DS569k Protein Embeddings Dataset

Donald Bertucci and Alex Endert

Georgia Institute of Technology

Abstract: This is a quick report on embeddings computed for 569 thousand proteins from the UniProtKB Swiss-Prot protein dataset. Anyone can easily download the pre-computed 128D ProteinCLIP embeddings for similarity search, visualization, down-stream tasks, and more! You can access the processed DS569k dataset easily through https://huggingface.co/datasets/donnyb/DS569k. We conclude by showing a couple use cases of DS569k: 2D projected scatterplot and similarity search interface with filtering.

1 Introduction

Neural network embeddings are very powerful at representing patterns in data in a fixed-dimension vector. By converting protein sequences into embeddings, people can leverage existing machine learning algorithms that require fixed-dimension vectors (similarity search, clustering, projection, and more).

In this paper, we've taken a powerful model ProteinCLIP [7] that produces protein embeddings in a joint embedding space with functional descriptions to embed the *Reviewed (Swiss-Prot)* Dataset[1] from UniprotKB [2]. We also add NCBI Taxonomy [6] information for filtering. The dataset is called DS569k.

We contribute the data in parquet format that can be used in a couple lines of code (see Figure 1). We hope the ease of access can foster new interfaces, reapplication to existing machine learning algorithms, and analysis of known and unknown proteins.

```python
import pandas as pd
DS596k = pd.read_parquet("https://huggingface.co/datasets/donnyb/DS569k/resolve/main/DS569k.parquet")
```

accession	embedding	protein_name	organism_name	ncbi_taxonomy_class	ncbi_taxonomy_phylum	gene_name	sequence_length
string · lengths	sequence · lengths	string · lengths	string · lengths	string · classes	string · classes	string · lengths	int64
6 10	128 128	2 149	7 183	263 values	117 values	1 43 ø	2 3k
Q6GZX4	[0.04449855908751488, -0.0262824147939682,…	Putative transcription…	Frog virus 3 (isolate Goorha)	Megaviricetes	Nucleocytoviricota	FV3-001R	256
Q6GZX3	[0.020507629960775375, 0.07163890451192856,…	Uncharacterized protein 002L	Frog virus 3 (isolate Goorha)	Megaviricetes	Nucleocytoviricota	FV3-002L	320
Q197F8	[-0.108150415122509, -0.09237941354513168,…	Uncharacterized protein 002R	Invertebrate iridescent virus 3	Megaviricetes	Nucleocytoviricota	IIV3-002R	458
Q197F7	[0.06144670024514198, 0.04602198302745819,…	Uncharacterized protein 003L	Invertebrate iridescent virus 3	Megaviricetes	Nucleocytoviricota	IIV3-003L	156
Q6GZX2	[-0.0005804159445688128,…	Uncharacterized protein 3R	Frog virus 3 (isolate Goorha)	Megaviricetes	Nucleocytoviricota	FV3-003R	438
Q6GZX1	[0.006679930724203587, 0.14680644869804382,…	Uncharacterized protein 004R	Frog virus 3 (isolate Goorha)	Megaviricetes	Nucleocytoviricota	FV3-004R	60
Q197F5	[0.011756743304431438, 0.10397190600633621,…	Uncharacterized protein 005L	Invertebrate iridescent virus 3	Megaviricetes	Nucleocytoviricota	IIV3-005L	217

Figure 1: In a couple lines of code, you can fetch the dataframe which contains the entire DS569k dataset. The first seven rows above were screenshot from https://huggingface.co/datasets/donnyb/DS569k.

[1] https://www.uniprot.org/help/downloads

2 Usage

You can either download the data, then import wherever you'd like, or you can directly read like shown in Figure 1. The download link is: https://huggingface.co/datasets/donnyb/DS569k/resolve/main/DS569k.parquet. We recommend using the Pandas library for the ease of use in Python (see Figure 1). Alternative huggingface data download options are shown on the main dataset page: https://huggingface.co/datasets/donnyb/DS569k.

If you're using Pandas, you can then access the embeddings easily by simply indexing the column as `DS569k["embedding"]`. Please see all the columns at https://huggingface.co/datasets/donnyb/DS569k.

