Protein Cellular Component Ontology Prediction

Data Challenge 2023

Waël Doulazmi, Ambroise Odonnat, Roman Plaud

Master MVA, ENS Paris-Saclay

name.surname@ens-paris-saclay.fr

Advanced Learning for Text and Graph Data

January 27, 2023

Overview

Overview

Cellular Component Ontology Prediction

Classification task

18 classes of Cellular Component Ontology

Multiple Representations

- Sequences of amino acids
- Graphs $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
  - ⋆ nodes $\mathcal{V} \rightarrow$ amino acids
  - ⋆ edges $\mathcal{E} \rightarrow$ based on distance and chemical properties

Protein Data

Sequence Representation

A-Y-I-A-K-Q-R-Q-I-S-F-V-K-S-H-F-S-R-Q-L-E-E-R-L-G-L-S-R-V-G-D-G-T-Q-D-N-L-S-G-A-E-K-A-V-Q-V-K-V-K-A-L-P-D-A-Q-F-E

Figure 1: A sequence of amino acids

Graph Representation

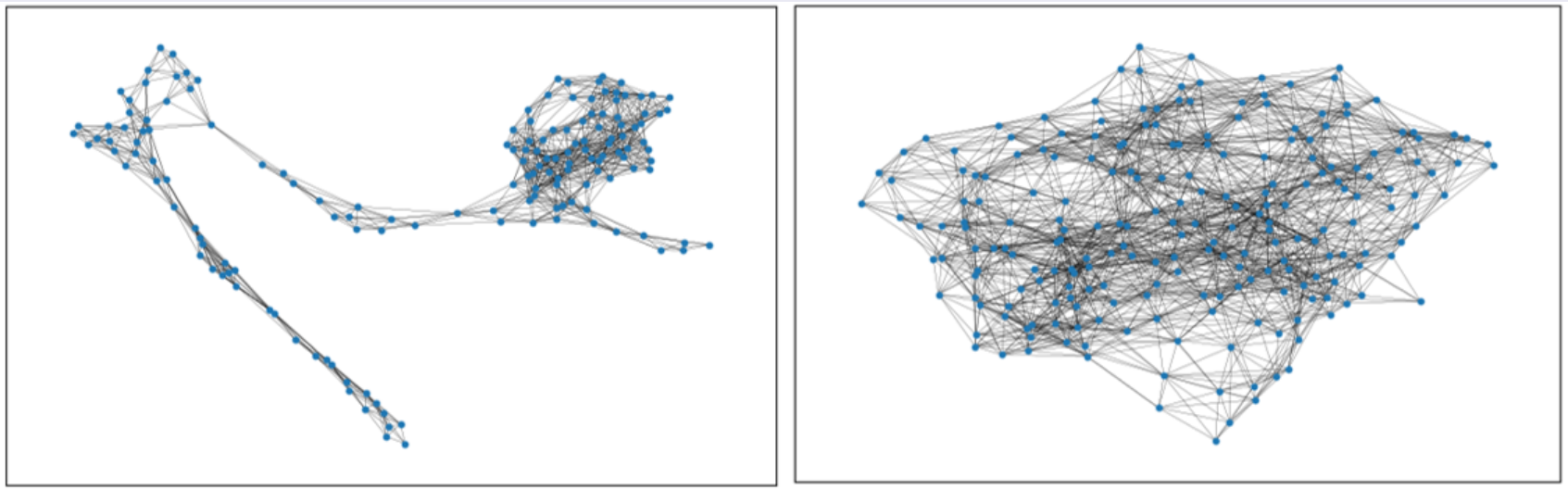

Figure 2: Graph representation of 2 proteins from class 8

Dataset

Data	Labels	# train set	# test set
6111 proteins	18 classes	4888	1223

Table 1: Dataset description

Split	$\#\lvert\mathcal{G}\rvert$	Avg. $\lvert\mathcal{V}\rvert$	Avg. $\lvert\mathcal{E}\rvert$
Train	4888	258	4486
Test	1223	254	4379

