ViralFP: A Web Application of Viral Fusion Proteins

Pedro Moreira [1], Ana Marta Sequeira [1], Sara Pereira [2], Rúben Rodrigues [1], Miguel Rocha [1]* and Diana Lousa [2]*

[1] Centro de Engenharia Biológica, Escola de Engenharia da Universidade do Minho, Braga, Portugal, [2] ITQB NOVA, Instituto de Tecnologia Química e Biológica António Xavier, Universidade Nova de Lisboa, Oeiras, Portugal

Viral fusion proteins are attached to the membrane of enveloped viruses (a group that includes Coronaviruses, Dengue, HIV and Influenza) and catalyze fusion between the viral and host membranes, enabling the virus to insert its genetic material into the host cell. Given the importance of these biomolecules, this work presents a centralized database containing the most relevant information on viral fusion proteins, available through a free-to-use web server accessible through the URL https://viralfp.bio.di.uminho.pt/. This web application contains several bioinformatic tools, such as Clustal sequence alignment and Weblogo, including as well a machine learning-based tool capable of predicting the location of fusion peptides (the component of fusion proteins that inserts into the host's cell membrane) within the fusion protein sequence. Given the crucial role of these proteins in viral infection, their importance as natural targets of our immune system and their potential as therapeutic targets, this web application aims to foster our ability to fight pathogenic viruses.

Keywords: fusion proteins, fusion peptides, database, web application, machine learning

1. INTRODUCTION

Enveloped viruses, a group of viruses that includes SARS-CoV-2, Dengue, Influenza and HIV, are characterized by having an exterior lipid envelope that contains at least one type of glycoprotein on its surface (1, Preface I) (2, 3). These proteins play an essential role during the virus entry into the host cell, enabling the binding of the virus to receptors on the host cell and catalyzing the fusion between the viral envelope and the host cell membrane.

In some viruses, a single protein can induce the binding to the host receptor and catalyze the fusion reaction, whereas other viruses require multiple proteins for these tasks. Viral fusion proteins (VFP) are key players in a process called membrane fusion, through which the virus inserts its genetic material into the host cell, enabling it to replicate (4, 5).

During membrane fusion, the viral envelope and the host cellular membrane are brought into close contact in order to merge into a single membrane (6, 7). According to the most widely accepted membrane fusion model, this process is initiated when the VFP binds to the host receptor. Triggered by an external stimulus, the protein becomes extended and inserts its fusion peptide (FPep) into the host membrane. The VFP folds back, which leads to the approximation of the two membranes. After this step, the mechanism passes through a hemifusion intermediate, which encompasses the fusion of the outer leaflets of the membranes without the contact of the inner leaflets. This hemifused step forms a stalk-like structure, diminishing the contact area to a minimum, lowering the free energy barrier required to overcome hydration repulsion. Finally,

fusion of the unattached inner leaflets occurs, which starts to form a hydrophilic pore that will eventually enlarge, connecting the 2 initially separated intermembrane volumes (4, 5, 8–10).

VFPs can assume multiple possible forms depending on their corresponding virus. Currently, the consensus is that VFPs can be grouped into 3 classes: I, II, and III, each with their own characteristics, such as their prefusion, pre-hairpin and post-fusion structures, orientation within virus membrane, types of possible triggering mechanisms and location of their FPep (4, 5, 7, 11).

The FPep is a conserved sequence within VFPs of the same viral family that can insert into the host membrane, allowing the fusion of the viral envelope with the host cell membrane. This peptide can promote the reaction of fusion of membrane vesicles by itself, even if it is detached from the rest of the VFP (4, 11). All FPeps share common characteristics, which are determinant for their function: they are hydrophobic, rich in Gly and Ala residues, contain aromatic residues and are usually conserved within a species (mutations frequently lead to a loss of function) (4, 11). Apart from these general characteristics, FPeps from virus belonging to different families can be quite diverse (4). Some FPeps (e.g., Influenza and HIV) are located at the N-terminal tip of the fusion protein, whereas the peptides of other viruses (e.g., Dengue and Ebola) are internal fusion loops (4). The peptides from different families are also quite different at the sequence level and structure levels. The influenza FP is helical in lipidic environments (12, 13), whereas the HIV FP tends to adopt β-sheet structures (14), although it can become helical depending on membrane composition (15). Other FPeps, such as the one from Dengue virus, which only has 14 residues, do not have a defined secondary structure (16). It is not clear how peptides with such distinct characteristics play a common role in membrane fusion.

The relevance of VFP and their FPep for membrane fusion highlights them as potential therapeutic targets (4, 7). As an example, the therapeutic application of VFPs as a target can be found in the defining pandemic of 2020: COVID-19. Being an enveloped virus, one of the key therapeutic targets of SARS-CoV-2 is its VFP, the spike (S) protein (17). The most promising and effective vaccines under development or already administrated for this virus are based on the S protein's mRNA. This group of vaccines include the Moderna's mRNA-1273 (18) and the Pfizer-BioNTech's BNT162b2 mRNA (19). In both, the mRNA molecule is inserted into a lipid nanocarrier that stabilizes the mRNA and delivers it into human cells. The mRNA once inside a cell is translated to produce the S protein in its prefusion conformation. The presence of those proteins in the human cells induces an immune response on the organism, which will produce antibodies that can target those proteins. For several other enveloped viruses there are known antibodies that target that region, like in cases of Dengue (20), Zika (21), HIV (22), and Influenza (23).

The known information for a given VFP can be found in biological databases, like UniProt (24), NCBI Protein (25) and PDB (26). However, a VFP oriented repository that contains VFP related information does not exist, forcing researchers to search several databases when they need to have comprehensive characterization of one or several of these proteins.

To better understand how VFPs and their components are characterized, some bioinformatics tools can be used, like sequence alignment [e.g., BLAST (27), Clustal (28), and HMMER (29)], that allow researchers to discover a protein's conserved residues, as well as other pieces of information regarding their features. These analyses can provide valuable insights about how VFPs operate.

Researchers have also been using machine learning (ML) to study VFPs and their inhibitors. Several promising tools have been developed using ML models trained using VFP and their features, allowing the prediction of FPeps location within VFP sequences (30, 31) and the prediction of inhibitors for different VFPs to potentially be used in therapies of some diseases (32).

