re-Searcher: GUI-based bioinformatics tool for simplified genomics data mining of VCF files

Daniyar Karabayev[1,*], Askhat Molkenov[1], Kaiyrgali Yerulanuly[1,2], Ilyas Kabimoldayev[1], Asset Daniyarov[1], Aigul Sharip[1], Ainur Seisenova[1], Zhaxybay Zhumadilov[1,3] and Ulykbek Kairov[1,*]

[1] Laboratory of Bioinformatics and Systems Biology, Center for Life Sciences, National Laboratory Astana, Nazarbayev University, Nur-Sultan, Kazakhstan

[2] L.N. Gumilyov Eurasian National University, Nur-Sultan, Kazakhstan

[3] School of Medicine, Nazarbayev University, Nur-Sultan, Kazakhstan

*Equal first author

Corresponding Author:

Ulykbek Kairov[1]

53 Kabanbay Batyr ave., Nur-Sultan, 010000, Kazakhstan

Email address: ulykbek.kairov@nu.edu.kz

Abstract

Background. High-throughput sequencing platforms generate a massive amount of high-dimensional genomic datasets that are available for analysis. Modern and user-friendly bioinformatics tools for analysis and interpretation of genomics data becomes essential during the analysis of sequencing data. Different standard data types and file formats have been developed to store and analyze sequence and genomics data. Variant Call Format (VCF) is the most widespread genomics file type and standard format containing genomic information and variants of sequenced samples.

Results. Existing tools for processing VCF files don't usually have an intuitive graphical interface, but instead have just a command-line interface that may be challenging to use for the broader biomedical community interested in genomics data analysis. re-Searcher solves this problem by pre-processing VCF files by chunks to not load RAM of computer. The tool can be used as standalone user-friendly multiplatform GUI application as well as web application (https://nla-lbsb.nu.edu.kz). The software including source code as well as tested VCF files and additional information are publicly available on the GitHub repository (https://github.com/LabBandSB/re-Searcher).

Introduction

Recent achievements in high-throughput sequencing technologies (Goodwin, McPherson & McCombie, 2016; van Dijk et al., 2018) have led to the generation of massive amounts of genomic data (Gao et al., 2019; The ICGC/TCGA Pan-Cancer Analysis of Whole Genomes Consortium, 2020) available for the research community. Many omics databases have been

40 developed and have collected freely accessible datasets (Molkenov et al., 2019; Rigden &
41 Fernández, 2020) for analysis by the bioinformatics community. Modern bioinformatics tools
42 and methods are in high demand for analyzing and interpreting the big omics data generated by
43 the different types of multi-omics platforms available. Different standard data types and file
44 formats have been developed to store and analyze sequence and genomics data. Variant Call
45 Format (VCF) (Danecek et al., 2011) is a tab-delimited text file format that is often used in
46 bioinformatics to store genomic variants. A VCF file consists of the header, including meta-
47 information lines and field definition lines (column names), and the body (data section). An
48 arbitrary number of meta-information lines start with '##' and provide a description of the VCF
49 file. The body of the file consists of eight mandatory columns: chromosome (CHROM), starting
50 position of a variant (POS), variant identifiers (ID), the reference allele (REF), a list of alternate
51 alleles (ALT), a PHRED-scaled quality score (QUAL), filter information regarding variant
52 validity (FILTER), and annotation information (INFO). Additional columns describing samples
53 can also be added. Each row of the file describes specific genomic variants (SNVs, INDELs,
54 CNVs, and other structural variants) at the given chromosome and genomic position.
55 VCF files often store information about numerous samples and can therefore reach huge sizes –
56 gigabytes, or sometimes terabytes. This creates an issue for the readability of VCF files and
57 further analysis for non-programmers, as manual data extraction and analysis using Microsoft
58 Excel or other table processing software may not be possible due to the RAM capacity limitation
59 of standard computers. We introduce re-Searcher, bioinformatics tool specifically developed for
60 simplified mining and analysis of big-size VCF files. We developed a multi-platform user-
61 friendly graphical user interface (GUI) tool for offline access, while re-Searcher web application
62 can be used online via web browser. re-Searcher has been developed for the broader biomedical
63 community and solves the problem of working and analyzing genomics data stored in VCF
64 format.
65

