Protein-STORY: Semantic Text-Oriented Representation Yields biologically meaningful Protein embeddings

Anonymous ACL submission

Abstract

Unsupervised representation learning using masked language modeling on the *'language of life'* has transformed protein research, enabling the analysis of a protein universe that is expanding at an exponential pace. However, most current models rely solely on sequence data, overlooking decades of expert-curated biological knowledge stored in natural language. While recent multimodal and knowledge-graph-based approaches attempt to bridge this gap, they often rely on shallow functional labels that lack the contextual depth of full textual narratives. We present Protein-STORY, a general pipeline that synthesizes protein embeddings from diverse, multi-source text descriptions. At the core of our approach is a novel network architecture designed for the semantic compression of multi document embeddings, which integrates high-fidelity functional and structural insights into a unified representation. Our experiments demonstrate that Protein-STORY produces biologically meaningful embeddings ($r \approx 0.75$) that outperform existing models on diverse downstream tasks ($+2\%$ F1 in function prediction). Furthermore, by projecting the *'story'* of a protein into a natural language semantic space, our model enables effective zero-shot text-prompted protein search.

1 Introduction

Proteins are the fundamental building blocks of life, governing a complex network of interconnected functional mechanisms within an organism (Levitt, 2009). As their biological functions and 3D structures are primarily encoded in their amino acid sequences, effectively the *'language of life'*(Heinzinger et al., 2019), a plethora of research has been conducted to develop sequence-based protein representation learners (Ibtehaz and Kihara, 2023). This trend has been further inspired by the success of Natural Language Processing , employing unsupervised masked language modeling (Devlin et al., 2019), which has managed to learn robust protein representations suitable for several applications (Meier et al., 2021; Lin et al., 2023). While the exponential growth of unannotated protein databases makes unsupervised learning a necessity (Bateman et al., 2023), this paradigm often overlooks decades of expert-curated biological knowledge stored in textual formats.

While unsupervised representation learning is the standard in NLP due to the scarcity of large-scale annotations, protein research benefits from decades of expert-curated biological knowledge. Current protein language models largely overlook these insights, treating even well-studied proteins as unannotated and fails to leverage established textual information. We argue that integrating such structured knowledge into representation learning captures higher-order biological insights that sequence-only models cannot fully comprehend.

Recent efforts have leveraged knowledge graphs (KGs) to incorporate functional semantics. For example, OntoProtein (Zhang et al., 2022) and its successors, KeAP (Zhou et al., 2023) and Kara (Zhang et al., 2025), utilize Gene Ontology (GO) terms to align sequence embeddings with textual descriptions. However, these models rely exclusively on restricted functional annotations and consequently focus on narrow downstream tasks. While broader KG-based learners, such as, Otter-Knowledge (Lam et al., 2023) incorporate pathways and protein families, they often utilize only category names, lacking the contextual depth necessary for nuanced biological reasoning. Consequently, a significant gap remains in developing general-purpose representations that move beyond simple label-matching to integrate high-fidelity text-based biological knowledge.

To address these limitations, we propose Protein-STORY, a pipeline for generating unified protein representations from variable, heterogeneous textual data. Our primary contributions include:

- A semantic compression network specifically designed to distill multi-view document embeddings into a cohesive semantic signature.
- The integration of high-fidelity biological narratives, capturing structural, functional and evolutionary contexts that enhance performance on downstream tasks.
- A text-aligned embedding space that facilitates zero-shot protein search, bridging the gap between natural language and proteins.

2 Methodology

2.1 Problem Formulation

We consider a protein as a set of text annotations describing it, e.g., structure, function, evolution, interaction etc. Formally, a protein $\mathcal{P}$ is defined as $\mathcal{P} = \{t_1, t_2, \ldots, t_n\}$

Our goal is to generate a unified embedding $\mathcal{E} \in \mathbf{R}^d$ that aggregates all aforementioned text information, i.e., $\mathcal{E} = \textsc{Protein-Story}(\mathcal{P})$

In a way, we aim to perform semantic compression of a set of embeddings, maintaining a balance between local and global context.

2.2 Data Source

Proteins, being well-studied, are analyzed in millions of literature and hundreds of curated databases. For this work, we limit our text description data to the following features:

i) Family: Proteins sharing origin and function.

ii) Domain: Independent functional or structural unit of a protein.

iii) Homologous Superfamily: Proteins possibly evolutionarily related, low sequence similarity but sharing a structural fold

Domains and families offer local vs global context, while homologous superfamilies reveal distant evolution and remote homology that is challenging to detect. For each protein, we collect the text descriptions of the associated domains, families and supperfamilies from InterPro (Blum et al., 2021), which is a compilation of 13 large databases of these features and also provides expert curated, literature supported rich text descriptions for them.

We collected 11.29 M proteins from InterPro database, having 312,517 different combinations of aforementioned text features. Proteins from SwissProt database (Poux et al., 2017), which are expert curated, are used in different evaluations and thus proteins having similar features are removed from the training dataset, reducing the dataset size to 6.59 M proteins and 309,930 feature combinations.

