.

- Interactive visualization: One of our plans about the project is to include visualization of the gene and variant or mutation. This process will ease the understandability of the output when it is implemented.

- Estimating unknown variants' effects: We will provide known variants' effects on the human body, but we will also implement a system that accepts unknown variants and predicts the potential diseases and effects of the given variants.

The given output is planned to include these columns:

- Official symbol: The symbol for the gene which is affected by the given variant. Official symbols are often given by HGNC to define genes separately.

- Official full name: The full name of the related gene. This name also comes from the HGNC database.
- Primary source: The source of the gene in HGNC database.

- See related: This area often includes a string that can be searched on Ensembl genome browser for more further technical details.

- Gene type: Protein-coding genes or pseudo genes (non-coding).

- RefSeq status: The status of the gene in RefSeq database. Genes are researched and added to the database. This status variable indicates the current progress of the research about the gene.

- Organism: The organism of the gene. Our project will be more focused on humans but other variations from other organisms can also be annotated briefly.

- Lineage: Indicates the ancestry, family of the organism in a chained structure.

- Also known as: Other unofficial symbols for genes in the literature.

- Summary: A brief description of the gene. This section can include possible effects of the related gene and variation.

- Expression: This is about tissues which used the related gene during a process such as protein production. This information also indicates that if a mutation occurs in this gene, these tissues would be likely affected.

2 Literature Survey

There are different annotation tools but the most important one is the ANNOVAR.[1] ANNOVAR (ANNOtate VARiation) is a bioinformatics software that is written mainly in the Perl language. It can be run via terminal commands. As a widely used software, it lacks an interface other than command-line commands.

ANNOVAR includes Gene-Based annotation, Region-Based annotation, and Filter-based annotation. It gets VCF formatted input and gives annotated data in CSV format. It can use public databases and has a long background but its output can sometimes not be human-readable as it is. [1]

Thus, we have decided a system is needed with a simple interface that does not force the end-user to use the terminal. Also, this system can give an output with more wide details and ranges on annotated data, not a raw CSV file. Also, a web application can be easily reachable to nearly every computer or smartphone with an internet connection so the terminal ceases to be a requirement or a must for simple annotation tasks.

There are also tools for estimation of variants' association with disease groups such as blood etc. It uses weighted and unweighted prediction algorithms to predict disease association and gives the result as a prediction as a score. Thus, it gives an insight about our application to use these prediction methods to give users a more comprehensive understanding.

NCBI is also a web app that can be used on searching variants in databases like SNP etc. but one cannot search a VCF file that contains thousands of individual variants in the website one by one. Our app does this by sending requests to NCBI for every variant at the same time and shows important data got from NCBI in an extremely short time for a human to do so and enable to walk through between variant data.[2]

3 Novel Aspects and Technological Contributions

In this project, we are aiming to implement a system that does the work of classical annotation tools but in a much beautified manner. The main difference between our project and other annotation tools is the output of the system. In this project, we are focusing on how to give results in a more readable and understandable way that everyone can understand the given output.

Some tools can give useful knowledge such as the symbol of genes, the organism, etc. But in order to understand the meaning of these concepts, an additional research is needed. Our project is aimed to eliminate this time-killing process and give every detail of the query to the user. This way, our system will be one of the most user-friendly annotation tools in the literature.

We also want to implement a feature that predicts the effect of the given variant using an algorithm that is developed by ourselves. Although there are tools that do this work, we are willing to implement our own algorithm in our system.

Our technical contributions to the literature will be:

- A new concept of annotating variants: As we said above, we are willing to provide a system with a much better usability and understandability. Technical information will also be given, but the need for further researching and surfing on the Internet will be prevented because our system will give the required information within its results for everyone that is using our system.

- Better interface for better understandability: Our project will be implemented as a web server, so we will also work on the user interface (UI) and user experience (UX) for presenting results in a beautified way. Many annotation tools use command line interfaces (terminals). This way, we will provide an interface which is more usable than other tools.

4 System Requirements

- Fast responding website with simple interface
- Users can upload local VCF files
- System shows annotated data in data table, graphical and model form with no exception
- User can filter data of these forms easily
- User can search a keyword

4.1 Use Cases / User Stories

User Stories

Person 1 (Annotation)
Uploads VCF file
Annotated data shows up as Data table
Filters table by column value
Searches gene or data by keyword
Downloads annotated data as csv

Person 2 (NCBI)
Uploads VCF file
NCBI data shows up as Data table, Graphs and Models
Filters table by column value
Searches gene or data by keyword
Analyzes graphs
Walks through model to observe variations
Downloads NCBI data as csv file

5 Project Plan

This section contains the project plan which is prepared for determining resources, work breakdown, work assignment and time plan.