Materials & Methods
Implementation
68 re-Searcher application was written in Python 3 (Rossum, Drake & Van Rossum, 2010) with the
69 implementation of Tkinter (Lundh, 1999) package to build the GUI, and Pandas (McKinney &
70 PyData Development Team, 2017) library to extract columns. For convenience of users who
71 would like to build re-Searcher into their pipelines command line interface (CLI) is available as
72 python script (see Availability). CLI mirrors functionality of GUI regarding files
73 processing. We developed web version of re-Searcher for users to manipulate VCF files
74 without downloading CLI or GUI versions of re-Searcher. The web application was developed
75 using Django web framework (*Django*, 2013) to run python script via WSGI on Apache web
76 server (*Apache*, 2020). The web version runs re-Searcher scripts on server and takes input files
77 from website and returns processed files to user to download.
78 re-Searcher solves the problem of analyzing large VCF files by not loading the whole file
79 directly into RAM, but instead pre-processing it in chunks and utilizing a simple and intuitive

80   interface (Figure 1). The main advantage of re-Searcher in comparison with other tools is the
81   presence of a simple and user-friendly interface, GUI and web interface, instead of a CLI, as well
82   as a lack of confused installation procedures typical for existing tools. The generalized workflow
83   of re-Searcher consists of several steps: selecting an input file, setting up necessary filtering
84   parameters, data processing, and exporting a filtered output VCF file (Figure 2). re-Searcher
85   browses and opens VCF files with extensions ".txt" or ".vcf", before performing the following
86   filtering and extraction options:

87   **Header extraction**
88   VCF files can be large and, if a user needs to know only certain information in a file header (e.g.
89   a particular meta-line or sample ID), the software can extract only the header from the original
90   VCF file and save it into a new file.

91   **Keyword search**
92   If genomic variants need to be filtered according to the presence of a keyword, the software can
93   find these rows and extract them into a new VCF file. Users may input multiple keywords by
94   accessing the entry field or by uploading a .txt file with keywords.

95   **Sample extraction**
96   If the user needs only particular sample IDs in a VCF file, the software can extract the necessary
97   sample columns into a new file. Similar to a keyword search, users may input multiple sample
98   IDs by accessing the entry field or by uploading a .txt file with the IDs.

99   **Genotype format conversion**
100   re-Searcher can convert numeric genotype (GT) format into letter format. The conversion option
101   is one of the most used operations when working with VCF files, for example, in further
102   comparative analysis of genetic variants or SNPs. The original GT format in VCF files is
103   numeric (*0/0*, *0/1*, *1/1* for biallelic sites or *1/2*, *2/3*, etc. for multiallelic sites), where *0* is a
104   reference (REF) allele, *1* is a first alternative (ALT) allele, *2* is a second ALT allele and so on
105   (Danecek et al., 2011; Campbell et al., 2016). After GT format conversion REF number is
106   replaced with REF letter and ALT number with corresponding ALT letter (Figure 3). For
107   instance, if GT of first sample is *0/1* and GT of second sample is *1/2* in numeric format, while
108   REF and two ALT are *GC, T* and *CAA* respectively, then after conversion first sample's letter
109   GT becomes *GC, T* and second sample's letter GT becomes *T, CAA*.
110
111   The final output file is a filtered and processed VCF file and is generated with a complement log
112   file containing the file processing information, name of specified file and work directory with
113   outputs.
114

115   **Results and Discussion**
116   We have compared the main features of re-Searcher with other existing and open source tools
117   VIVA (Tollefson et al., 2019), VCFtools (Danecek et al., 2011), GEMINI (Paila et al., 2013),
118   BrowseVCF (Salatino & Ramraj, 2016), VCF.Filter (Müller et al., 2017), VCF-Miner (Hart et
119   al., 2016) and prepared a detailed table (Table 1). The compared tools have been developed and