2.3 Proposed Architecture

We propose a multi-stage architecture to decompose protein-related textual descriptions into a unified embedding. The model is engineered to handle heterogeneous data from diverse sources and in varying quantities; it is specifically designed to project the textual narrative of a protein into the semantic representation space of natural language. The primary components are as follows (Fig. 1):

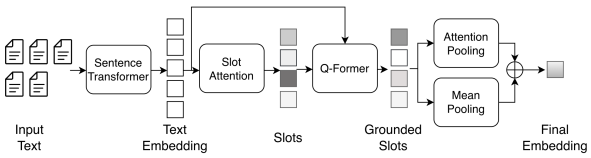

Figure 1: Overview of Protein-STORY.

i) Embedder: Individual texts are encoded using a Sentence Transformer model to generate consistent and comparable embeddings. Specifically, we employ the pubmedbert-base-embeddings model (Mezzetti; Gu et al., 2022) due to its proficiency in processing biomedical literature.

$$e_i = \textit{Sentence-Transformer}(t_i), \forall t_i \in \mathcal{P}$$

ii) Disentangler: Protein descriptions often contain redundancy and varying levels of focus. To address this, we employ slot attention (Locatello et al., 2020) and disentangle the descriptions. This module samples $k(= 8)$ slots, S from a normal distribution, which then compete to understand the input features through an iterative competitive attention mechanism. This process partitions the input into a set of specialized concepts, suppressing redundancy while preserving distinctive features.

$$S = \textit{Slot-Attention}(\{e_1, e_2, \ldots, e_n\})$$

iii) Contextual Grounder: To ensure that the abstracted slots remain contextually faithful to the original input, we utilize a novel Slot-Conditioned Q-Former (Li et al., 2023). Rather than using globally learned query vectors, we utilize the slots themselves as queries to attend over the sequence of embeddings (e). This grounding mechanism allows the model to refine disentangled representations by re-incorporating fine-grained details from the raw input context. Furthermore, by conditioning on dynamic slots rather than static learnable queries, the model can accommodate open-ended protein descriptions without the template-based biases inherent in learnable but shared query sets.

$$x = \mathit{QueryFormer}(q = S, kv = e)$$

iv) Aggregator: Finally, we aggregate the grounded slots using an attention-based pooling mechanism, driven by a single layer of multi-head self-attention. To preserve global information, we incorporate a residual connection by adding the mean of the slots to the attention output. This fusion of learned attention and mean pooling generates our final protein representation, $\mathcal{E}$.

$$\mathcal{E} = \mathit{Attn\text{-}Pool}(x) + \mathit{Mean\text{-}Pool}(x)$$

By aligning the embedding $\mathcal{E}$ with the sentence transformer's representation space, we effectively capture the protein's *'story'* within a single vector

2.4 Training

We train the model primarily with the retrieval loss based on SupCon (Khosla et al., 2020). Moreover, we propose some coverage, activity and diversity loss for regulating our slots (see Appendix). The model was trained for 30 epochs with the AdamW optimizer (Loshchilov and Hutter, 2019). 20% of the training data was used as the validation split.

3 Experiments and Results

3.1 Multi-Document Retrieval Performance

We evaluate our ability to generate compressed yet informative representations through a retrieval task: the aggregated embedding serves as a query to retrieve its constituent input embeddings from a shared vector space. To ensure robustness, we select proteins from SwissProt with at least 10 associated text features, resulting in 3,801 proteins.

We compare our approach against several baselines: mean and attention pooling, multi-head self-attention (via a [CLS] token), and the Set Transformer (Lee et al., 2019). As shown in Table 1, we report Mean Average Precision (mAP), Normalized Discounted Cumulative Gain (nDCG), Recall@|R| (total relevant items), and Median Positive Rank. Our method consistently outperforms all baselines across all metrics. This demonstrate our model's capacity to aggregate disparate information into a unified representation while preserving the fine-grained signals of individual views, effectively balancing global context with local detail.

3.2 Biologically Meaningful Embedding Space

Protein-STORY embeddings provide a compressed representation of rich biological information. While protein language models (PLMs) are the de-facto standard (Weissenow and Rost, 2025), sequence-based patterns alone often fail to capture the depth of curated experimental and expert knowledge. We evaluate Protein-STORY's representational quality against ESM2-3B (Lin et al., 2023), a state-of-the-art PLM.

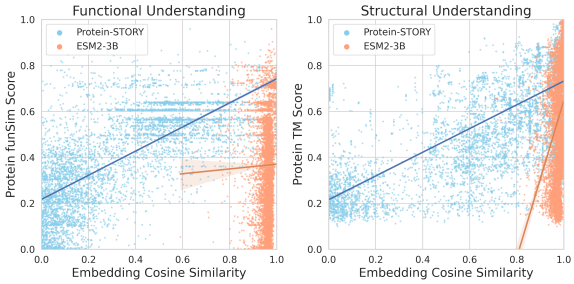

Figure 2: Functional and Structural Meaningfulness.

First, we measure the correlation between embedding similarity and biological similarity (funSim for function, TM-score for structure). Protein-STORY embeddings show strong correlations (76.03 and 74.50), surpassing the weak alignment of ESM2-3B (32.65 and 1.42; Fig. 2). Additionally, linear probing for Enzyme Commission (EC) and CATH class prediction (Sillitoe et al., 2021) (Table 3) confirms that our representations substantially outperform ESM2-3B in macro F1 scores.