5.1 Project Resources

In our project, we will use the following resources in order to develop the system properly:

- Software development computer with at least 4GB memory
- A modern browser
- Hosting (Heroku)
- Flask backend framework
- Python (version >= 3.7)
- Numpy library (version >= 1.19)
- JavaScript (version >= ECMAScript 6)
- HTML5
- CSS3

5.2 Work Breakdown and Work Assignment

We tried to distribute the work evenly between us. Our work breakdown and work assignment can be listed as below in an itemized manner:

- Software Requirement Analysis

- System Design
    - Web Server (Muhammed Furkan Kamer)
    - Database (Muhammed Furkan Kamer)
    - Processing (Burak Bozdağ)

- Software Design
    - Frontend (Muhammed Furkan Kamer)
    - Backend (Muhammed Furkan Kamer)
    - Database Operations (Muhammed Furkan Kamer)
    - Visualization and Processing (Burak Bozdağ)
    - Other Imported Modules

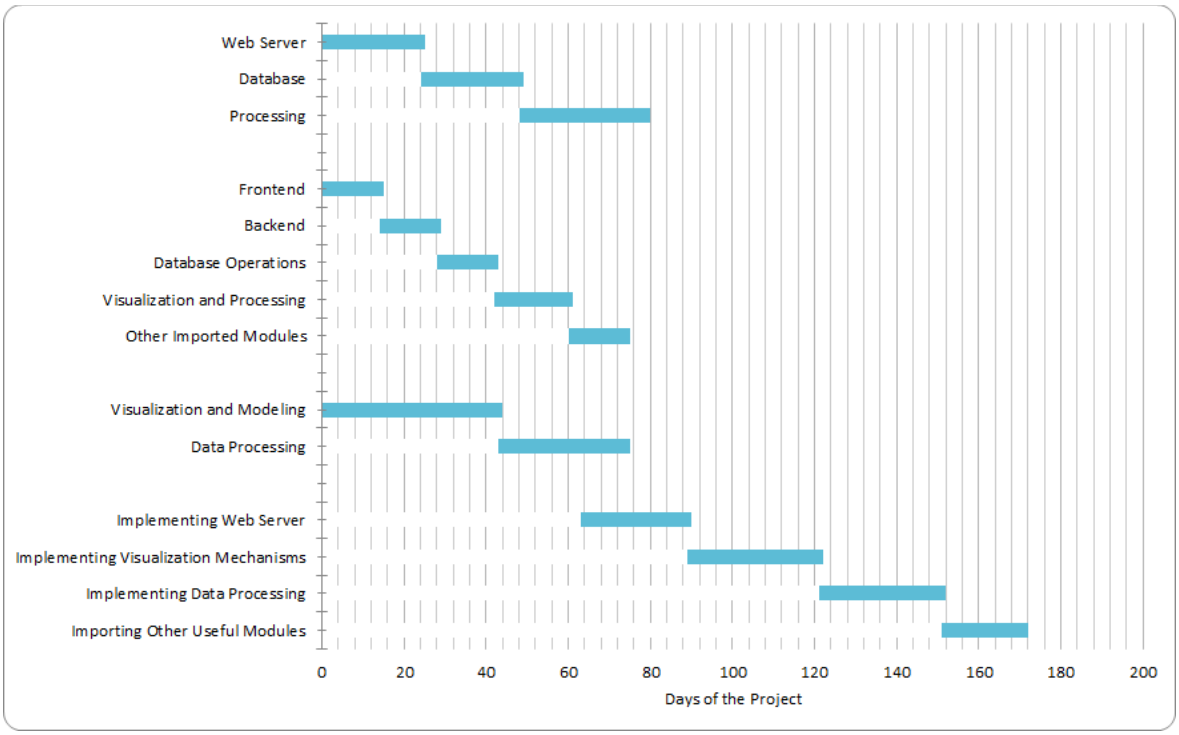
- Visualization and Data Processing System Design
    - Visualization and Modeling (Burak Bozdağ)
    - Data Processing (Muhammed Furkan Kamer)

- Interim Report Preparation
- Software Development
    - Implementing Web Server (Muhammed Furkan Kamer)
    - Implementing Visualization Mechanisms (Burak Bozdağ)
    - Implementing Data Processing (Muhammed Furkan Kamer)
    - Importing Other Useful Modules (Burak Bozdağ)
- Mutation Data Collection

5.3 Time Plan

The GANTT diagram based on the tasks in the subsection is present below.

Table 5.3: GANTT diagram of the project



6 Goals and Evaluation Criteria

In this section we enlisted our goals at the end of our project.

Afterwards we enlisted roughly the criteria which we will be evaluating the system at the end of the project. Briefly we gave numerical values and other verbal criterias for the functional and non-functional aspects of our project.

- Selection of Data Processing Techniques
    - Selection of Candidate Techniques: At least 2 candidate techniques to be selected

- Visualization Method and Parameters
    - Selection of Candidate Techniques: At least 2 candidate techniques to be selected

- Tagged Mutation Data
    - Number of Tagged Data: At least 10 tagged mutation data

- Information Extraction Engine Object Code
    - Extraction of Information: Successfully extract the information

- Visualization Engine Object Code
    - Visualization of Data: Successfully visualize the data

- Web Client App
    - Smoothly Running App: 50 hours of MTTF

- Unit Test Results
    - Code Coverage: Test pass success for at least %20 code coverage

- Visualization Performance Results
    - Understandability of the Visualized Data: Successful understanding of data

7 References

[1] Yang, H., & Wang, K. (2015). Genomic variant annotation and prioritization with ANNOVAR and wANNOVAR. *Nature protocols*, *10*(10), 1556–1566. https://doi.org/10.1038/nprot.2015.105

[2] Sherry ST, Ward MH, Kholodov M, Baker J, Phan L, Smigielski EM, Sirotkin K. dbSNP: the NCBI database of genetic variation. Nucleic Acids Res. 2001 Jan 1;29(1):308-11.