120   implemented in different programming languages (Julia, C++, Perl, Java and Python) and
121   dedicated libraries. VCFtools is one of the most cited and advanced tools for processing VCF
122   files, but it requires additional computational skills for effective usage. VIVA is the only tool
123   that provides the possibility for advanced visualization and plotting figures. In addition to re-
124   Searcher, other tools with a GUI available for users are BrowseVCF, VCF.Filter, and VCF-
125   Miner, while web interface is available in only in re-Searcher (Figure 4) and GEMINI. From
126   these, only re-Searcher, BrowseVCF and VCF.Filter tools support multiple operational systems
127   (Windows, MacOS, Linux). Searching the whole VCF file by keyword and the corresponding
128   extraction of data based on keywords are features available on re-Searcher and BrowseVCF,
129   whereas genotype conversion is a unique feature of re-Searcher.
130   re-Searcher is a multi-platform tool and can be run on MacOS, Windows and Linux operating
131   systems. Performance of re-Searcher has been evaluated on these PC platforms (Windows 10 Pro
132   OS and Linux Mint 17 OS-based PC: CPU Intel Core i5-8250U 1.80 GHz, RAM 4Gb and
133   MacOS Catalina based PC: CPU Intel Core i7, 3.2 GHz, RAM 8 Gb) with different VCF file
134   sizes. Different size VCF files (0.081 Gb, 0.814 Gb, 1.320 Gb, 1.980 Gb and 7.950 Gb) were
135   used as input datasets for evaluating re-Searcher performance. The results of the performance
136   benchmarking are shown in Table 2.
137   We have used big VCF files from the 1000 Genomes Project (The 1000 Genomes Project
138   Consortium, 2015) and then generated the different sized testing VCF files from this dataset. The
139   Linux-based systems had the fastest execution time for different operations in comparison with
140   Windows and MacOS systems.
141

142   **Availability**
143   re-Searcher executable software including source code, tested VCF files and additional
144   information are publicly available on the GitHub repository https://github.com/LabBandSB/re-
145   Searcher. re-Searcher is free bioinformatics tool and open to all users without login and
146   registration requirements and do not require an installation of additional tools. CLI version of re-
147   Searcher is also available on the GitHub repository for incorporation into other pipelines. In
148   addition, web version of re-Searcher is available at https://nla-lbsb.nu.edu.kz.
149

150   ## Conclusions
151   Exploring and analyzing VCF files generated after the bioinformatics processing of sequencing
152   data is one of the important steps performed by researchers during analysis and meta-analysis of
153   genotype/phenotype associations. We have developed and introduced an easy-to-use
154   bioinformatics tool, re-Searcher, with several unique features for mining big VCF files and
155   realized with a simple graphical user interface and web interface that makes it easily available
156   for clinicians and researchers without any computational skills. Several improvements such as
157   visualization options (clustering and plotting functions) with Principal Component Analysis and
158   heatmap methodologies are under future development of re-Searcher.
159

160 **Acknowledgements**

161 This work is dedicated to the blessed memory of Dr.Vasily Ogryzko.

162

163 **References**

164 *Apache HTTP Server*. 2020. Apache Software Foundation.

165 Campbell IM, Gambin T, Jhangiani SN, Grove ML, Veeraraghavan N, Muzny DM, Shaw CA,

166 Gibbs RA, Boerwinkle E, Yu F, Lupski JR. 2016. Multiallelic Positions in the Human

167 Genome: Challenges for Genetic Analyses. *Human Mutation* 37:231–234. DOI:

168 10.1002/humu.22944.

169 Danecek P, Auton A, Abecasis G, Albers CA, Banks E, DePristo MA, Handsaker RE, Lunter G,

170 Marth GT, Sherry ST, McVean G, Durbin R, 1000 Genomes Project Analysis Group.

171 2011. The variant call format and VCFtools. *Bioinformatics* 27:2156–2158. DOI:

172 10.1093/bioinformatics/btr330.

173 van Dijk EL, Jaszczyszyn Y, Naquin D, Thermes C. 2018. The Third Revolution in Sequencing

174 Technology. *Trends in Genetics* 34:666–681. DOI: 10.1016/j.tig.2018.05.008.

175 *Django*. 2013. Django Software Foundation.