In this context, the main objective of the current work is to develop a CRUD (Create, Read, Update and Delete) web application, named ViralFP, that is capable of storing, retrieving, displaying, and analysing the gathered data on enveloped viruses [through information gathered from NCBI Taxonomy (33) entries] and their VFPs, as well as to enable the data navigation through a user-friendly interface. The web application is freely available and aims to be useful for all elements of the scientific community who are working with VFPs and FPeps. The underlying data are stored in a relational database, which contains relevant information regarding over 500 VFPs. Manual curation of the database information was performed by reviewing the original protein and taxonomy repositories' data entries, as well researching through scientific publications.The back-end component of the application is built using Django and Django REST API library, whilst its front-end is built using Angular technology, a Typescript framework to build web applications. This web application also includes a set of selected bioinformatics tools, such as sequence alignment tools (like BLAST and Clustal) and ML models capable to detect FPeps within the VFP sequence.

2. MATERIALS AND METHODS

The structure of the ViralFP application can be seen in the **Figure 1**. There are three distinct components, each with its own Docker container:

- The MySQL database-stores all the data that the server requires.
- Django back-end, with Django REST framework–enables the access and management of the database, besides containing several data views that will be called by the front-end.
- Angular front-end-the application designed to be used by the end user that displays the database information and allows the execution of several tools.

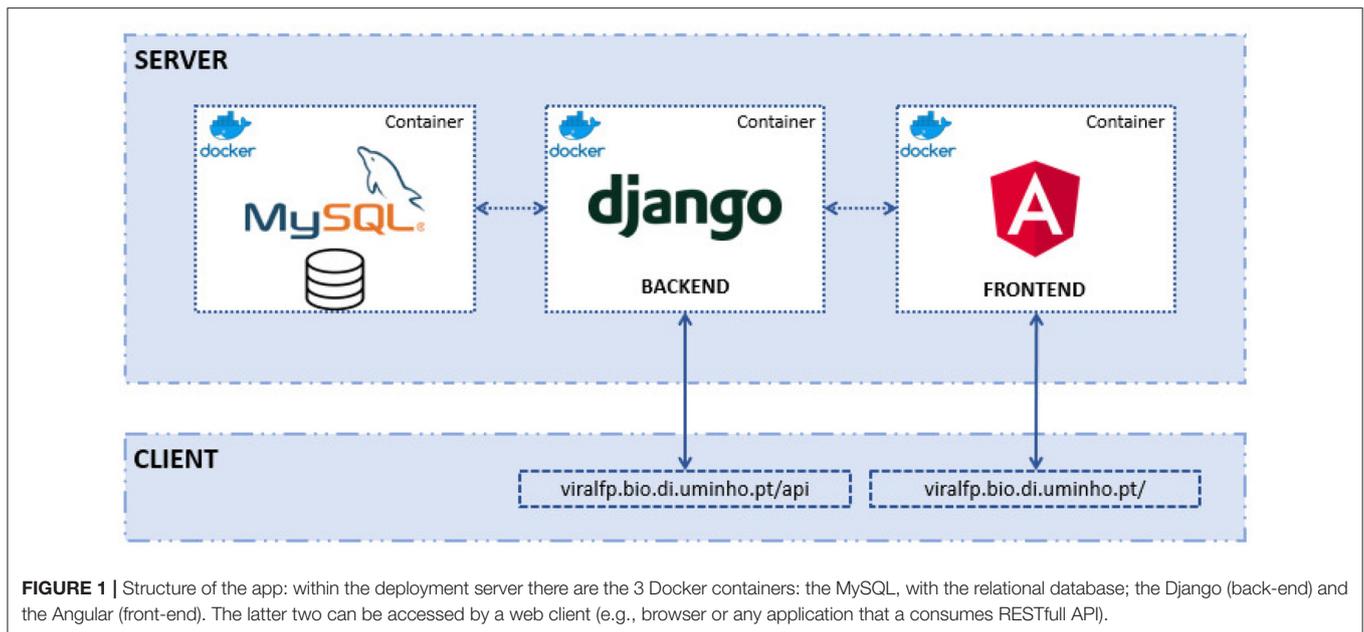In the next sections, a deeper explanation for each of those components will be made.

Abbreviations: FPep, Fusion Peptide; ML, Machine Learning; VFP, Viral Fusion Protein.

FIGURE 1 | Structure of the app: within the deployment server there are the 3 Docker containers: the MySQL, with the relational database; the Django (back-end) and the Angular (front-end). The latter two can be accessed by a web client (e.g., browser or any application that a consumes RESTfull API).

2.1. Database

The VFP/FPep data present in the MySQL database was retrieved from UniProt, NCBI Protein and PDB entries, whilst the taxonomy information was originally from NCBI Taxonomy. Its relational structure, names of the tables and attributes, as well as the type of data stored in each attribute can be seen on the scheme present in **Figure 2**.

The core table of this database is called `Protein` and contains the information regarding each VFP's name, class, type of activation, fusogenic peptide description (its name, location within the protein and sequence), as well as having the UniProt and NCBI Protein entries for the protein.

Another important table is the `Fusion Peptide`, which describes the location of this peptide within the VFP sequence (attribute `Residues`), sequence and some attributes regarding annotation method and experimental evidence.

The last key table is the `Taxonomy`, which has all the relevant taxonomy of the virus to which the protein belongs to (family, genus, species, and subspecies/strain), as well as having the virus NCBI Taxonomy entry.

Additional tables within the database include a table with the possible structures of the VFP (through PDB entries); a table for the virus' hosts, which has the common name and NCBI Taxonomy of each virus host organism; an inhibitor/antibody table, that contains information regarding those types of protein that are associated with a particular VFP, and, finally, tables for additional bibliographical references.

The data present in the database was manually curated by verifying the original repositories of the data entries. Finally, the inhibitors and antibodies entries for each VFP were filled using *BioPython* (34) for the Pubmed automated searches, and *pypdb* (35) for the PDB searches.

2.2. Back-End

The main purpose of the back-end is to enable the access to the data from the MySQL database. This application component converts data from SQL queries into JSON objects. Those objects can be viewed and managed by a web client, which in this case implements the front-end. The back-end has a RESTful API which enables the submission or modification of data stored in the MySQL database.

One view for each database table was built with the same name of the respective table. Some of those views use serializers that retrieve data from multiple tables related between themselves by foreign keys. Whenever a query request is made, the returned data will be returned as response (also in a JSON format). All these pages will allow to receive requests to insert, update or delete data entries. Besides that, all of those will contain features to allow to filter and/or search data, besides containing pagination methods.

Since bioinformatics packages are commonly written in Python, those tools are integrated within this application layer, that later will be accessed by the front-end by requests.

Clustal Omega analysis was wrapped in our application using a Clustal console (36). This view will receive the sequences and the additional parameters as part of the request. As result, it performs a Clustal alignment which provides the sequence alignment output as a text response.

To build Weblogos, the packages *weblogo*[1] and *logomaker* (37) are used.

