CS612 Proposed Project – Classifying Proteins Using Neural Networks and Protein Embeddings

Description

The overall goal is to classify proteins using neural networks. Protein sequences, being represented as strings of letters, cannot be fed to neural networks as-is. Therefore, they have to be converted into arrays or matrices of numbers. A simple way to represent sequences is one-hot encoding - representing every amino acid with bit vectors where every bit is 0 except one 1. There are several other ways, see here for a comprehensive survey but not very successful example:

`https://medium.com/@rwalroth89/neural-nets-for-protein-classification-1b122cde5b6f`

In recent years, many large language models (LLM) have been used for protein sequence embedding. ProtBert is based on Bert model which pretrained on a large corpus of protein sequences in a self-supervised fashion. This means it was pretrained on the raw protein sequences only, with no humans labeling them in any way (which is why it can use lots of publicly available data) with an automatic process to generate inputs and labels from those protein sequences.

One important difference between our Bert model and the original Bert version is the way of dealing with sequences as separate documents. This means the Next sentence prediction is not used, as each sequence is treated as a complete document. The masking follows the original Bert training with randomly masks 15% of the amino acids in the input.

The model could be used for protein feature extraction or to be fine-tuned on downstream tasks.

See more detailed description here: `https://huggingface.co/Rostlab/prot_bert`

For example, ProtBert-BFD is a ProtBert based model, good for secondary structure, localization and prediction of membrane proteins. See here:

`https://dataloop.ai/library/model/rostlab_prot_bert_bfd/`

Prot-t5-xl-uniref50, good for feature extraction and secondary structure prediction:

`https://dataloop.ai/library/model/rostlab_prot_t5_xl_uniref50/`

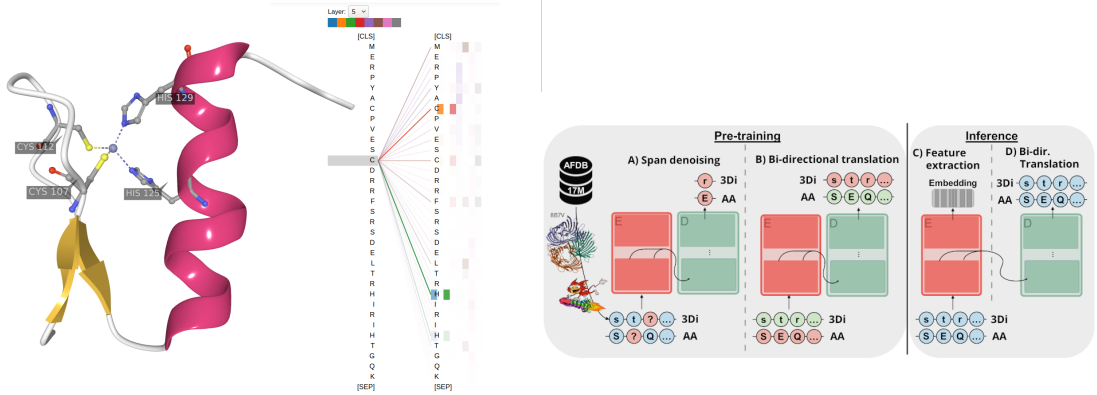
And many others. Check `https://huggingface.co/Rostlab/`.

Maybe more importantly, these models can create embeddings, which capture the structural and functional properties of the proteins. See here:

`https://www.uniprot.org/help/embeddings`

Here is a very good example of a tool that provides a large number of Bert or Bert-like models to create embeddings and predictions. I highly recommend using it for this project. The code is available on github:

`github.com/agemagician/ProtTrans/tree/master`



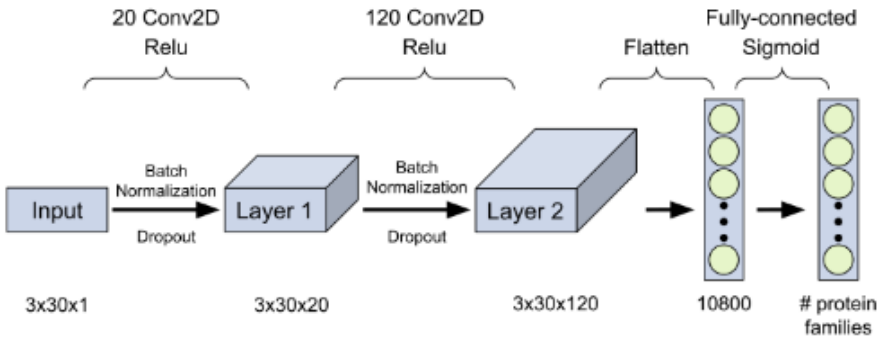
Neural Networks

Protein sequence data can be used to solve complex bioinformatics classification problems. We mentioned neural networks in class, albeit briefly. Nowadays there are many types of neural networks. Simple(ish) examples include for example – a 1D or 2D CNN (Convolutional neural network). A CNN has one or more convolutional layers, optionally pooling layers to reduce the dimension, fully connected layers and backpropagation for learning and training. Here is a rather simple example that does not work so well but is the scope of what I expect:

`https://stephanheijl.com/protein_sequence_ml.html`

Here is another tutorial that uses transfer learning and pre-trained models (a bit much for this kind of project)

`https://medium.com/@rwalroth89/neural-nets-for-protein-classification-1b122cde5b6f`



Many implementations of CNN exist using keras, tensorflow, pytorch etc. see here:

`https://www.tensorflow.org/tutorials/images/cnn`

`https://www.kaggle.com/code/kanncaa1/convolutional-neural-network-cnn-tutorial`

`https://www.analyticsvidhya.com/blog/2021/08/beginners-guide-to-convolutional-neural-network-`

`https://pyimagesearch.com/2021/07/19/pytorch-training-your-first-convolutional-neural-network`

`https://medium.com/@daniel.schuetzler/neural-networks-in-python-complete-guide-mastering-ai-n`

Please use an existing code as basis and don't re-invent the wheel.

https://academic.oup.com/bioinformatics/article/33/22/3685/4092933
https://github.com/rvinas/predicting-subcellular-location

The Assignment

Use a neural networks to classify proteins based on sequence information and probably other data.

- Input: Protein sequences - examples will be provided.
- Come up with an embedding (use a Bert based model, not one-hot encoding for this project. It's too easy.)
- Build a neural network of your choice, or more than one.
- Select a task to predict. Above examples include, for example, secondary structures, cellular localization, structural features, protein families etc.). Feel free to get ideas from there. Their "predict" module may be an overkill, though, and I will not accept using their code as-is anyway.

  You should discuss:

  1. What did you try to predict
  2. What architecture you used
  3. Other considerations - programming languages, packages, tested data etc.

The not-so-successful example above discusses representations of sequences as input features, most of them rather simple.

`https://medium.com/@rwalroth89/neural-nets-for-protein-classification-1b122cde5b6f`

An interesting test would be to test the features as given in the link above vs. autoencoders or embeddings and test the difference in performance, and/or use different architectures.

Deliverables

- The code.
- A 3-4 page document outlining the following:
  - A brief introduction to the problem (a paragraph or so).
  - Implementation details: Programming language, choice of representation of proteins, network architecture - layers, activation methods, dimensions.
  - Results: Reconstruction error, example of at least one MSA. You can use the plotting functions presented in the links above.
  - The Neural network parameters, choice of architecture, classification ability.