3 Specification

The entire dataset in the parquet file is 301 Megabytes and 569,192 rows. Each row represents a UniprotKB protein from the *Reviewed (Swiss-Prot)* Dataset[2] [2]. We filtered out proteins greater than 3000 residues long to reduce computational costs.

For each protein, we computed the ProteinCLIP [7] projected ESM2 [4] model embedding. To reduce computation, we used the smallest ESM2 model, the `esm2_t6_8M_UR50D` and the corresponding ProteinCLIP projection. Specifically we took the last layers embeddings from `esm2_t6_8M_UR50D`, normalized the results with the 2-norm, then projected it with the `proteinclip_esm2_6.onnx` model. The result is a 128-dimensional vector per protein.

We heavily reused the excellent ProteinCLIP [7] code[3] to easily use the `esm2_t6_8M_UR50D` and `proteinclip_esm2_6.onnx` models. All the credit goes to them [7]. If you want to embed more proteins, use their repository or alternatively we distilled some of their code in one file[4] for ease of use.

For the other columns, we parsed the description from UniprotKB [2] into the UniProt `accession`, `protein_name`, `organism_name`, `ncbi_taxonomy_class`, `ncbi_taxonomy_phylum`, and more. We used the https://github.com/zyxue/ncbitax2lin [8] library to grab the NCBI Taxonomy information. All credit goes to UniprotKB [2] for the data.

4 Examples

To show how simple `DS569k` is to use, we created a couple examples.

First, we simply uploaded a 250k sample to the Nomic Atlas service. They take embeddings and reduce them down to 2D points for visualization. They also layer on topic modeling to label regions of text. See Figure 2. The result is a map that we can explore the space of known proteins with labeled regions. Zoom in and interact for yourself on Nomic: https://atlas.nomic.ai/data/donnybertucci/lackadaisical-goodfellow/map/519a3223-f6cd-4ff0-add6-28de33c0be37.

Second, we created a similar protein search web interface. In the interface, we have a text box for an input query and display cosine similarity between the top similar proteins among the entire `DS569k`. In Figure 3, we query with a protein with unknown function found in the Ganaspis hookeri wasp venom [1, 3, 5] downloaded from https://venome.cqls.oregonstate.edu/protein/Gh_comp2027_c0_seq1 against other `DS569k` proteins in the Insecta taxonomic class. The code for the viewer in Figure 3 is at https://github.com/xnought/DS569k-viewer. Hopefully finding structurally similar proteins embedded with similar functions will reveal possible functions of the query protein.

[2] https://www.uniprot.org/help/downloads
[3] https://github.com/wukevin/proteinclip
[4] https://github.com/xnought/DS569k-viewer/blob/main/viewer/server/src/embed_proteinclip.py#L107

5 Conclusion

We provide a simple way to use pre-computed embeddings in the `DS569k` dataset at https://huggingface.co/datasets/donnyb/DS569k. These embeddings can be used very easily to create interfaces or do protein analysis within Python Pandas or any other dataframe library.

Acknowledgements

Thank you to Dr. Nathan Mortimer and Michael Youkhateh for the feedback to add NCBI Taxonomic data to the dataset as a helpful field to filter by. The `DS569k` viewer example was also built with them in mind.