Table 2: Dataset statistics

Unbalanced Classification

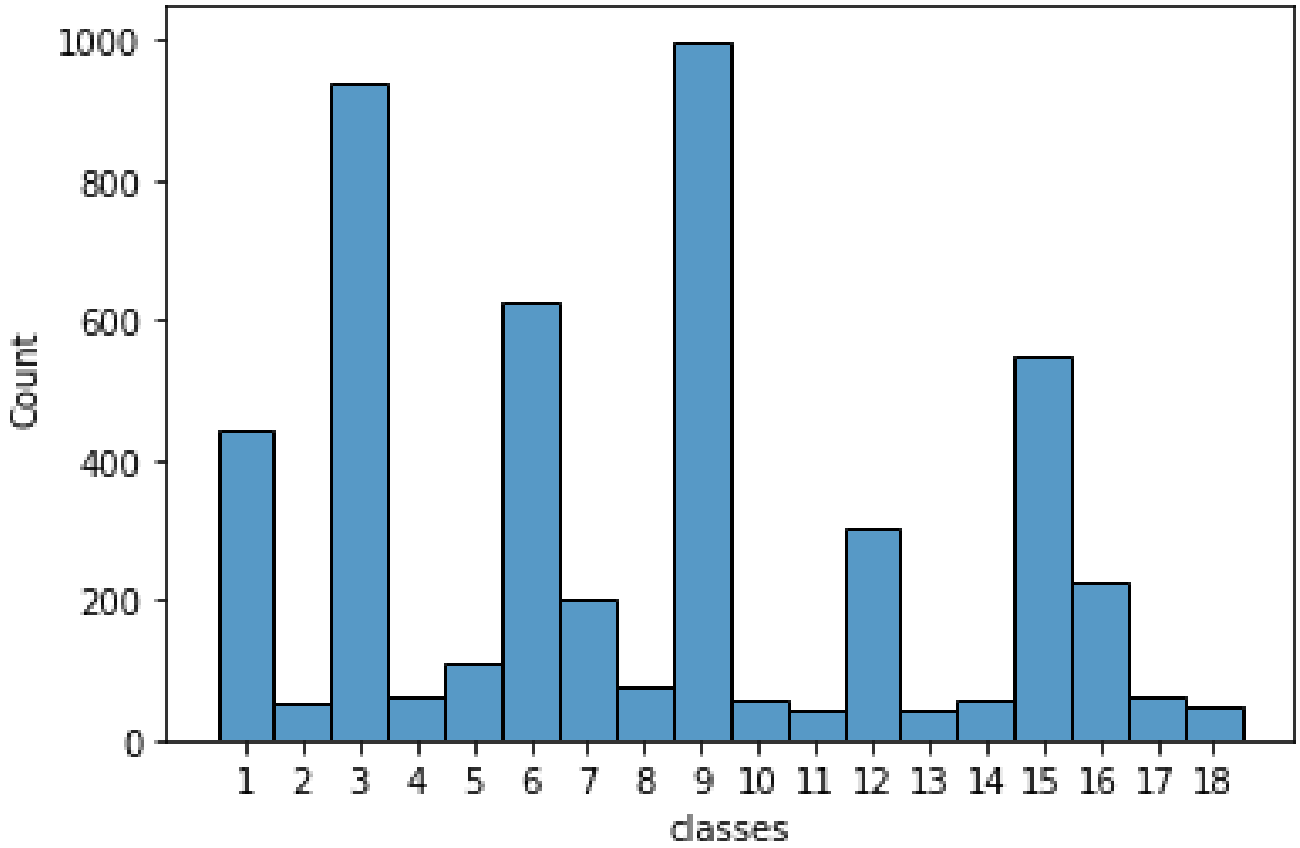

Figure 3: Repartition of data in training set

Weighted Loss for Training

- Minority classes seen less often than majority classes
- Idea $\rightarrow$ give more impact on the loss to minority classes
- Weighted PyTorch implementation of negative log-likelihood

Choice of weight

For the class i of cardinal N_i, $\quad w_i = \frac{1}{\sqrt{N_i}}$

Overview

Edge Features

5 attributes associated to each edge:

- Distance between two connected nodes (amino acids)
- Membership binary variable for each of the 4 types of edges:
  a. distance-based edge $\rightarrow$ 99.91 % of edges
  b. peptide bond edge $\rightarrow$ 10.80 % of edges
  c. k-NN edge $\rightarrow$ 0 % of edges
  d. hydrogen bond edge $\rightarrow$ 0.36 % of edges

Node Features

86 attributes associated to amino acids:

- 3D Coordinates (3)

Node Features

86 attributes associated to amino acids:

- 3D Coordinates (3)
- One-hot encoding of amino acid type (20)

Node Features

86 attributes associated to amino acids:

- 3D Coordinates (3)
- One-hot encoding of amino acid type (20)
- Hydrogen bond acceptor / donor status (2)

Node Features

86 attributes associated to amino acids:

- 3D Coordinates (3)
- One-hot encoding of amino acid type (20)
- Hydrogen bond acceptor / donor status (2)
- Chemical EXPASY features (61)

Node Features

86 attributes associated to amino acids:

- 3D Coordinates (3)
- One-hot encoding of amino acid type (20)
- Hydrogen bond acceptor / donor status (2)
- Chemical EXPASY features (61)

We ignore features that are redundant with edges.

Node Features - EXPASY

EXPASY

- **61** chemical properties of the amino acid in the protein
- Might be **redundant** with amino acid type
- Might be too **fine-grained** for our task

Node Features - EXPASY

EXPASY

- **61** chemical properties of the amino acid in the protein
- Might be **redundant** with amino acid type
- Might be too **fine-grained** for our task

$\longrightarrow$ No knowledge on biology, so we explore them with **PCA**

Node Features - EXPASY

PCA

- All features from all nodes $\rightarrow 1,572,264 \times 61$ data-frame
- Normalize all features to zero mean and unit variance
- Keep components explaining 80% of total variance