The ML view allows to perform the FPep prediction for a given sequence using models trained with the in-house *Propythia* package (38). This package enables the development of ML and deep learning models for classification of peptides. It allows to obtain protein descriptors from their sequences
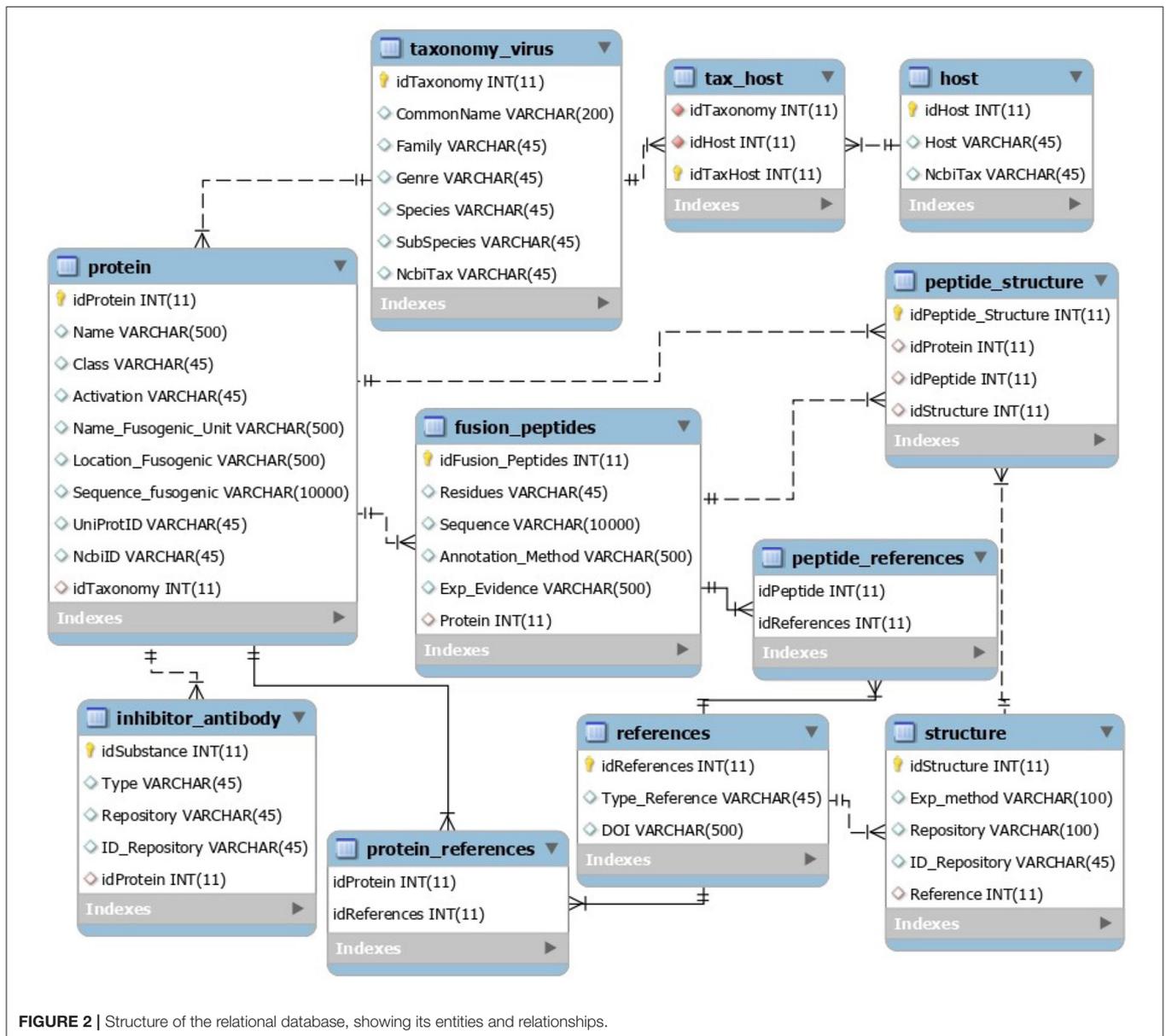
---

[1]https://weblogo.readthedocs.io/

FIGURE 2 | Structure of the relational database, showing its entities and relationships.

such as physicochemical features, and includes methods for data preprocessing, feature selection, dimensionality reduction, clustering and pipelines to train, evaluate and optimize ML and deep learning models (38). The models used in this application were trained from a dataset generated by extracting the physicochemical features using the Descriptor module of Propythia from a dataset where the negative instances are composed of protein sequences from transmembrane domains and random sequences extracted from VFP sequences after excluding the FPep. The group of descriptors that the Descriptors module provide include physicochemical features calculated from the peptide's sequence (e.g., length, charge, charge density, number of C, H, N, O, and S atoms within the amino acids, molecular weight, gravy, aromacity, isoelectric point, instability index, secondary structure, molar

extinction coefficient, flexibility, aliphatic index, Boman index and hydrophobic index); its amino acid and pseudo-amino acid compositions (including dipeptide and tripeptide), CTD (Composition, Transition and Distribution) information, Conjoint Triad and sequence order, as well additional base class peptide features. These descriptors are obtained from functions available in the packages *PyDPI* (39), *Biopython* (34), *modlAMP* (40), and *pfeature* (41). Calculated features went through a wrapper feature selection process using a Support Vector Machine (SVM) model, which had the best performance in this dataset. From these features, the ones that were considered as the most relevant include tripeptide and dipeptide composition, secondary structure, autocorrelation and CTD descriptors. Finally, ML models were trained with the Propythia's *train_best_model* function from the

shallow machine learning class. Those models are stored in Pickle files[2].

If a certain model is selected, the file containing that model will be loaded and used in an instance of the *Propythia*'s ML class. This ML view also requires the selection of the window size and a size for the gap. After defining those parameters, the *predicted_window* function of the Propythia is executed to perform the likelihood prediction of the subpeptide present in a given window to be a FPeps. Those probabilities are returned in a JSON object by the view.

2.3. Front-End

The front-end was built using a framework called Nebular[3], which is an open-source Angular User Interface Library that consists in a pre-built modular, customizable and configurable Angular tool. This framework enables the creation of Angular-based web applications and speeds up the development process of a front-end by providing several useful user-friendly and visually appealing web components, like tables and card formats.

2.3.1. Tables' Pages

The information of the database is found on the "Tables & Data" option of the front-end menu. This menu option contains a list of links for the `Taxonomy`, `Fusion Protein` and `Fusion Peptides` tables. All tables are organized in a Tree Grid format: each data entry will correspond to a table line; upon clicking on one row, its related data, which in this app corresponds to links to other tables connected by foreign keys, will appear below.

2.3.2. Sequence Alignment Pages

The application's tools can be grouped together in two categories: sequence alignment and sequence prediction. This section will cover the first group, while the next section handles the latter.