176 Gao GF, Parker JS, Reynolds SM, Silva TC, Wang L-B, Zhou W, Akbani R, Bailey M, Balu S,

177 Berman BP, Brooks D, Chen H, Cherniack AD, Demchok JA, Ding L, Felau I, Gaheen S,

178 Gerhard DS, Heiman DI, Hernandez KM, Hoadley KA, Jayasinghe R, Kemal A,

179 Knijnenburg TA, Laird PW, Mensah MKA, Mungall AJ, Robertson AG, Shen H,

180 Tarnuzzer R, Wang Z, Wyczalkowski M, Yang L, Zenklusen JC, Zhang Z, Liang H,

181 Noble MS. 2019. Before and After: Comparison of Legacy and Harmonized TCGA

182 Genomic Data Commons' Data. *Cell Systems* 9:24-34.e10. DOI:

183 10.1016/j.cels.2019.06.006.

184   Goodwin S, McPherson JD, McCombie WR. 2016. Coming of age: ten years of next-generation

185   sequencing technologies. *Nature Reviews Genetics* 17:333–351. DOI:

186   10.1038/nrg.2016.49.

187   Hart SN, Duffy P, Quest DJ, Hossain A, Meiners MA, Kocher J-P. 2016. VCF-Miner: GUI-

188   based application for mining variants and annotations stored in VCF files. *Briefings in*

189   *Bioinformatics* 17:346–351. DOI: 10.1093/bib/bbv051.

190   Lundh F. 1999. An introduction to tkinter.

191   McKinney W, PyData Development Team. 2017. Pandas - Powerful Python Data Analysis

192   Toolkit.

193   Molkenov A, Zhelambayeva A, Yermekov A, Mussurova S, Sarkytbayeva A, Kalykhbergenov

194   Y, Zhumadilov Z, Kairov U. 2019. Transcriptomic Databases. In: *Encyclopedia of*

195   *Bioinformatics and Computational Biology*. Elsevier, 341–351. DOI: 10.1016/B978-0-

196   12-809633-8.20208-2.

197   Müller H, Jimenez-Heredia R, Krolo A, Hirschmugl T, Dmytrus J, Boztug K, Bock C. 2017.

198   VCF.Filter: interactive prioritization of disease-linked genetic variants from sequencing

199   data. *Nucleic Acids Research* 45:W567–W572. DOI: 10.1093/nar/gkx425.

200   Paila U, Chapman BA, Kirchner R, Quinlan AR. 2013. GEMINI: Integrative Exploration of

201   Genetic Variation and Genome Annotations. *PLoS Computational Biology* 9:e1003153.

202   DOI: 10.1371/journal.pcbi.1003153.

203   Rigden DJ, Fernández XM. 2020. The 27th annual Nucleic Acids Research database issue and

204   molecular biology database collection. *Nucleic Acids Research* 48:D1–D8. DOI:

205   10.1093/nar/gkz1161.

206   Rossum G van, Drake FL, Van Rossum G. 2010. *The Python language reference*. Hampton, NH:

207   Python Software Foundation.

208   Salatino S, Ramraj V. 2016. BrowseVCF: a web-based application and workflow to quickly

209   prioritize disease-causative variants in VCF files. *Briefings in Bioinformatics*:bbw054.

210   DOI: 10.1093/bib/bbw054.

211   The 1000 Genomes Project Consortium. 2015. A global reference for human genetic variation.

212   *Nature* 526:68–74. DOI: 10.1038/nature15393.

213   The ICGC/TCGA Pan-Cancer Analysis of Whole Genomes Consortium. 2020. Pan-cancer

214   analysis of whole genomes. *Nature* 578:82–93. DOI: 10.1038/s41586-020-1969-6.

215   Tollefson GA, Schuster J, Gelin F, Agudelo A, Ragavendran A, Restrepo I, Stey P, Padbury J,

216   Uzun A. 2019. VIVA (VIsualization of VAriants): A VCF File Visualization Tool.

217   *Scientific Reports* 9:12648. DOI: 10.1038/s41598-019-49114-z.