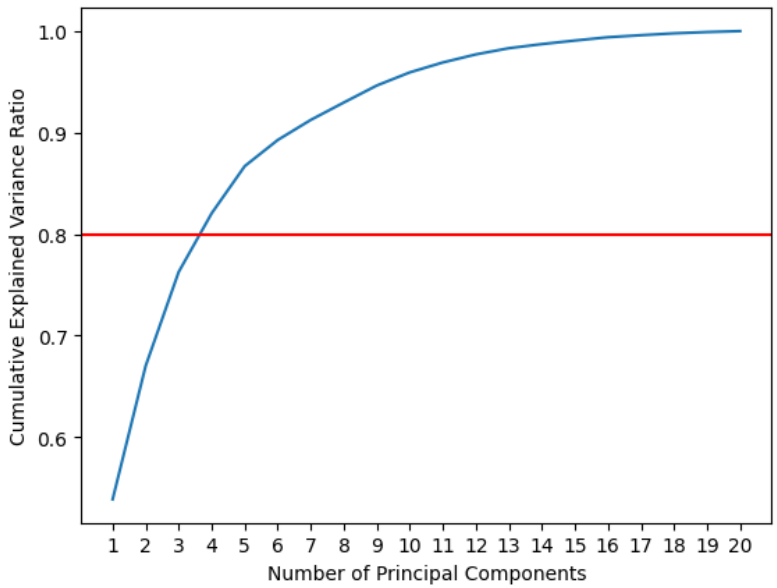

Figure 4: We keep the 4 first components

Node Features - EXPASY

PCA

- Keep **4 components** $\rightarrow$ compact information
- Project the features on the first 2 components
- All features seems to be of equal importance

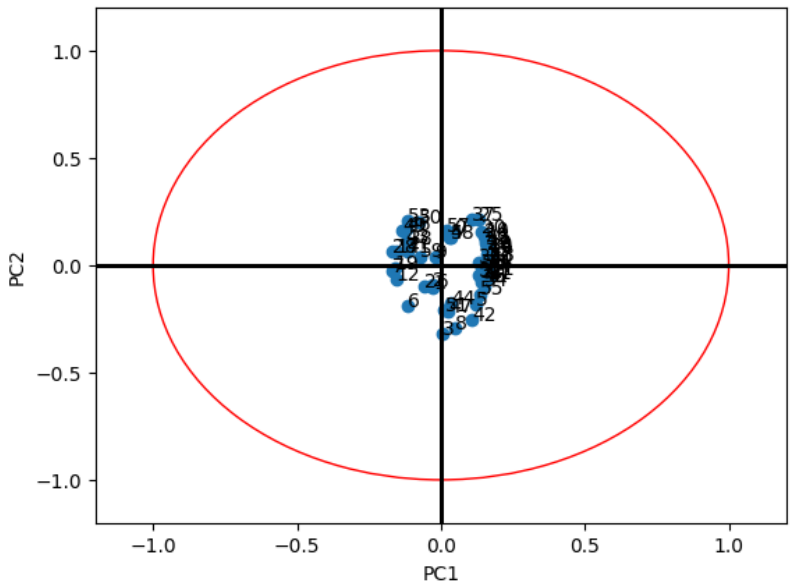

Figure 4: Circle of correlations

Node Features - Amino acid type

- **Salient** feature: different amino acids have different properties
- Influence at both **local** and **global** level

Node Features - Amino acid type

- **Salient** feature: different amino acids have different properties
- Influence at both **local** and **global** level

$\longrightarrow$ We put most of our efforts on these features

Node Features - Amino acid type

- **One-hot** encoding: inconvenient for Machine Learning
- Multiple options to get to a **dense** representation (SVD, . . .)
- We use tools from **NLP** (Word2Vec, BERT, . . .)

Overview

Protein Language

NLP for proteins

- Sequences on the vocabulary of amino acids
- Various studies show that NLP approach is relevant [Ofer et al., 2021]

Protein Language

NLP for proteins

- Sequences on the vocabulary of amino acids
- Various studies show that NLP approach is relevant [Ofer et al., 2021]

For our task

- Amino acids embeddings as node features
- Protein embeddings for classification / multi-modal models

NLP approaches

- Word2Vec $\longrightarrow$ amino acids embeddings
- TF-IDF, BoW, GoW $\longrightarrow$ protein sequence embeddings

NLP approaches

- Word2Vec $\longrightarrow$ amino acids embeddings
- TF-IDF, BoW, GoW $\longrightarrow$ protein sequence embeddings

Language models like BERT provides both !