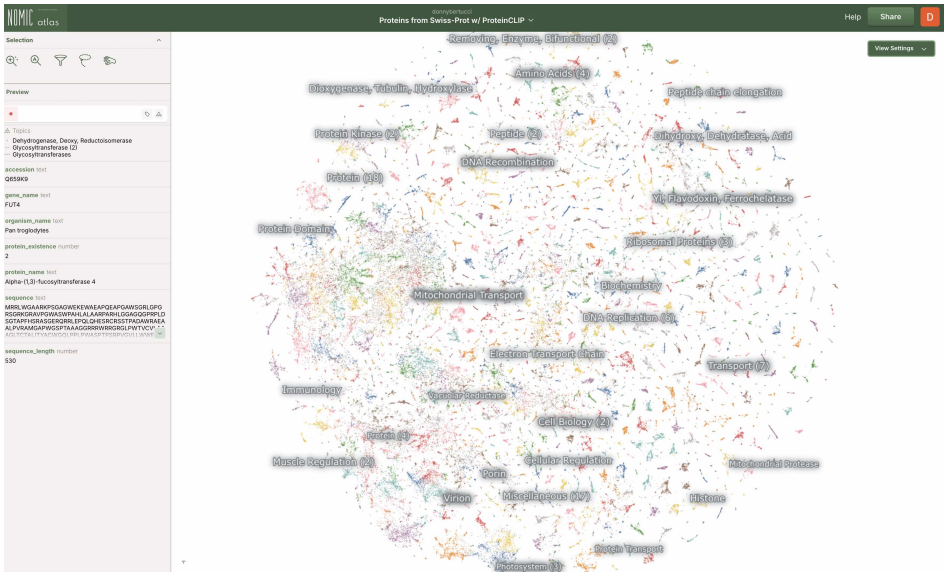

Figure 2: Each protein is a point on the scatterplot and groups represent similar proteins.

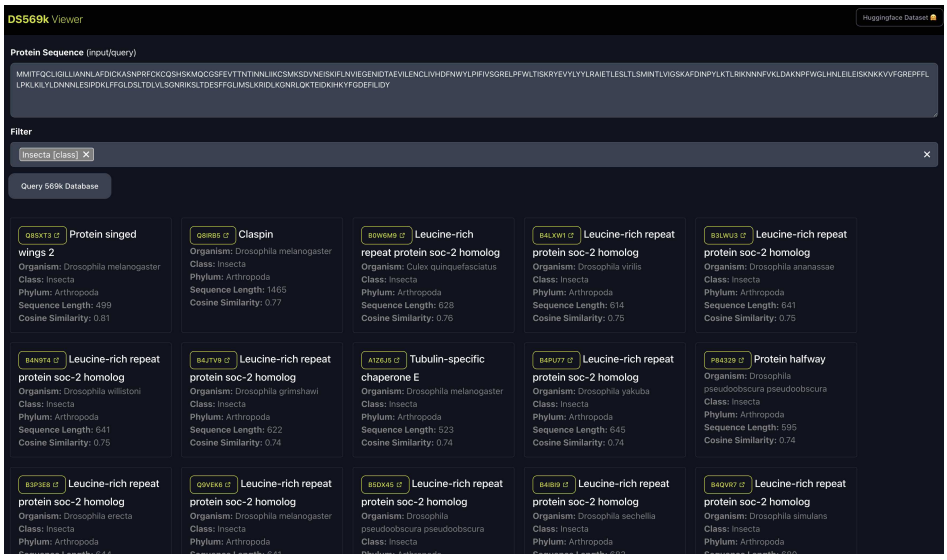

Figure 3: A query interface to find similar proteins within `DS569k`.

References

[1] Gloria Alvarado et al. "Bioinformatic analysis suggests potential mechanisms underlying parasitoid venom evolution and function". In: *Genomics* 112.2 (2020), pp. 1096–1104.

[2] The UniProt Consortium. "UniProt: the Universal Protein Knowledgebase in 2023". In: *Nucleic Acids Research* 51.D1 (Nov. 2022), pp. D523–D531. ISSN: 0305-1048.

[3] Jeremy Goecks et al. "Integrative approach reveals composition of endoparasitoid wasp venoms". In: *PloS one* 8.5 (2013), e64125.

[4] Zeming Lin et al. "Evolutionary-scale prediction of atomic-level protein structure with a language model". In: *Science* 379.6637 (2023), pp. 1123–1130.

[5] Nathan T Mortimer et al. "Parasitoid wasp venom SERCA regulates Drosophila calcium levels and inhibits cellular immunity". In: *Proceedings of the National Academy of Sciences* 110.23 (2013), pp. 9427–9432.

[6] Conrad L Schoch et al. "NCBI Taxonomy: a comprehensive update on curation, resources and tools". In: *Database* 2020 (2020), baaa062.

[7] Kevin E Wu, Howard Chang, and James Zou. "ProteinCLIP: enhancing protein language models with natural language". In: *bioRxiv* (2024), pp. 2024–05.

[8] Zhuyi Xue. *ncbi2taxlin*. 2017. URL: https://github.com/zyxue/ncbitax2lin.