In the Sequence Alignment tools, a page that enables the submission of sequences to the NCBI BLAST was added. This page encompasses parameters that allow the selection of the type of BLAST and the database which BLAST will run against (27). Similarly, the HMMER page allows the selection of a sequence and the database to perform the alignment in the EMBL-EBI portal, showing the results in their web portal (42).

The Weblogo page contains two possible ways to build sequence logos. The first uses the Weblogo 3 portal[4] (43), in which the page will ask a minimum of three sequences, which must be in a FASTA format. Those sequences are submitted to the back-end to perform a Clustal Omega alignment and from that the web portal builds a weblogo provided in PNG or raw data formats. The second option to generate weblogos consists in the usage of the integrated sequence logo generator of the back-end. The PNG output will appear in the same page, on which the user can select the type of coloring provided by the *logomaker* package, as well as the number of stacks per line; the text output will appear in a pop-up page.

___
[2]The datasets, as well a small tutorial on how the models were trained, can be seen in https://github.com/pdMM11/Dockers/tree/master/Django_/crmapp/ml_models/DATASETS

[3]https://akveo.github.io/nebular/

[4]http://weblogo.threeplusone.com/create.cgi

The Clustal page alignment will be run in the back-end, in a command line. The user can select parameters such as the format of output. Upon submitting the request, one pop-up will open in the browser to display the alignment in the required format; a second one can appear if the guide tree data was requested.

2.3.3. Sequence Prediction Pages

The web application contains two tools relative to sequence prediction: epitope prediction and FPep prediction.

Regarding the epitope prediction (epitopes are binding sites that antibody molecules can attach to) the site will access to the IEDB portal (44) to obtain an antibody epitope prediction. The page will send a request with the sequence, selected analysis method and window size to the back-end that will redirect to the IEDB API. The results will be displayed in a Smart Table format. Both Bepipred methods' results will appear into 2 different result tables, the rest of them just require one (45).

Regarding ML-based prediction of the location of FPeps within the VFP, this page allows to define a VFP target sequence, as well as specifying which model to use, from the set of available models trained in Propythia: Support Vector Machine, Random Forest, Gradient Boosting, K-Nearest Neighbors, Linear Regression, Gaussian Nave Bayes, and Artificial Neural Network models. Finally, the user can select the window and gap sizes. The page can display those predictions in two types of output: a table which contains every possible peptide probability of being a FPep; and a graphical interface that, for each amino acid, displays a color that represents the maximum probability of all peptides that contain that particular character of being a FPep. Moreover, for each amino acid, a popup box will show up, informing the user about the position within the VFP sequence and the probability value of belonging to a FPep. These results can be downloaded into a text file.

3. RESULTS

There are two key components of the front-end that can require an explanation of their elements: the tables and the tools.

Figure 3 shows the representation of the `Fusion Protein` page. This image contains an example of a table component and will be used to explain all table elements present in our web application. The list above corresponds to the description of the table functionalities from the left to the right and top to down:

- On the dashed green rectangle on the left (number 1), there is the side menu of the application, where the user can access the three main data tables, as well as the bioinformatics tools present in the server.
- On the top right, in the black box (number 2), the user can find the link to the global search tool; upon entering a search term, a pop-up window will be displayed if the main pages contain results.
- The red rectangle (number 3) contains the search methods previously described, that will determine what is shown in the table; the search box contains an auto-complete system with term suggestions that is executed after the input of 3 characters.
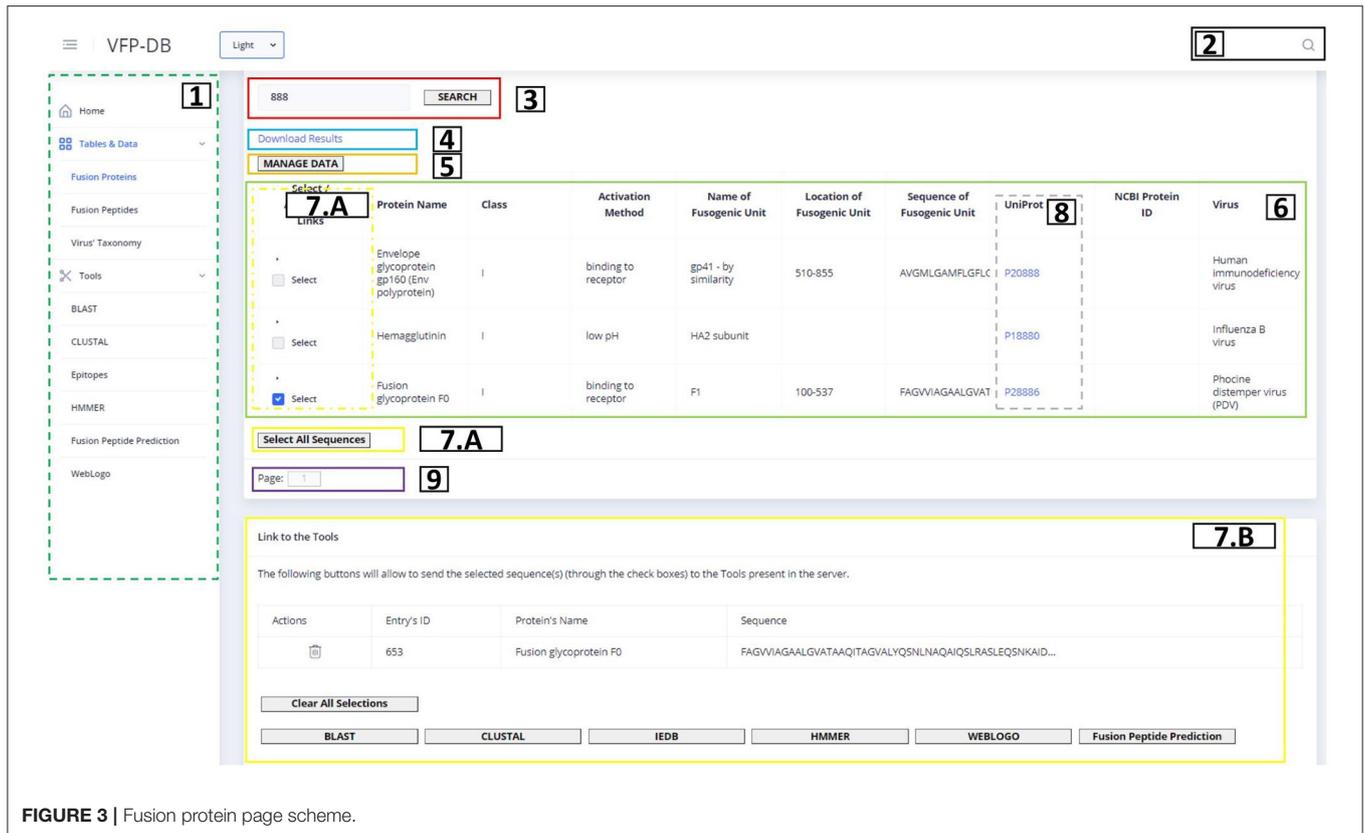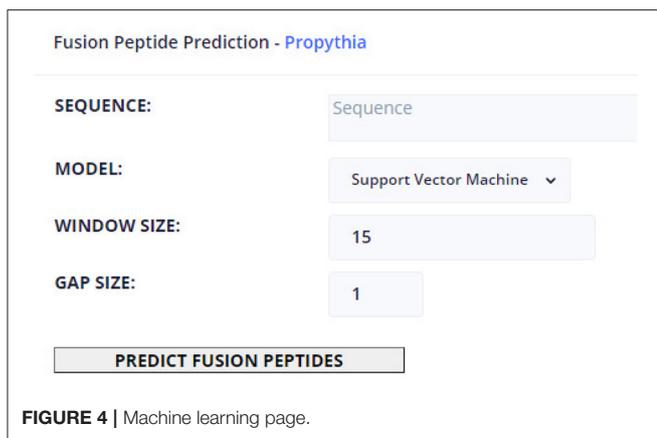
FIGURE 3 | Fusion protein page scheme.