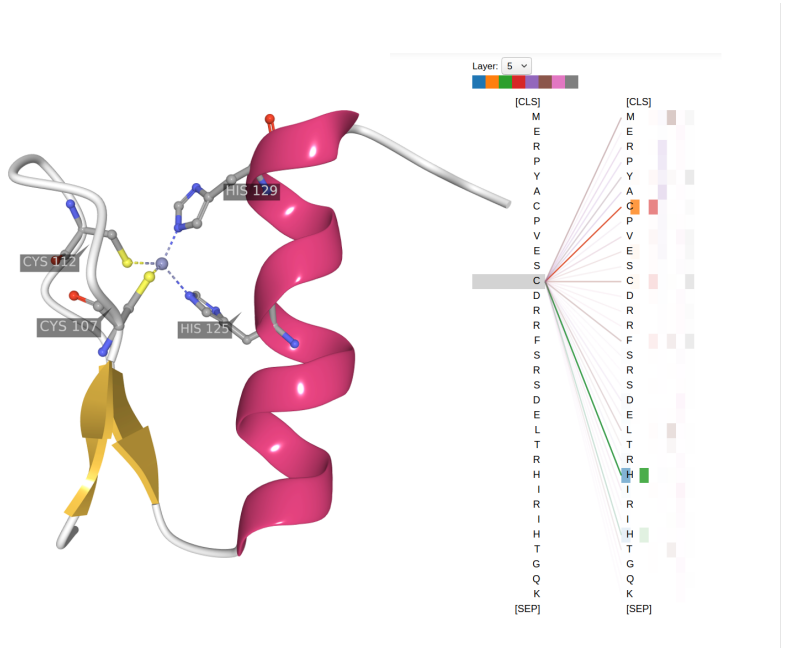

Figure 5: ProtBERT [Elnaggar et al., 2021]

NLP - Amino Acids embeddings

- ProtBERT was trained on UniRef100 (257 millions proteins)
  - ⋆ Masked Language Modelling
  - ⋆ Produce **contextual** embeddings
- Features for multi-modal GNNs, carry information on both:
  - ⋆ The amino acid
  - ⋆ Its role at protein level

$\longrightarrow$ No need to fine-tune it !

NLP - Protein embeddings

$\longrightarrow$ But are ProtBERT protein embeddings powerful enough to work only with sequences?

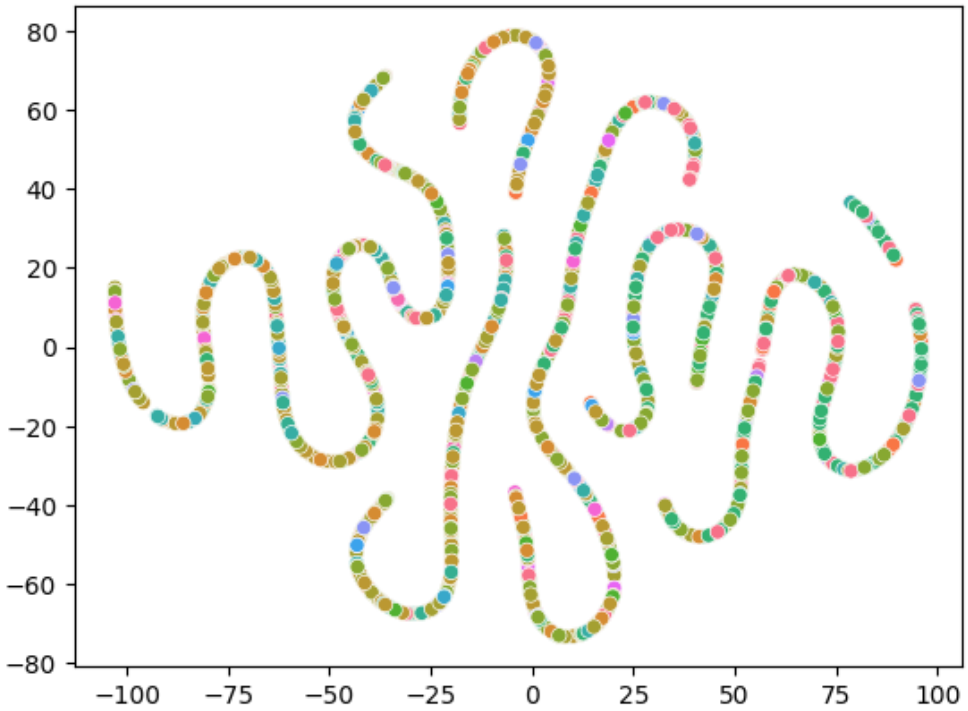

Figure 6: ProtBERT sequence embeddings, t-SNE visualization

$\longrightarrow$ **No !**

NLP - Protein embeddings

$\longrightarrow$ With 250 epochs of fine-tuning on our task:

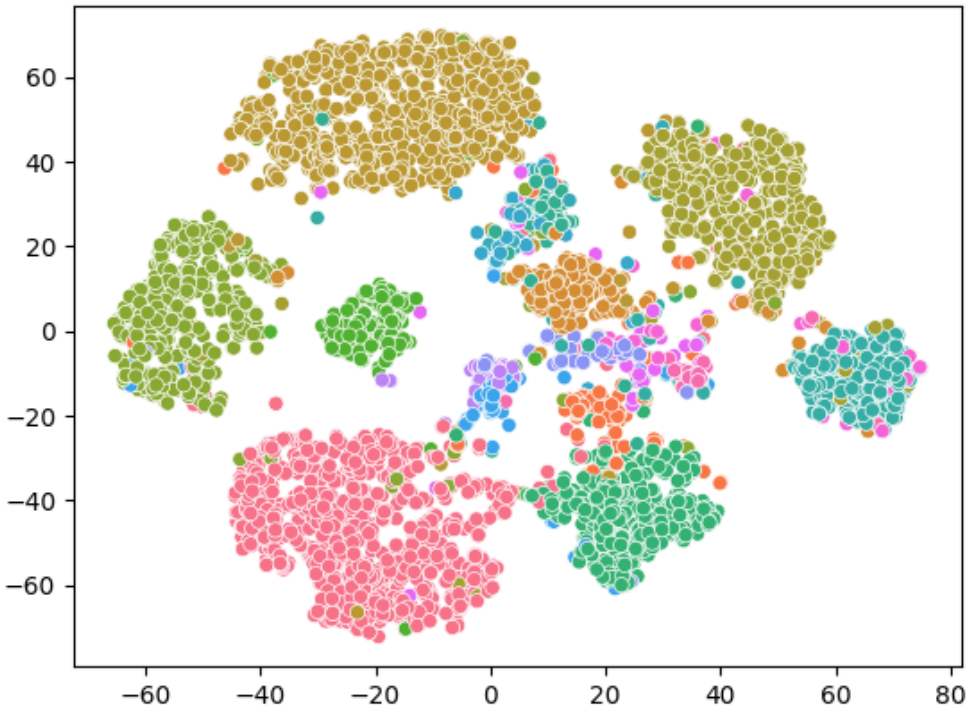

Figure 7: ProtBERT sequence embeddings, t-SNE visualization

$\longrightarrow$ Dominant classes start to cluster

NLP - Protein embeddings

$\longrightarrow$ With 250 epochs of fine-tuning on our task, and weighted cross-entropy:

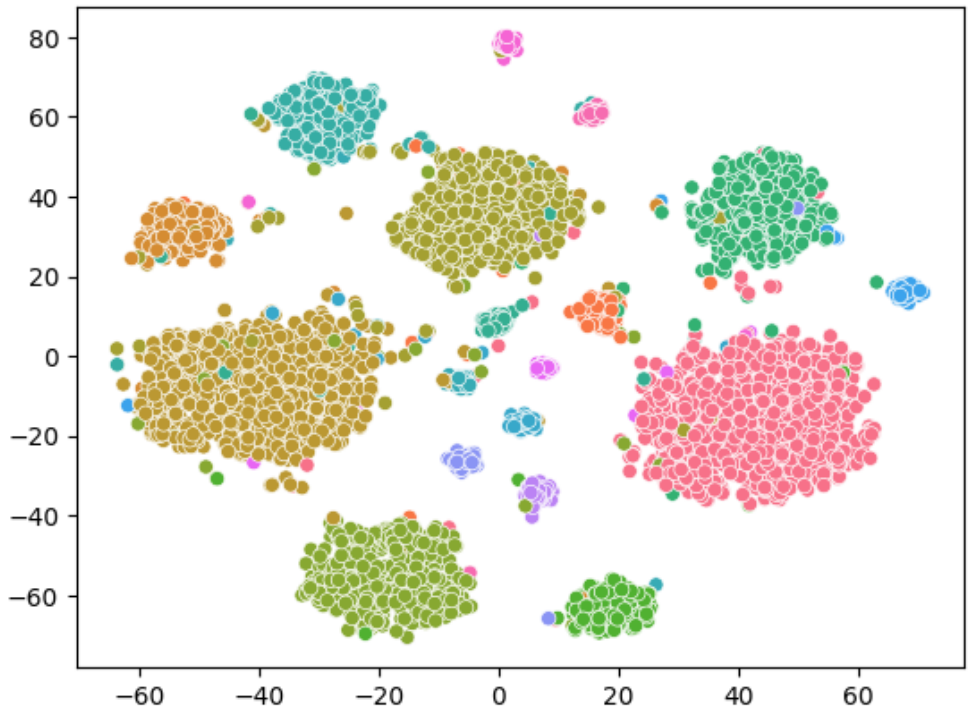

Figure 8: ProtBERT sequence embeddings, t-SNE visualization

$\longrightarrow$ Interesting embeddings !

NLP - Protein embeddings

$\longrightarrow$ With 250 epochs of fine-tuning on our task, and weighted cross-entropy:

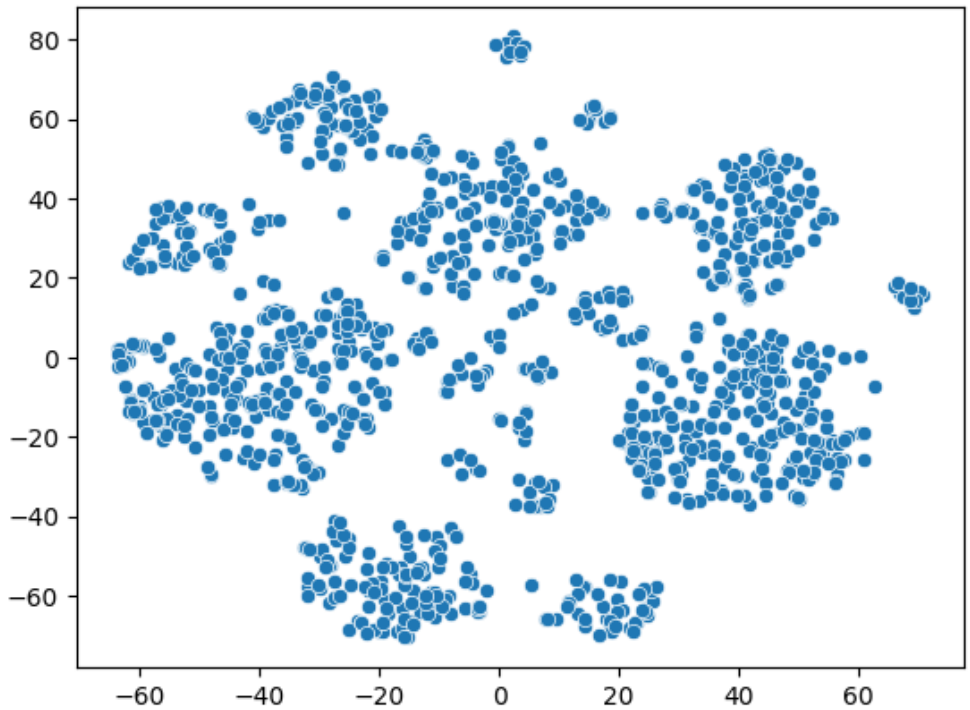

Figure 9: ProtBERT sequence embeddings, t-SNE visualization of test dataset

$\longrightarrow$ This structure is also present in test data !

NLP - Protein embeddings

$\longrightarrow$ Are we done ?

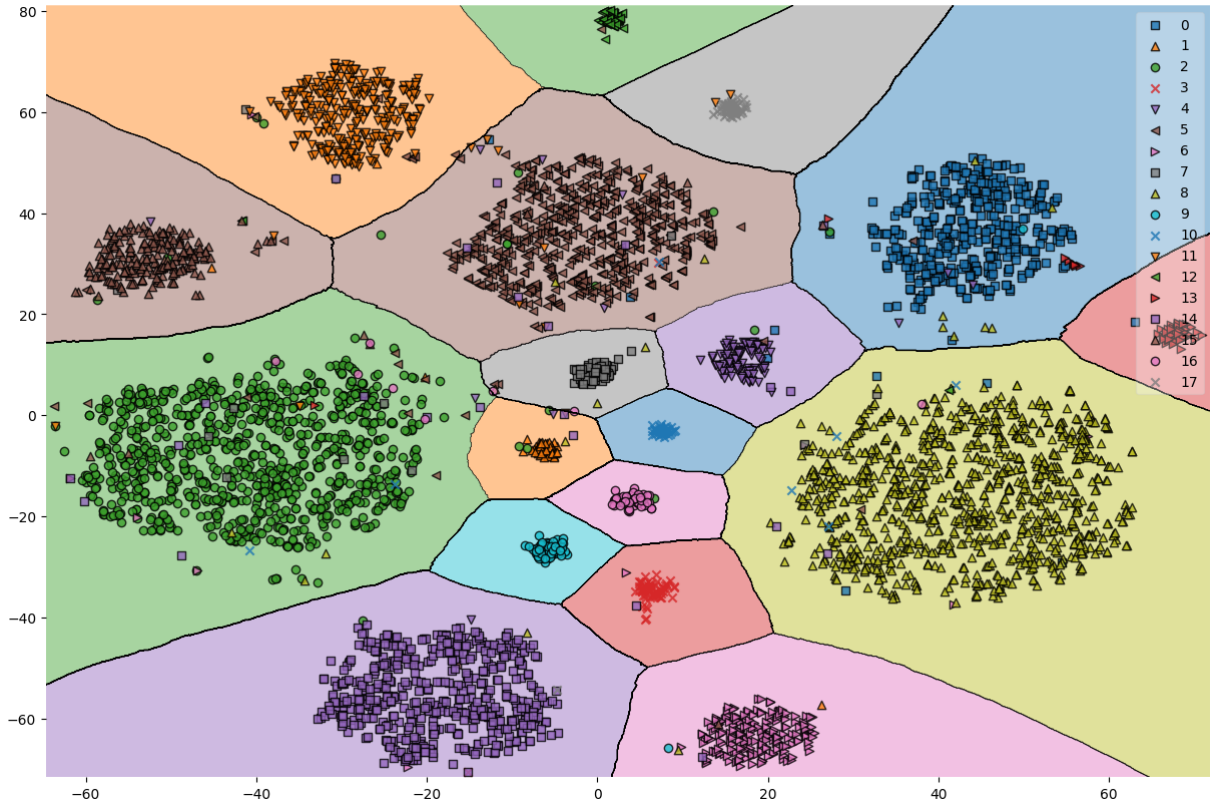

Figure 10: 40-NN classifier, on 2D protein sequence embeddings

$\longrightarrow$ Might be good for accuracy, terrible for loss !

NLP - Protein embeddings

$\longrightarrow$ Are we done ?

Classifiers on Protein embeddings

- Trained various classifiers: LogReg, SVM, MLP...
- Models tend to be very confident
- Errors are heavily penalized by the loss

$\longrightarrow$ Good starting point, but we can do better with structure!

Overview

Structure Modelling

Node features

- Original node attributes (86)
- BERT embeddings of amino-acids of (1024)

Structure Modelling

Node features

- Original node attributes (86)
- BERT embeddings of amino-acids of (1024)

Edge filtering

- Use all provided edges
- Use only distance-based edges
- Use only peptide bond edges
- A subset of edges based on their attributes.