218

Figure 1

Main window of re-Searcher GUI
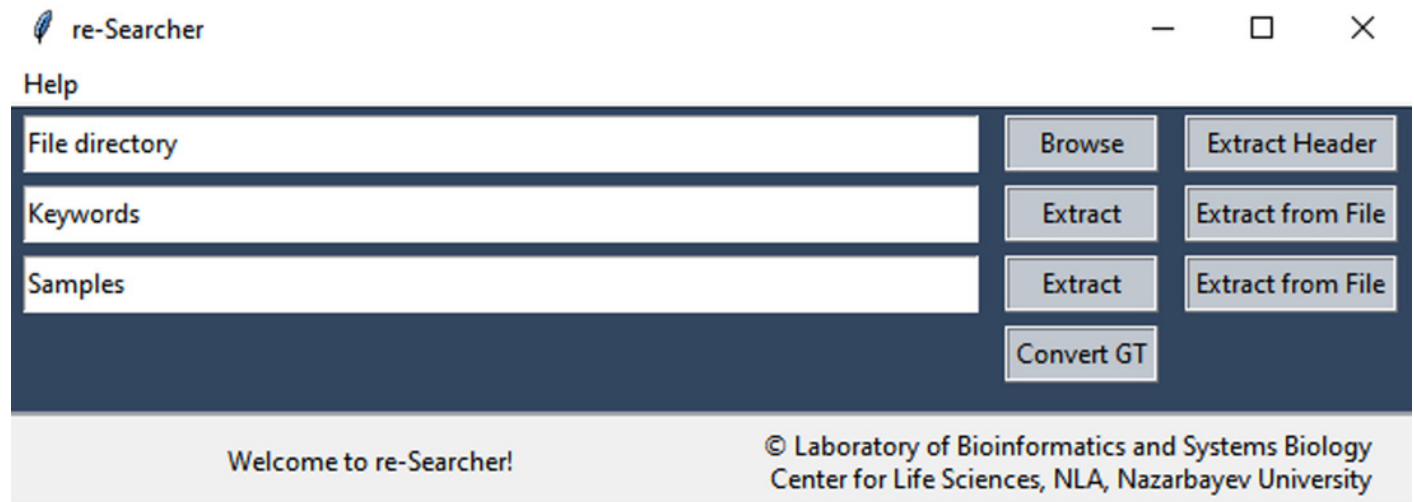
Figure 2

Data processing workflow
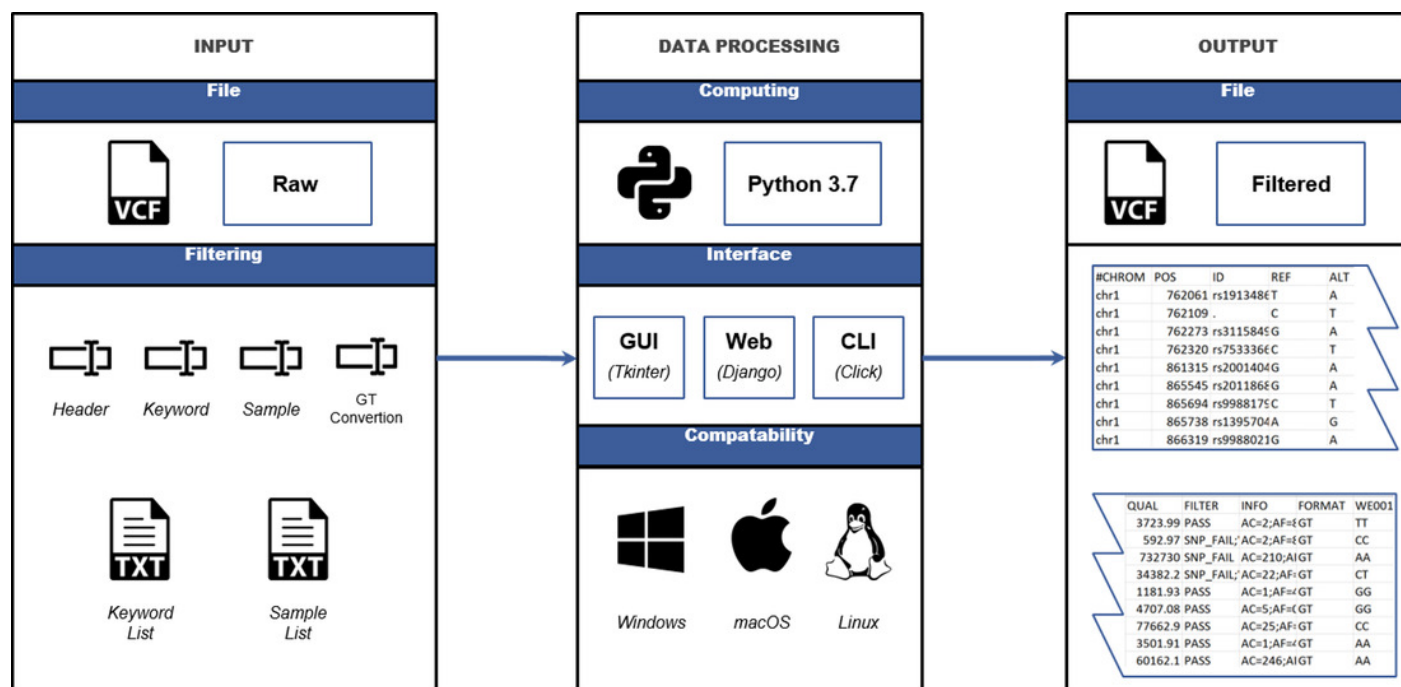
Figure 3

Genotype conversion example

Figure 4

Web interface of re-Searcher available via browser at https://nla-lbsb.nu.edu.kz

Comparison of re-Searcher's features with similar tools

1   **Table 1. Comparison of re-Searcher's features with similar tools**

Categories	Features	re-Searcher	VIVA	VCFtools	GEMINI	BrowseVCF	VCF.Filter	VCF-Miner
Technical Aspects	Compatibility with operation system	Windows, MacOS, Linux	Windows, MacOS, Linux	Windows, MacOS, Linux	Windows, MacOS, Linux	Windows, MacOS, Linux	Windows, MacOS, Linux	Windows
	Language	Python	Julia	C++, Perl	Python	Python, JavaScript, CSS, HTML5	Java	Java
	Interface	GUI, Web Browser, CLI	CLI, Jupyter Notebook	CLI	Web Browser, CLI	GUI, CLI	GUI	GUI
	Works offline	V	V	V	X	X	V	V
	Portable launcher	V	X	X	X	V	X	X
Functionality	Search by keyword	V	X	X	X	V	X	X