FIGURE 4 | Machine learning page.

- The blue rectangle (number 4) allows to download the query result into a CSV file.
- The orange rectangle (number 5) redirects to the back-end's administrator page.
- The green rectangle (number 6) is the table with results from the query request determined by the search term and the current page; within each line, there are links to related data (if available).
- The yellow rectangles (numbers 7.A and 7.B) show the selection of sequences that will be provided to the web

application tools, chosen in the checkboxes on the yellow dashed rectangle, or through the "Select All Sequences" button; upon choosing at least one sequence, the rectangle visible at 7.B will show up, allowing the user to delete some (in the table) or all (in the "Clear All Sequences" option) of the selected sequences and redirect the chosen sequences to the Tools, accessible by the buttons on the bottom of the page.

- Some columns like the one inside the dashed gray rectangle (number 8) will contain links to redirect to the original source of the data; the links can be UniProt, NCBI Protein, PDB or NCBI Taxonomy data entries.
- The purple box buttons (number 9) correspond to the pagination methods that will also determine the content of the table.

Figure 4 shows an example of how all tool pages are structured. Regarding the available tools, they contain a text box to insert the sequence(s) input, which can be auto-filled if protein sequences were selected in the Fusion Protein page: some tools (e.g., the multiple alignment tools) require a minimum of 3 protein sequences in a FASTA format; however, the most basic input of the text box is a single protein sequence. To instruct the user about the type of input that each tool requires, the text boxes contain placeholders with a small description of expected input. After the text box is filled, the user can set custom values for each of the rest of the parameters. The results of the tools can appear in new tabs with text results (e.g., using the Clustal or Weblogo), or to access other web portals (e.g., BLAST or HMMER). The results
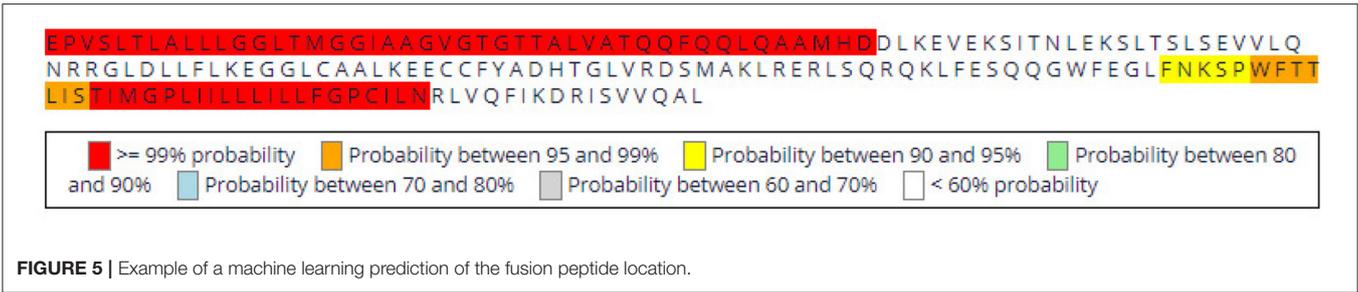
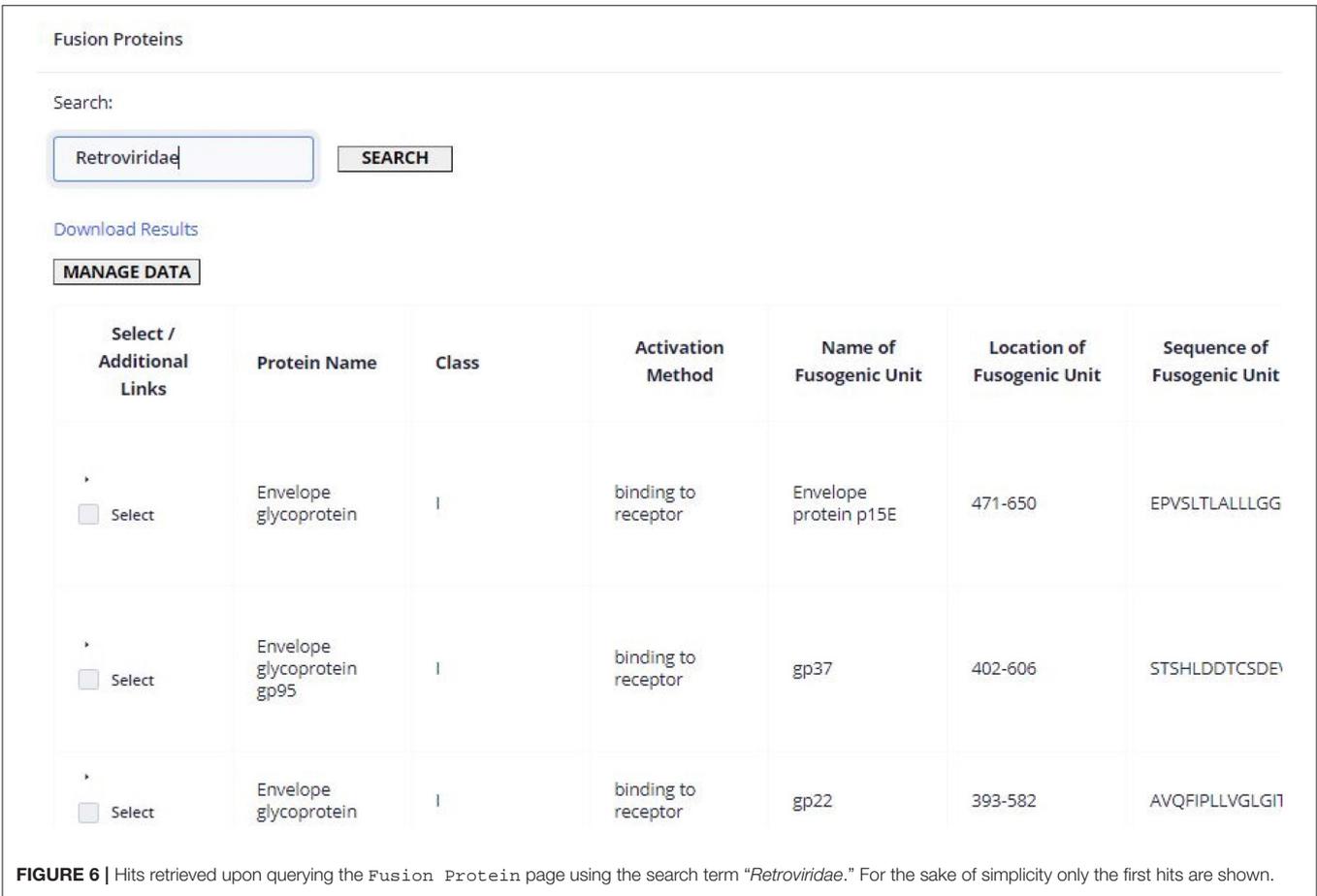FIGURE 5 | Example of a machine learning prediction of the fusion peptide location.