These results demonstrate that while unsupervised pLMs are versatile, Protein-STORY more effectively condenses heterogeneous biological knowledge into a unified embedding that can accurately reflect essential protein attributes.

3.3 Downstream Task Performance

To evaluate the downstream utility of our embeddings, we utilize the PROBE benchmark (Unsal et al., 2022) across three tasks: semantic similarity, gene ontology, and drug-target protein family classification. As shown in Table 2, Protein-STORY yields consistent improvements across all metrics.

Notably, our embeddings outperform Domain-PFP (Ibtehaz et al., 2023) by 2%, which is remarkable given that Domain-PFP is specifically developed to leverage the same features through functional alignment. This suggests that our model effectively integrates biologically rich text information for downstream tasks. Furthermore, Protein-STORY achieves higher semantic similarity scores than OntoProtein (Zhang et al., 2022) and KeAp (Zhou et al., 2023). This is particularly striking, as it implies our model implicitly captures functional

Table 1: Retrieval performance of multi-view embeddings compared to baseline

Method	#params	mAP	nDCG@20	Recall@|R|	Median Positive Rank
Mean Pooling	0	0.462	0.59	0.456	18
Attention Pooling	0.79 M	0.47	0.597	0.464	17
MHSA	26.78 M	0.494	0.618	0.48	15
Set Transformer	21.06 M	0.486	0.612	0.474	15
Protein-STORY	19.42 M	**0.512**	**0.626**	**0.492**	**14**

Table 2: PROBE benchmark evaluation : we report the best score achieved by any other method as PROBE-best.

	Semantic Similarity (ρ)				Gene Ontology (weighted F1)				Drug Target Protein (MCC)			
	MF	BP	CC	Avg.	MF	BP	CC	Avg.	Random	50%	30%	15%
PROBE-best	0.57	0.58	0.51	0.51	0.92	0.72	0.74	0.79	0.92	0.92	0.92	0.90
Protein-STORY	**0.73**	**0.66**	**0.55**	**0.65**	**0.93**	**0.73**	**0.76**	**0.81**	**0.94**	**0.93**	**0.93**	**0.91**

Table 3: Linear probing macro-F1 Score.

Classification	Protein-STORY	ESM2-3B
EC	78.24 ± 2.27	61.1 ± 2.45
CATH	$\mathbf{83.45 \pm 0.23}$	$\mathbf{73.29 \pm 1.66}$

relationships more effectively than methods that utilize explicit functional information.

3.4 Zero-Shot Protein Search via Natural Language

A key advantage of Protein-STORY is the direct projection of protein representations into the language model's embedding space. This alignment treats proteins as textual narratives, mapping them into a semantic space consistent with natural language and enables text-prompted protein retrieval.

To evaluate this capability, we considered SwissProt protein embeddings as a vector database. Query embeddings were generated by processing natural language prompts through the same sentence transformer model. We performed similarity searches using the FAISS library (Douze et al., 2025), employing an exact search configuration (roughly half a million entries). For each query, we retrieved the top 10 nearest neighbors.

The system was tested using complex functional prompts. For the query: *'Identify extracellular proteins involved in the regulation of blood coagulation that utilize specialized structural modules to bind to membrane phospholipids or other protein mediators.'* all top 10 results belonged to the Fibrinogen C-terminal domain-containing family. These proteins facilitate platelet aggregation via the binding of platelet receptor integrin α(IIb)-β(3) to the fibrinogen C-terminal D domain (Podolnikova et al., 2003). This demonstrates that the systen successfully captures the specific functional and structural requirements specified in the query.

Similarly, for the query: *'Find intracellular signaling proteins containing SH3 domains that localize to the plasma membrane upon phosphorylation to regulate actin cytoskeleton remodeling.'* the top 3 results were explicitly actin cytoskeleton-regulatory proteins, and the remaining 7 contained SH3 domains. Notably, 9 of the top 10 results are localized to the plasma membrane. These findings demonstrate that the embedding space effectively encodes multi-faceted biological constraints, including domain architecture, subcellular localization, and specific signaling pathways.

3.5 Model Interpretation and Ablation Study

Please refer to the appendix.

4 Conclusion

In this work, we have developed Protein-STORY, a text-guided representation learner for proteins. While existing self-supervised methods are highly competitive, they often neglect decades of rich biological knowledge stored in textual formats. Our comprehensive analyses demonstrate that integrating textual narratives as features significantly enhances protein representations, bridging the gap between raw sequences and established functional insights. These results underscore the potential of text-aligned models to capture a more holistic understanding of biological systems.

Moving forward, we intend to explore a broader range of textual sources to further enrich the semantic depth of protein representations. Furthermore, we plan to evaluate the generalizability of the framework in domains beyond biology.

Limitations

The primary limitation of this work is the restricted scope of protein-associated textual data. Currently, we utilize only family, domain, and superfamily annotations, omitting critical metadata such as biochemical properties, metabolic pathways, and scientific literature. Incorporating these multi-faceted sources would likely enhance the semantic robustness and versatility of the resulting embeddings.

Additionally, our methodology simplifies text into fixed-length vectors via sentence transformers. While effective for broad conceptual alignment, this 'bottleneck' may overlook fine-grained semantic nuances and token-level relationships. Future work should explore using Large Language Models (LLMs) to operate directly on raw text, enabling a more sophisticated synthesis of complex protein-text interactions.