Edge Filtering

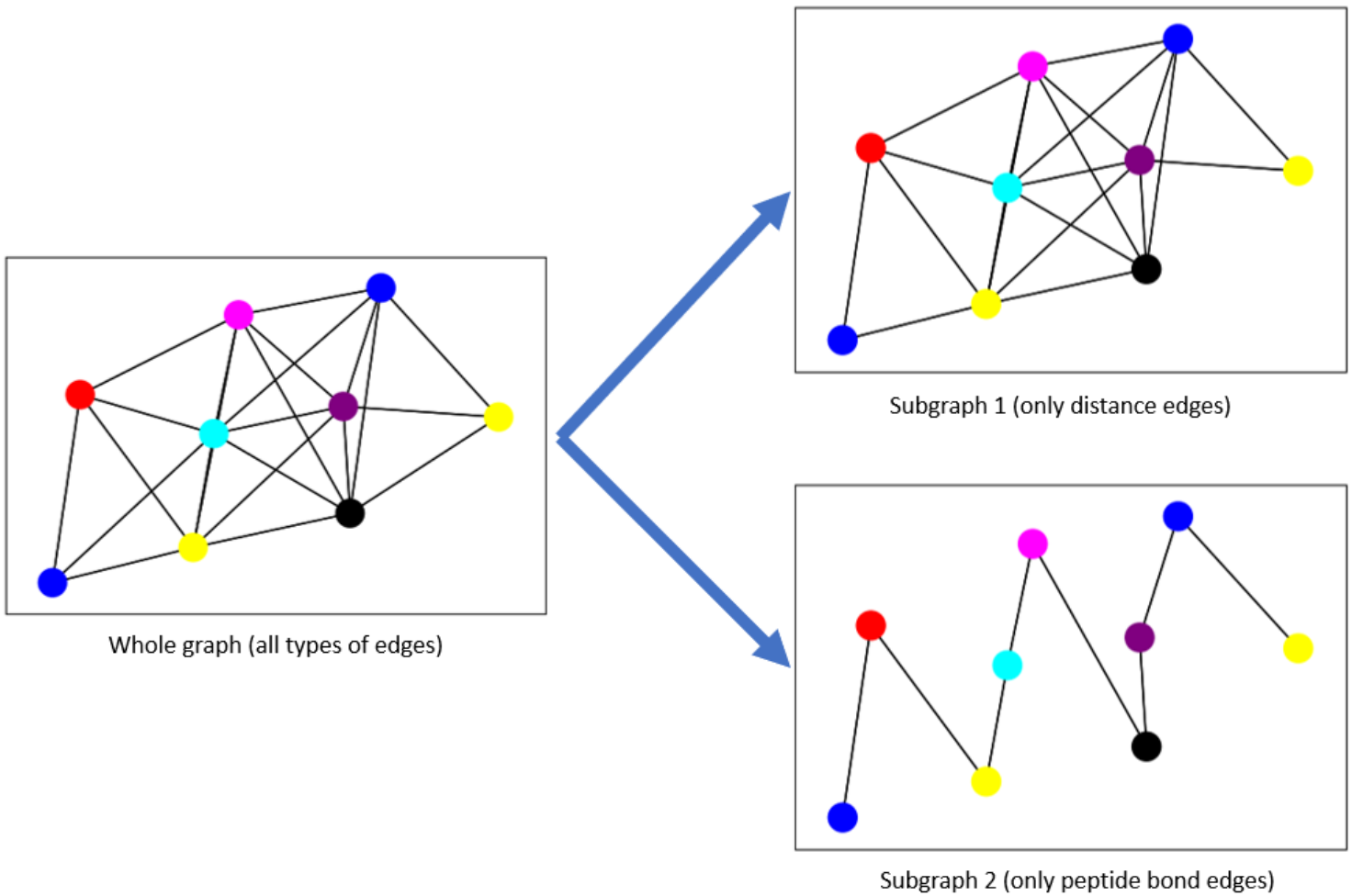

Figure 10: Edge filtering

Overview

Framework

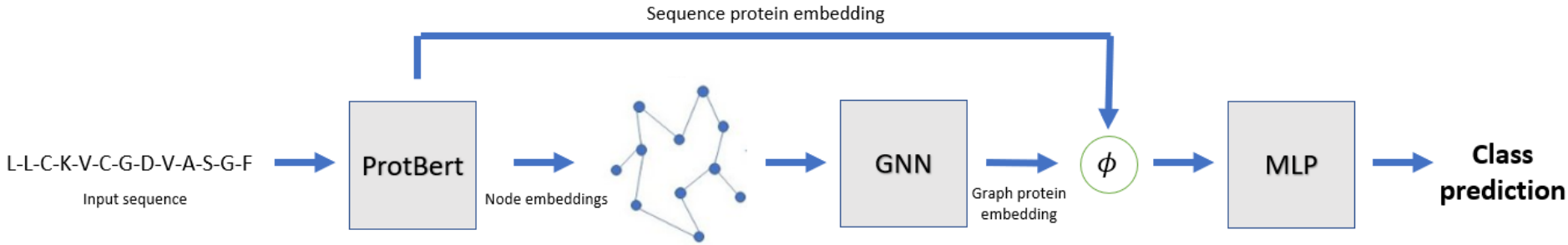

Figure 11: Architecture overview

GNNs

- Implementation with DGL [Wang et al., 2019]
- GCN [Kipf et al., 2017]
- GAT [Veličković et al., 2018]
- HGPSL [Zhang et al., 2019]