FIGURE 6 | Hits retrieved upon querying the `Fusion Protein` page using the search term "*Retroviridae*." For the sake of simplicity only the first hits are shown.

can also appear within the same page, like in the case of ML prediction (as shown in **Figure 5** and the image result of Weblogo built from the back-end's functions.

3.1. Use Case-Analysis of the *Retroviridae* Conservation

To illustrate the application of ViralFP, we describe the case of a researcher that aims to obtain a general perspective of the conservation of the VFP of a given viral family, such as *Retroviridae*, to determine which regions could be interesting universal therapeutic targets. This can be done by accessing the `Fusion Protein` page, searching the "*Retroviridae*" term (**Figure 6**), selecting all the sequences,

and sending them to the Weblogo tool page through the corresponding option.

The sequences sent are already in the required FASTA format. If the user wants to obtain the graphical option of the weblogo, he/she can select the appropriate coloring to explore the conservation (in the case of **Figure 7** the selected coloring is based on the amino acid physical-chemical properties) and the number of characters per line (default: 25).

After running the analysis, the output is shown below (**Figure 8**). Using this representation, the regions with high conservation can be easily located, since they correspond to positions with larger stacks. In those, the conserved region found around positions 15 (first line)

FIGURE 7 | View of the weblogo page with the *Retroviridae* fusion protein sequences provided as input.
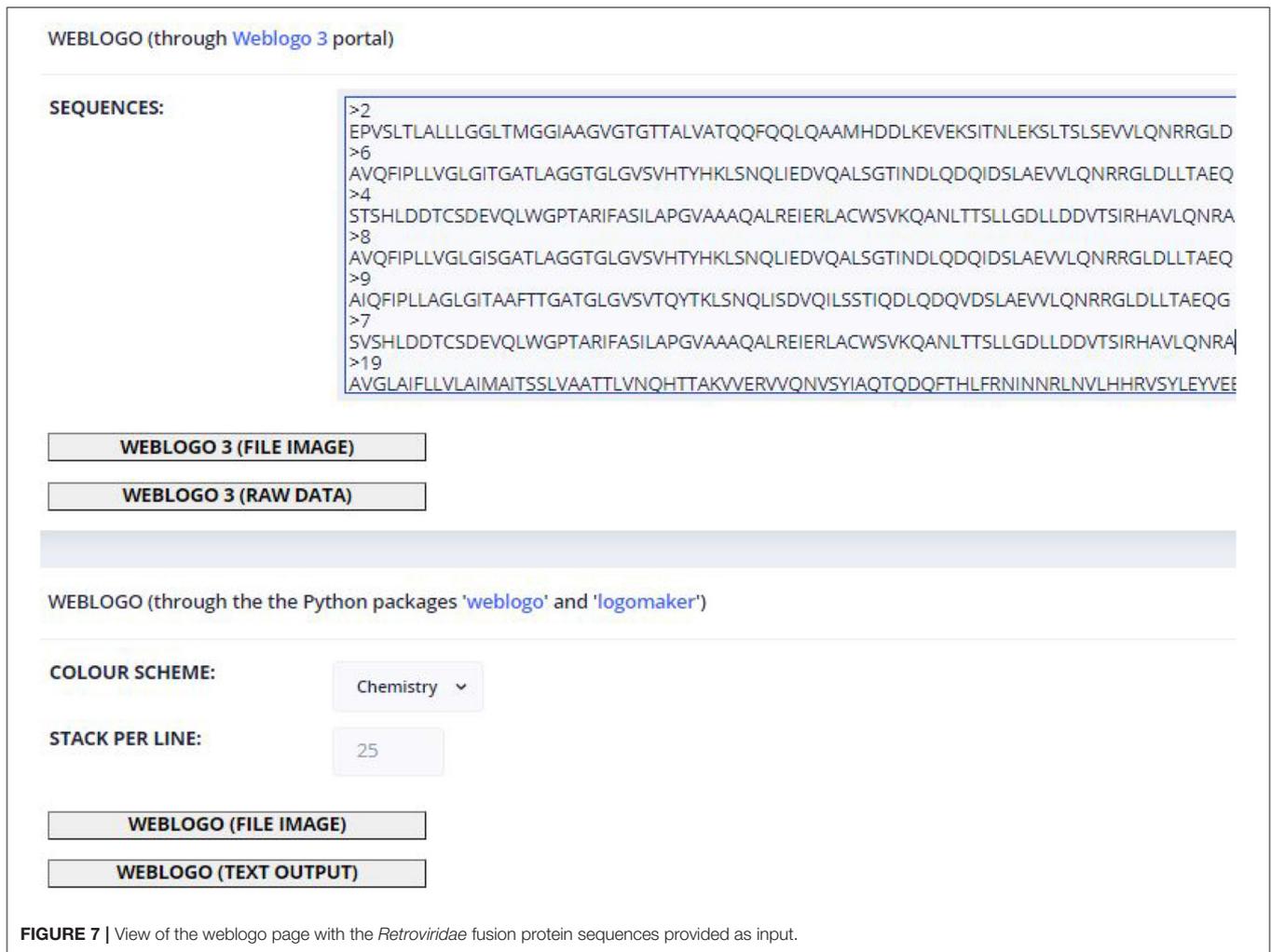
through 60 (third line) corresponds to the FPep, a very conserved region.