References

Alex Bateman, Maria-Jesus Martin, Sandra Orchard, Michele Magrane, Shadab Ahmad, Emanuele Alpi, Emily H Bowler-Barnett, Ramona Britto, Hema Bye-A-Jee, Austra Cukura, Paul Denny, Tunca Dogan, ThankGod Ebenezer, Jun Fan, Penelope Garmiri, Leonardo Jose da Costa Gonzales, Emma Hatton-Ellis, Abdulrahman Hussein, Alexandr Ignatchenko, and 95 others. 2023. UniProt: the Universal Protein Knowledgebase in 2023. *Nucleic Acids Research*, 51(D1):D523–D531.

Matthias Blum, Hsin-Yu Chang, Sara Chuguransky, Tiago Grego, Swaathi Kandasaamy, Alex Mitchell, Gift Nuka, Typhaine Paysan-Lafosse, Matloob Qureshi, Shriya Raj, Lorna Richardson, Gustavo A Salazar, Lowri Williams, Peer Bork, Alan Bridge, Julian Gough, Daniel H Haft, Ivica Letunic, Aron Marchler-Bauer, and 14 others. 2021. The InterPro protein families and domains database: 20 years on. *Nucleic Acids Research*, 49(D1):D344–D354.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 4171–4186, Stroudsburg, PA, USA. Association for Computational Linguistics.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2025. The Faiss library.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2022. Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23.

Michael Heinzinger, Ahmed Elnaggar, Yu Wang, Christian Dallago, Dmitrii Nechaev, Florian Matthes, and Burkhard Rost. 2019. Modeling aspects of the language of life through transfer-learning protein sequences. *BMC Bioinformatics*, 20(1):723.

Nabil Ibtehaz, Yuki Kagaya, and Daisuke Kihara. 2023. Domain-PFP allows protein function prediction using function-aware domain embedding representations. *Communications Biology*, 6(1):1103.

Nabil Ibtehaz and Daisuke Kihara. 2023. Application of Sequence Embedding in Protein Sequence-Based Predictions. In *Machine Learning in Bioinformatics of Protein Sequences*, pages 31–55. WORLD SCIENTIFIC.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Hoang Thanh Lam, Marco Luca Sbodio, Marcos Martínez Galindo, Mykhaylo Zayats, Raúl Fernández-Díaz, Víctor Valls, Gabriele Picco, Cesar Berrospi Ramis, and Vanessa López. 2023. Otter-Knowledge: benchmarks of multimodal knowledge graph representation learning from different sources for drug discovery.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam R. Kosiorek, Seungjin Choi, and Yee Whye Teh. 2019. Set Transformer: A Framework for Attention-based Permutation-Invariant Neural Networks.

Michael Levitt. 2009. Nature of the protein universe. *Proceedings of the National Academy of Sciences*, 106(27):11079–11084.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. 2023. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. 2020. Object-Centric Learning with Slot Attention.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

Joshua Meier, Roshan Rao, Robert Verkuil, Jason Liu, Tom Sercu, and Alex Rives. 2021. Language models enable zero-shot prediction of the effects of mutations on protein function. In *Advances in Neural Information Processing Systems*, volume 34, pages 29287–29303. Curran Associates, Inc.

David Mezzetti. NeuML/pubmedbert-base-embeddings (Version 1.0.0). Hugging Face.

Nataly P. Podolnikova, Valentin P. Yakubenko, George L. Volkov, Edward F. Plow, and Tatiana P. Ugarova. 2003. Identification of a Novel Binding Site for Platelet Integrins αIIbβ3 (GPIIbIIIa) and α5β1 in the γC-domain of Fibrinogen. *Journal of Biological Chemistry*, 278(34):32251–32258.

Sylvain Poux, Cecilia N Arighi, Michele Magrane, Alex Bateman, Chih-Hsuan Wei, Zhiyong Lu, Emmanuel Boutet, Hema Bye-A-Jee, Maria Livia Famiglietti, Bernd Roechert, and The UniProt Consortium. 2017. On expert curation and scalability: UniProtKB/Swiss-Prot as a case study. *Bioinformatics*, 33(21):3454–3460.

Ian Sillitoe, Nicola Bordin, Natalie Dawson, Vaishali P Waman, Paul Ashford, Harry M Scholes, Camilla S M Pang, Laurel Woodridge, Clemens Rauer, Neeladri Sen, Mahnaz Abbasian, Sean Le Cornu, Su Datt Lam, Karel Berka, Ivana Hutařová Varekova, Radka Svobodova, Jon Lees, and Christine A Orengo. 2021. CATH: increased structural coverage of functional space. *Nucleic Acids Research*, 49(D1):D266–D273.

Serbulent Unsal, Heval Atas, Muammer Albayrak, Kemal Turhan, Aybar C. Acar, and Tunca Doğan. 2022. Learning functional properties of proteins with language models. *Nature Machine Intelligence*, 4(3):227–245.

Konstantin Weissenow and Burkhard Rost. 2025. Are protein language models the new universal key? *Current Opinion in Structural Biology*, 91:102997.