Implementation Details

Training

- 50 epochs with early stopping, batch size $= 64$
- Adam optimizer, lr $= 0.001$, StepLR
- Train-val split of 85%

Models

- GCN & GAT $\rightarrow$ 2 graphs layers + 2-layer MLP
- HGPSL $\rightarrow$ default setting
- $n_{hid} \in [128, 256, 512, 1024]$ for message passing layers

Overview

Notations

Node embedding of dimension k

- $G_{all} \rightarrow$ original node attributes, $k = 86$ (baseline)
- $G_{BERT} \rightarrow$ BERT embeddings, $k = 1024$

Notations

Node embedding of dimension k

- $G_{all} \rightarrow$ original node attributes, $k = 86$ (baseline)
- $G_{BERT} \rightarrow$ BERT embeddings, $k = 1024$

Protein embedding

- $P_{Tfidf} \rightarrow$ TF-IDF features of the protein sequences (baseline)
- $P_{BERT} \rightarrow$ BERT embeddings of protein sequences

Notations

Node embedding of dimension k

- $G_{all} \rightarrow$ original node attributes, $k = 86$ (baseline)
- $G_{BERT} \rightarrow$ BERT embeddings, $k = 1024$

Protein embedding

- $P_{Tfidf} \rightarrow$ TF-IDF features of the protein sequences (baseline)
- $P_{BERT} \rightarrow$ BERT embeddings of protein sequences

Multi-modal $G_{BERT} + \lambda P_{BERT}$

- Scale protein embedding by $\lambda \in \{0.1, 0.2\}$
- Sum graph and protein embeddings

Results

Model	Embeddings	Hidden dimension	$\mathcal{L}$
LogReg	P_{Tfidf}	$\star$	1.69
LogReg	$\mathbf{P_{BERT}}$	$\star$	**0.964**
HGPSL	$\mathbf{G_{BERT}}$	**128**	**1.106**
HGPSL	$G_{BERT} + P_{BERT}$	128	1.132
GCN	G_{all}	64	1.966
GCN	G_{BERT}	256	0.788
GCN	$\mathbf{G_{BERT} + \lambda P_{BERT}}$	**256**	**0.779**
GCN	G_{BERT}	512	0.848
GCN	$\mathbf{G_{BERT} + \lambda P_{BERT}}$	**512**	**0.809**
GCN	G_{BERT}	1024	0.845
GCN	$\mathbf{G_{BERT} + \lambda P_{BERT}}$	**1024**	**0.794**

Table 3: Loss value on test set

Overview

Conclusion

Promising results

- Use of adapted models for structured data
- Great performance even with simple GNNs
- Combining embeddings outperforms both approaches

Further work

- Other LM like T5 and XLNet might outperform BERT
- Investigate other approaches for unbalanced classification
- Take advantage of graph tools like k-core, graph kernels, . . .

Thanks for your attention !

References

Yang et al. (2019)
XLNet: Generalized Autoregressive Pretraining for Language Understanding
Advances in Neural Information Processing Systems

Ofer et al. (2021)
The language of proteins: NLP, machine learning & protein sequences
Computational and Structural Biotechnology Journal

Elnaggar et al. (2021)
ProtTrans: Towards Cracking the Language of Lifes Code Through Self-Supervised Deep Learning and High Performance Computing
IEEE Transactions on Pattern Analysis and Machine Intelligence

Raffel et al. (2020)
Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer
Journal of Machine Learning Research

References

Kipf et al. (2017)
Semi-Supervised Classification with Graph Convolutional Networks
International Conference on Learning Representations

Veličković et al. (2018)
Graph Attention Networks
International Conference on Learning Representations

Zhang et al. (2019)
Hierarchical Graph Pooling with Structure Learning
Advances in Neural Information Processing Systems

Wang et al. (2019)
Deep Graph Library: Towards Efficient and Scalable Deep Learning on Graphs
Computing Research Repository

Appendix

Model	Embeddings	Hidden dimension	$\mathcal{L}$
LogReg	P_{Tfidf}	$\star$	1.69
LogReg	$\mathbf{P_{BERT}}$	$\star$	**0.964**
HGPSL	$\mathbf{G_{BERT}}$	**128**	**1.106**
HGPSL	$G_{BERT} + P_{BERT}$	128	1.132
GCN	G_{all}	64	1.966
GCN	G_{BERT}	128	0.856
GCN	$G_{BERT} + \lambda P_{BERT}$	128	0.856
GCN	G_{BERT}	256	0.788
GCN	$\mathbf{G_{BERT} + \lambda P_{BERT}}$	**256**	**0.779**
GCN	G_{BERT}	512	0.848
GCN	$\mathbf{G_{BERT} + \lambda P_{BERT}}$	**512**	**0.809**
GCN	G_{BERT}	1024	0.845
GCN	$\mathbf{G_{BERT} + \lambda P_{BERT}}$	**1024**	**0.794**

Table 4: Loss value on test set