4. DISCUSSION

This work provides a useful resource for the scientific community working with viruses and their proteins, since it presents a tool that can be used by any researcher interested in obtaining several types of information regarding VFPs.

Currently, the application implements several useful functionalities: it provides an easy access to the MySQL database containing VFP information through a user-friendly web interface. The proposed bioinformatics tools are implemented in the application, through web interfaces. ML models capable of predicting FPeps within the VFP sequence are also available. Finally, a full manual curation of the database information was performed.

Future tasks within the website will include improving the tool by improving current features and adding new ones, including the possibility to define more parameters within the application's tools, improving the initial loading time of the website and the app's aesthetic appearance.

Alongside these tasks, to improve the utility of this application for the scientific community, we will try implement tools capable of predicting the most conserved regions within proteins, which will allow to complement the ML output, due to FPeps tending to be conserved regions. ML models will be improved as new data regarding FPeps is discovered and added to the database, allowing us to update and expand its training dataset.

Regarding the database's information, it is imperative to add all the relevant new entries published after the submission of this manuscript, hopefully by implementing methodologies capable of performing this type of search automatically and periodically. We are going to add methods that allow users to insert new data, as well as update some inconsistencies that they might find in the application. We aim to add

FIGURE 8 | Weblogo analysis of the fusion proteins from the *Retroviridae* family.

information regarding receptors of VFPs, as well as each VFP's known cleavage sites. Finally, it is important to redo the automated searches of known inhibitors/antibodies within Pubmed and PDB in a way that can take into consideration the virus' strain.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: https://github.com/pdMM11/Dockers/tree/master/Django_/crmapp/ml_models/DATASETS.

AUTHOR CONTRIBUTIONS

DL, MR, and PM conceptualized the web application and wrote the manuscript. PM was the main developer of the web application, by implementing all the different components that allow the functioning of the front-end, back-end and integration of bioinformatics tools, procedures described in the Materials and Methods chapter. SP was responsible for gathering the information present in the protein repositories that later was inserted into the web application database. RR coordinated the development of the web application and deployed it into the server. AS developed *Propythia* and the ML pipelines used to train the models present in the application. DL, MR, and RR supervised the work. All authors revised the manuscript.

FUNDING

ACKNOWLEDGMENTS

REFERENCES

1. Marsh M. *Current Topics in Microbiology and Immunology. Vol. 6*. Berlin: Springer. (2005).
2. Marsh M, Helenius A. Virus entry: open sesame. *Cell*. (2006) 124:729–40. doi: 10.1016/j.cell.2006.02.007
3. Weissenhorn W, Hinz A, Gaudin Y. Virus membrane fusion. *FEBS Lett*. (2007) 581:2150–5. doi: 10.1016/j.febslet.2007.01.093
4. Apellániz B, Huarte N, Largo E, Nieva JL. The three lives of viral fusion peptides. *Chem Phys Lipids*. (2014) 181:40–55. doi: 10.1016/j.chemphyslip.2014.03.003
5. Harrison SC. Viral membrane fusion. *Virology*. (2015) 479-480:498–507. doi: 10.1016/j.virol.2015.03.043
6. Baquero E, Albertini AA, Vachette P, Lepault J, Bressanelli S, Gaudin Y. Intermediate conformations during viral fusion glycoprotein structural transition. *Curr Opin Virol*. (2013) 3:143–50. doi: 10.1016/j.coviro.2013.03.006
7. White JM, Delos SE, Brecher M, Schornberg K. Structures and mechanisms of viral membrane fusion proteins: multiple variations on a common theme. *Crit Rev Biochem Mol Biol*. (2008) 43:189–219. doi: 10.1080/10409230802058320
8. Hughson FM. Structural characterization of viral fusion proteins. *Curr Biol*. (1995) 5:265–74. doi: 10.1016/S0960-9822(95)00057-1
9. Schneck E, Sedlmeier F, Netz RR. Hydration repulsion between biomembranes results from an interplay of dehydration and depolarization. *Proc Natl Acad Sci USA*. (2012) 109:14405–9. doi: 10.1073/pnas.1205811109
10. Teissier É, Pécheur EI. Lipids as modulators of membrane fusion mediated by viral fusion proteins. *Eur Biophys J*. (2007) 36:887–99. doi: 10.1007/s00249-007-0201-z
11. Epand RM. Fusion peptides and the mechanism of viral fusion. *Biochim Biophys Acta Biomem*. (2003) 1614:116–21. doi: 10.1016/S0005-2736(03)00169-X
12. Han X, Bushweller JH, Cafiso DS, Tamm LK. Membrane structure and fusion-triggering conformational change of the fusion domain from influenza hemagglutinin. *Nat Struct Biol*. (2001) 8:715–20. doi: 10.1038/90434
13. Lorieau JL, Louis JM, Bax A. The complete influenza hemagglutinin fusion domain adopts a tight helical hairpin arrangement at the lipid:water interface. *Proc Natl Acad Sci USA*. (2010) 107:11341–11346. doi: 10.1073/pnas.1006142107
14. Haque ME, Koppaka V, Axelsen PH, Lentz BR. Properties and structures of the influenza and hiv fusion peptides on lipid membranes: implications for a role in fusion. *Biophys J*. (2005) 89:3183–94. doi: 10.1529/biophysj.105.063032
15. Lai AL, Moorthy AE, Li Y, Tamm LK. Fusion activity of hiv gp41 fusion domain is related to its secondary structure and depth of membrane insertion in a cholesterol-dependent fashion. *J Mol Biol*. (2012) 418:3–15. doi: 10.1016/j.jmb.2012.02.010
16. Melo MN, Sousa FJR, Carneiro FA, Castanho MARB, Valente AP, Almeida FCL, et al. Interaction of the dengue virus fusion peptide with membranes assessed by nmr: the essential role of the envelope protein Trp101 for membrane fusion. *J Mol Biol*. (2009) 392:736–46. doi: 10.1016/j.jmb.2009.07.035
17. Li J, Zhan P, Liu X. Targeting the entry step of SARS-CoV-2: a promising therapeutic approach. *Signal Transduct Targeted Ther*. (2020) 5:1–2. doi: 10.1038/s41392-020-0195-x
18. Jackson LA, Anderson EJ, Rouphael NG, Roberts PC, Makhene M, Coler RN, et al. An mRNA vaccine against SARS-CoV-2-preliminary report. *N Engl J Med*. (2020) 383:1920–31. doi: 10.1056/NEJMoa2022483
19. Polack FP, Thomas SJ, Kitchin N, Absalon J, Gurtman A, Lockhart S, et al. Safety and efficacy of the BNT162b2 mRNA Covid-19 vaccine. *N Engl J Med*. (2020) 383:2603–15. doi: 10.1056/NEJMoa2034577
20. Deng YQ, Dai JX, Ji GH, Jiang T, Wang HJ, Yang Ho, et al. A broadly flavivirus cross-neutralizing monoclonal antibody that recognizes a novel epitope within the fusion loop of E protein. *PLoS ONE*. (2011) 6:e16059. doi: 10.1371/journal.pone.0016059
21. Kam YW, Lee CYP, Teo TH, Howland SW, Amrun SN, Lum FM, et al. Cross-reactive dengue human monoclonal antibody prevents severe pathologies and death from Zika virus infections. *JCI Insight*. (2017) 2:e92428. doi: 10.1172/jci.insight.92428
22. Kong R, Xu K, Zhou T, Acharya P, Lemmin T, Liu K, et al. Fusion peptide of HIV-1 as a site of vulnerability to neutralizing antibody. *Science*. (2016) 352:828–33. doi: 10.1126/science.aae0474
23. Kallewaard NL, Corti D, Collins PJ, Neu U, McAuliffe JM, Benjamin E, et al. Structure and function analysis of an antibody recognizing all influenza a subtypes. *Cell*. (2016) 166:596–608. doi: 10.1016/j.cell.2016.05.073
24. The UniProt Consortium. UniProt: the universal protein knowledgebase in 2021. *Nucleic Acids Res*. (2020) 49:D480–9. doi: 10.1093/nar/gkaa1100
25. Geer LY, Marchler-Bauer A, Geer RC, Han L, He J, He S, et al. The NCBI BioSystems database. *Nucleic Acids Res*. (2009) 38:D492–6. doi: 10.1093/nar/gkp858
26. Burley SK, Berman HM, Bhikadiya C, Bi C, Chen L, Di Costanzo L, et al. RCSB protein data bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology and energy. *Nucleic Acids Res*. (2019) 47:D464–74. doi: 10.1093/nar/gky1004
27. Johnson M, Zaretskaya I, Raytselis Y, Merezhuk Y, McGinnis S, Madden TL. NCBI BLAST: a better web interface. *Nucleic Acids Res*. (2008) 36:W5–9. doi: 10.1093/nar/gkn201