Jiasheng Zhang, Delvin Ce Zhang, Shuang Liang, Zhengpin Li, Rex Ying, and Jie Shao. 2025. Retrieval-Augmented Language Model for Knowledge-aware Protein Encoding. In *Forty-second International Conference on Machine Learning*.

Ningyu Zhang, Zhen Bi, Xiaozhuan Liang, Siyuan Cheng, Haosen Hong, Shumin Deng, Qiang Zhang, Jiazhang Lian, and Huajun Chen. 2022. OntoProtein: Protein Pretraining With Gene Ontology Embedding. In *International Conference on Learning Representations*.

Hong-Yu Zhou, Yunxiang Fu, Zhicheng Zhang, Bian Cheng, and Yizhou Yu. 2023. Protein Representation Learning via Knowledge Enhanced Primary Structure Reasoning. In *The Eleventh International Conference on Learning Representations*.

A Appendix

A.1 Experimental Setup

All the experiments were performed in a linux server equipped with AMD EPYC 7313P 16-Core Processor, 128 GB Ram and 2x NVIDIA RTX A6000, 48 GB GPUs.

A.2 Model Parameter

Our model has a total of 19.42M parameters and requires 0.5–1.0 GFLOPs.

A.3 Code Availability

The model design, data processing, training, and experimental code were developed by the authors. The implementation is based on PyTorch v2.2.0+cu121. The code will be released under the GPL license.

A.4 Dataset Description

We collect text descriptions of protein from the InterPro database, which is a compilation of 13 databases dealing with various functional and structural features of proteins. The most useful thing about InterPro database is that they provide rich text description with literature evidence for the different protein features.

For our work, we considered 3 primary features of proteins:

1. Domain : Domains are independent structural or functional modules within a protein. Each domain typically executes a specific interaction or task, collectively supporting the protein's overall biological activity. Notably, these units are versatile, nearly identical domains are frequently identified across a wide range of proteins with otherwise unrelated functions.

2. Family : A protein family is a group of proteins that are evolutionarily linked. Because they descend from a common ancestor, they usually share similar amino acid sequences, three-dimensional shapes, and biological roles. Families can be further divided into subfamilies when certain members develop even more specific specialized functions.

3. Homolgous Superfamily : A protein superfamily is a broader category that sits above the family level. It contains multiple protein

families that are distantly related. While the proteins in a superfamily might not look very similar at a sequence level anymore, they still share a common structural 'fold' or a fundamental functional mechanism that proves they share an ancient common ancestor.

In the current version of the InterPro (Release 107.0, 16th October 2025), there are a total of 17,951 domains, 26,829 families and 3,511 homologous superfamilies. We collected text descriptions of all these entries from:

https://ftp.ebi.ac.uk/pub/databases/interpro/current_release/interpro.xml.gz

Once downloaded these text abstracts were sanitized and there embeddings were computed using the neuml/pubmedbert-base-embeddings sentence transformer, which is optimized on pubmed abstracts.

Next, we collect all the precomputed features of 11.29 million proteins from https://ftp.ebi.ac.uk/pub/databases/interpro/current_release/protein2ipr.dat.gz. The consist of 312,517 combinations of the aforementioned features. The distribution protein features are shown in Fig. 3

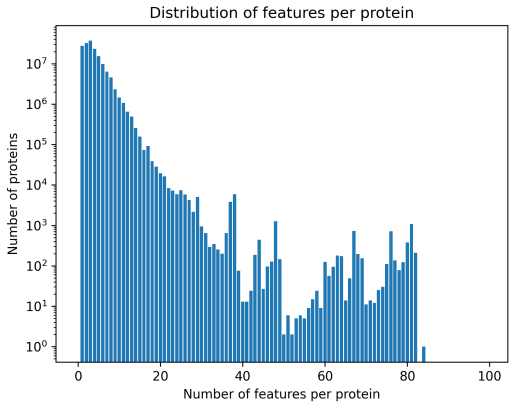

Figure 3: Distribution of features per protein

Since, well-curated proteins from SwissProt dataset are commonly used in benchmarks, we remove proteins having similar features. This reduces the dataset to 6.59 million proteins and 309,930 feature combinations.

A.5 Architecture Overview

Here we provide pseudocodes for our model implementation.

Algorithm 1: Protein-STORY Model

Input: InterPro feature tokens $T \in Z^{B\times N}$, mask $M \in \{0,1\}^{B\times N}$, optional slot count K

Output: Dictionary of embeddings and attention weights

// 1. Initialization and Embedding Extraction

$K \leftarrow (K > 0)\ ?\ K$: self.num_slots

$\mathbf{E} \leftarrow \text{Embedding}(T)$ `// Frozen pre-trained text embeddings`

$X \leftarrow \text{Linear}_{in}(\mathbf{E})$ `// Project to internal dimension` D

// 2. Feature Disentanglement

$S, \mathbf{A}_{slot} \leftarrow \text{SlotAttention}(X, K, M)$

// 3. Slot Grounding (Cross-Modal Alignment)

$S_g, \mathbf{A}_{ground} \leftarrow$ $\text{SlotConditionedQFormer}(S, X, M)$

// 4. Aggregation and Final Projection

$\mathbf{e}_{pool}, \mathbf{w}_{pool} \leftarrow$ $\text{SlotSelfAttentionPooling}(S_g)$