	Sample selection	V	V	V	V	V	V	V
	Genotype format conversion	V	X	X	X	X	X	X
	Visualization	X	V	X	X	X	X	X
	Export filtered VCF file	V	X	V	X	X	V	V

re-Searcher multi-platform run time comparison

1 **Table 2. re-Searcher multi-platform run time comparison**
2
3 OS – operational system, GT – genotype, sec – seconds, Gb – gigabyte.

OS	VCF file size (Gb)	Execution Time (sec)			
		Extract header	Keyword extraction	Sample ID extraction	GT conversion
Linux	0.081	2.227	9.995	15.243	38.375
	0.814	2.497	38.106	21.334	117.186
	1.320	2.462	67.260	61.159	206.868
	1.980	2.115	75.331	104.167	330.527
	7.950	6.145	366.137	200.347	1117.168
Windows	0.081	14.865	22.482	4.785	49.642
	0.814	18.898	48.820	21.398	139.641
	1.320	21.054	44.669	59.329	238.192
	1.980	10.919	53.958	90.446	339.275
	7.950	16.308	502.996	309.177	1320.197
MacOS	0.081	9.627	9.297	10.423	16.298
	0.814	5.262	20.916	19.705	116.82
	1.320	6.286	35.433	53.247	186.181
	1.980	3.231	37.923	119.136	286.544
	7.950	5.457	148.612	254.128	1130.225

6
7