28. Sievers F, Higgins DG. Clustal omega. *Curr Protoc Bioinform*. (2014) 48:3–13. doi: 10.1002/0471250953.bi0313s48

29. Finn RD, Clements J, Eddy SR. HMMER web server: interactive sequence similarity searching. *Nucleic Acids Res*. (2011) 39:W29–37. doi: 10.1093/nar/gkr367

30. Wu S, Han J, Liu R, Liu J, Lv H. A computational model for predicting fusion peptide of retroviruses. *Comput Biol Chem*. (2016) 61:245–50. doi: 10.1016/j.compbiolchem.2016.02.013

31. Wu S, Wu X, Tian J, Zhou X, Huang L. PredictFP2: a new computational model to predict fusion peptide domain in all retroviruses. *IEEE/ACM Trans Comput Biol Bioinform*. (2019) 17: 1714–20. doi: 10.1109/TCBB.2019.2898943

32. Xu Y, Yu S, Zou JW, Hu G, Rahman NABD, Othman RB, et al. Identification of peptide inhibitors of enveloped viruses using support vector machine. *PLoS ONE*. (2015) 10:1–15. doi: 10.1371/journal.pone.0144171

33. Federhen S. The NCBI Taxonomy database. *Nucleic Acids Res*. (2011) 40:D136–43. doi: 10.1093/nar/gkr1178

34. Cock PJA, Antao T, Chang JT, Chapman BA, Cox CJ, Dalke A, et al. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*. (2009) 25:1422–3. doi: 10.1093/bioinformatics/btp163

35. Gilpin W. PyPDB: a Python API for the protein data bank. *Bioinformatics*. (2015) 32:159–60. doi: 10.1093/bioinformatics/btv543

36. Sievers F, Wilm A, Dineen D, Gibson TJ, Karplus K, Li W, et al. Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. *Mol Syst Biol*. (2011) 7:539. doi: 10.1038/msb.2011.75

37. Tareen A, Kinney JB. Logomaker: beautiful sequence logos in Python. *Bioinformatics*. (2019) 36:2272–74. doi: 10.1093/bioinformatics/btz921

38. Sequeira AM, Lousa D, Rocha M. ProPythia: a python automated platform for the classification of proteins using machine learning. In: *Practical Applications of Computational Biology & Bioinformatics, 14th International Conference (PACBB 2020)*. Cham: Springer. (2021). p. 32–41.

39. Cao DS, Liang YZ, Yan J, Tan GS, Xu QS, Liu S. PyDPI: freely available python package for chemoinformatics, bioinformatics, and chemogenomics studies. *J Chem Inf Model*. (2013) 53:3086–96. doi: 10.1021/ci400127q

40. Müller AT, Gabernet G, Hiss JA, Schneider G. modlAMP: Python for antimicrobial peptides. *Bioinformatics*. (2017) 33:2753–5. doi: 10.1093/bioinformatics/btx285

41. Pande A, Patiyal S, Lathwal A, Arora C, Kaur D, Dhall A, et al. Computing wide range of protein/peptide features from their sequence and structure. *bioRxiv preprint*. (2019) doi: 10.1101/599126

42. Potter SC, Luciani A, Eddy SR, Park Y, Lopez R, Finn RD. HMMER web server: 2018 update. *Nucleic Acids Res*. (2018) 46:W200–4. doi: 10.1093/nar/gky448

43. Crooks G, Hon G, Chandonia J, Brenner S. WebLogo: a sequence logo generator. *Genome Res*. (2004) 14:1188–90. doi: 10.1101/gr.849004

44. Vita R, Overton JA, Greenbaum JA, Ponomarenko J, Clark JD, Cantrell JR, et al. The immune epitope database (IEDB) 3.0. *Nucleic Acids Res*. (2014) 43:D405–D412. doi: 10.1093/nar/gku938

45. Vita R, Mahajan S, Overton JA, Dhanda SK, Martini S, Cantrell JR, et al. The immune epitope database (IEDB):2018 update. *Nucleic Acids Res*. (2018) 47:D339–43. doi: 10.1093/nar/gky1006

46. Moreira P, Sequeira AM, Pereira S, Rodrigues R, Rocha M, Lousa D. ViralFP: a webserver of viral fusion proteins. *arXiv preprint arXiv:210314754*. (2021). Available online at: https://arxiv.org/abs/2103.14754