$\mathbf{e}_{protein} \leftarrow \text{Linear}_{out}(\mathbf{e}_{pool})$ `// Project back to LLM dimension`

$S_{grounded} \leftarrow \text{Linear}_{out}(S_g)$

return begin

"protein_emb" : $\mathbf{e}_{protein}$

"slots" : S

"grounded_slots" : $S_{grounded}$

"slot_attn" : $\mathbf{A}_{slot}$

"pool_weights" : $\mathbf{w}_{pool}$

end

Algorithm 2: Slot Attention Module

Input: Input features $X \in R^{B\times N\times D}$, number of slots K, mask $M \in \{0,1\}^{B\times N}$

Output: Refined slots $S \in R^{B\times K\times D}$, attention weights A_{raw}

// 1. Initialization

$X \leftarrow \text{LayerNorm}(X)$

$k \leftarrow XW_k, \quad v \leftarrow XW_v$ `// Project inputs to keys and values`

$\mu, \sigma \leftarrow$ learnable parameters $\in R^D$

$S \sim \mathcal{N}(\mu, \text{diag}(\sigma^2))$ `// Sample` K `slots from Gaussian prior`

// 2. Iterative Refinement

for $t = 1$ **to** T_{iter} **do**

$S_{prev} \leftarrow S$

$q \leftarrow \text{LayerNorm}(S)W_q$

// Competitive Attention (Softmax over Slots)

$logits \leftarrow \frac{1}{\sqrt{D}} qk^\top$

if M *is provided* **then**

$logits \leftarrow \text{MaskFill}(logits, M, -10^7)$

end

$A_{raw} \leftarrow \text{Softmax}(logits, \text{dim} = \text{slots})$ `//` Shape: (B, K, N)

// Weighted Mean Normalization (over Inputs)

if M *is provided* **then**

$A \leftarrow A_{raw} \cdot (\neg M)$ `// Ignore masked inputs`

$A \leftarrow A/(\sum_{j=1}^{N} A_{i,j} + \epsilon)$

end

else

$A \leftarrow A_{raw}/(\sum_{j=1}^{N} A_{i,j} + \epsilon)$

end

// Update via GRU and MLP

$updates \leftarrow Av$

$S \leftarrow \text{GRUCell}(\text{flatten}(updates), \text{flatten}(S_{prev}))$

$S \leftarrow S + \text{MLP}(\text{LayerNorm}(S))$

end

return S, A_{raw}

Algorithm 3: SlotConditioned QFormer

Input: Slots $S \in R^{B\times K\times D}$, input features $X \in R^{B\times N\times D}$, mask $M \in \{0,1\}^{B\times N}$

Output: Grounded slots $S_g \in R^{B\times K\times D}$, attention weights $\mathbf{A}$

// 1. Linear Projections

$Q \leftarrow SW_q$ `// Queries derived from slots`

$K \leftarrow XW_k, \quad V \leftarrow XW_v$ `// Keys and values derived from input sequence`

// 2. Attention Score Computation

$\text{Scores} \leftarrow \frac{QK^\top}{\sqrt{D}}$ `// Dot-product affinity` (B, K, N)

// 3. Masking and Normalization

if M *is provided* **then**

$\text{Scores} \leftarrow \text{MaskFill}(\text{Scores}, M, -10^7)$ `// Mask keys/values`

end

$\mathbf{A} \leftarrow \text{Softmax}(\text{Scores}, \text{dim} = -1)$ `// Normalize over sequence dimension` N

// 4. Context Projection

$S_g \leftarrow \mathbf{A}V$ `// Weighted sum of values`

return $S_g, \mathbf{A}$

Algorithm 4: SlotSelfAttention Pooling

Input: Grounded slots $S \in R^{B\times K\times D}$
Output: Pooled embedding $\mathbf{e}_p \in R^D$, attention weights $\mathbf{w} \in R^K$

// 1. Multi-Head Self-Attention (MHSA) among slots
$H \leftarrow$ number of heads, $\quad d_h \leftarrow D/H$
$Q, K, V \leftarrow SW_q, SW_k, SW_v$ `// Project to` (B, K, H, d_h)
$Q, K, V \leftarrow \text{Transpose}(Q, K, V)$
`// Reshape to` (B, H, K, d_h)
$\text{Scores} \leftarrow \frac{QK^\top}{\sqrt{d_h}}$ `// Compute inter-slot affinity` (B, H, K, K)
$\mathbf{A} \leftarrow \text{Softmax}(\text{Scores}, \text{dim} = -1)$
$O \leftarrow \mathbf{A}V$
$S_{attn} \leftarrow \text{Reshape}(\text{Transpose}(O))$ `// Back to` (B, K, D)
// 2. Norm-Based Importance Pooling
for *each slot* $i \in \{1, \ldots, K\}$ **do**
  $n_i \leftarrow \|s_{attn,i}\|_2$ `// Compute` L_2 `norm of refined slot`
end
$\mathbf{w} \leftarrow \text{Softmax}([n_1, \ldots, n_K])$
`// Normalize weights across` K `slots`
// 3. Aggregation and Global Residual
$\mathbf{e}_{context} \leftarrow \sum_{i=1}^{K} w_i \cdot s_{attn,i}$ `// Weighted sum of attended slots`
$\mathbf{e}_p \leftarrow \mathbf{e}_{context} + \frac{1}{K}\sum_{i=1}^{K} s_i$ `// Add global average of original slots`
return $\mathbf{e}_p, \mathbf{w}$

A.6 Loss Functions

A.6.1 Primary Loss

Since the effectiveness of our system effectively depends on how well the synthesized embedding can retrieve the input embedding components, we train the model with retrieval objective as the primary loss function. The popular SupCon loss was used for that purpose. SupCon loss has two variants, dealing with sum outside and inside of the log operation.

$$\mathcal{L}_{out}^{sup} = \sum_{i\in I} \frac{-1}{|P(i)|} \sum_{p\in P(i)} \log \frac{\exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_p/\tau\right)}{\sum_{a\in A(i)} \exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_a/\tau\right)}$$

$$\mathcal{L}_{in}^{sup} = \sum_{i\in I} -\log \left\{ \frac{1}{|P(i)|} \sum_{p\in P(i)} \frac{\exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_p/\tau\right)}{\sum_{a\in A(i)} \exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_a/\tau\right)} \right\}$$

Empirically, the out loss performs better and it is intuitive as well. The out loss focuses on satisfying the overall constraints, whereas the in loss handles each constraint independently and then combine them. Therefore, we can consider the out and in losses as global and local losses, respectively. In order to balance these two aspects, we consider our retrieval loss as follows:

$$\mathcal{L}_{retrieval} = \mathcal{L}_{out}^{sup} + 0.2 \times \mathcal{L}_{in}^{sup} \tag{1}$$

Moreover, we optimized the $\mathcal{L}_{retrieval}$ both ways, i.e., from protein to text retrieval and from text to protein retreival.

It should be noted that the baseline methods were also trained with the same loss function.

A.6.2 Slot Regularizer

To ensure the model learns a meaningful set of latent representations in the slots, that map effectively to input features (tokens), we employ a composite set of slot regularizer.

- Coverage loss : This slot penalizes if some input tokens don't receive sufficient attention

$$L_{coverage} = (1 - slot\_attn.sum(dim = 1))^2$$

- Activity loss : this slot penalizes lazy slots

$$L_{activity} = (avg\_attn - slot\_attn)^2$$

- Orthogonality loss : this loss compels the slots to learn different concepts.

$$L_{orthogonality} = (I - slot@slot.T)^2$$

- De-uniform loss : this loss prevents the slots from following a uniform pattern by minimizing negative entropy

$$L_{de-uniform} = -entropy(slot\_attn)$$

The combined slot regularizer loss is as follows:

$$\mathcal{L}_{slot} = L_{coverage} + L_{activity} + L_{orthogonality} + L_{de-uniform}$$

The overall loss is computed as a weighted combination:

$$\mathcal{L} = \mathcal{L}_{retrieval} + 0.3 \times \mathcal{L}_{slot}$$

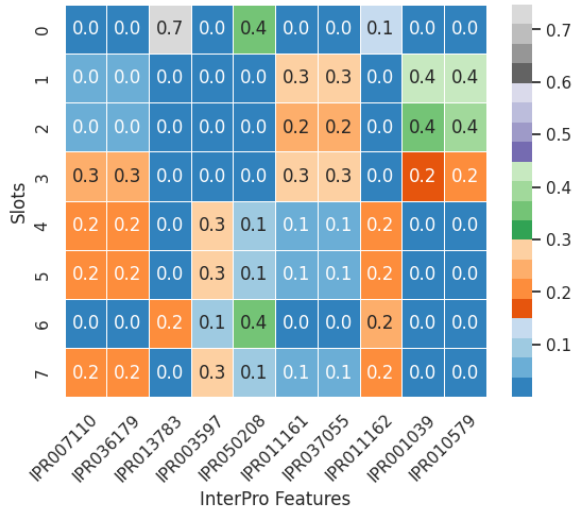

Figure 4: Slot attention weights for the protein 1A01_GORGO

A.7 Model Interpretation

A primary motivation behind our model design is interpretability, which is facilitated by our choice of Slot Attention and the Q-former architecture. As previously discussed, protein-related text features can often be redundant and inter-correlated. In such cases, a uniform pooling mechanism may over-emphasize repetitive content and fail to capture granular details. Slot Attention, however, is designed to disentangle these overlapping features.

We demonstrate this capability by analyzing the slot attention patterns for a specific protein in Fig. 4. In this visualization, slots are arranged along the y-axis and input text features along the x-axis. For example, IPR007110 and IPR036179 both correspond to immunoglobulin-like domains; across all slots, the features of these two inputs are highly correlated, suggesting the model recognizes their similarity and treats them as a unified concept. Similarly, IPR011161, IPR037055, IPR001039, and IPR010579, all related to MHC class I, exhibit correlation within the attention space. More specifically, while IPR011161 and IPR037055 relate to MHC class I-like antigen recognition, IPR011162 recognizes both MHC class I/II-like antigens, leading to shared attention across in some slots.

A.8 Ablation Study

A.8.1 Contribution of the individual components

The contribution of our individual components can be assessed by comparing the full model against the specific baselines. By removing Slot Attention and the Q-former, the model reverts to an architecture similar to Attention Pooling or a standard Multi-Head Self-Attention (MHSA) model. Furthermore, if we omit Slot Attention and utilize a standard Q-former with learned global queries, the architecture becomes equivalent to a Set Transformer. Consequently, the results presented in Table 1 serve as a sufficient ablation study of our model components.

A.8.2 Impact of retrieval loss

As retrieval loss, we considered a combination of $\mathcal{L}_{out}^{sup}$ and $\mathcal{L}_{in}^{sup}$. Empirically, $\mathcal{L}_{out}^{sup}$ performs much better than $\mathcal{L}_{in}^{sup}$, as it focuses on the global context and thus manages to satisfy majority of the constraints while also being less sensitive to outliers. On the contrary, $\mathcal{L}_{in}^{sup}$, focuses on the individual constraints and specific patterns and thus is more susceptible to noise. In our experiment we observed this interesting outcome as well. When, we trained the model with only $\mathcal{L}_{in}^{sup}$, during retrieval, it managed to extract a few hits in much earlier rank, but at the same time missed a good amount of candidates. On the other extreme, training the model with only $\mathcal{L}_{in}^{sup}$ improved overall recall, but the hits started coming at later ranks. Therefore, as a mean to balance this two opposing behavior we considered a weighted sum of the two losses.

A.8.3 Impact of slot regularizer

The selection of slot regularization functions proved to be particularly interesting and significant. Omitting the coverage and activity losses resulted in the under-representation of certain tokens and the inactivity of specific slots, respectively. Similarly, training the model without an orthogonality loss led to highly correlated and redundant slot features. The de-uniform regularizer was another critical component; without it, the attention patterns became almost entirely flat—exhibiting a uniform focus across all inputs—which effectively caused the attention mechanism to collapse. Only by incorporating these slot regularizers into the objective function were we able to achieve the meaningful attention patterns illustrated in Fig. 4.

A.8.4 Dependence of number of slots

The number of slots (K) to consider, is an important hyperparameter for our pipeline. We have observed that for our current set of inputs, the representation gets saturated after $K = 8$ and further increasing slots marginally changes the performance despite the added computational complexity. How-

Table 4: Impact of retrieval loss

Loss	hit@1	MRR	MAP	Recal@|R|
$\mathcal{L}_{out}^{sup}$	0.68	0.791	0.519	0.495
$\mathcal{L}_{in}^{sup}$	0.821	0.889	0.421	0.416
$\mathcal{L}_{out}^{sup} + 0.2 \times \mathcal{L}_{in}^{sup}$	0.686	0.797	0.512	0.492

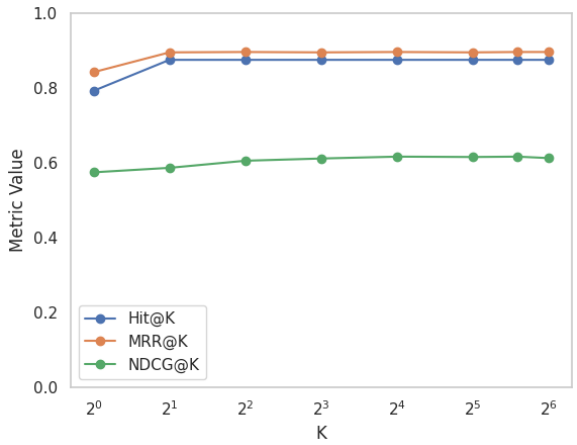

Figure 5: Retrieval Metrics vs K

ever, for less than 8 slots, the performance falls, as shown in Fig 5.

A.9 Details on Linear Probing Experiment

We conducted two linear probing experiments using the Swiss-Prot database. The tasks included Enzyme Commission (EC) number prediction and CATH class prediction, representing a functional and a structural task, respectively. Our dataset consisted of 7 EC classes and 5 CATH classes. In cases where a protein possessed multiple class memberships, we assigned the majority class label. Following this preprocessing, we obtained 270,689 proteins with EC annotations and 477,027 proteins with CATH classifications. We then performed 3-fold stratified cross-validation using a scikit-learn logistic regression model and reported the results using macro F1 scores.

A.10 Details on PROBE benchmark

The PROBE benchmark assesses the how functionally informative protein representations are. A list of 20,000 human proteins are provided by the benchmark, and users need to submit embeddings for those proteins. We considered 3 tasks from this benchmark.

1. **Semantic Similarity Inference**: This task measures the degree of functional semantic similarity of the protein representations, i.e., which representation vectors capture functional information by comparing the pairwise similarity of protein feature vectors (using Manhattan, Cosine, and Euclidean distances) against ground-truth functional similarities derived from Gene Ontology (GO) annotations through Lin similarity score. For the final evaluation Manhattan distance is considered.

2. **Ontology-based Protein Function Prediction (PFP)**: A supervised classification task where representations are used to predict specific GO terms across three categories: Molecular Function (MF), Biological Process (BP), and Cellular Component (CC). There are a total of 117 GO terms in this benchmark and they are annotated with experimental evidence. The representations are used as features to a linear classifier and 5 fold cross-validation is performed. The weight F1 score is metric considered in this task.

3. **Drug Target Protein Family Classification**: This task assesses the representation's capacity to identify structural and functional protein families (e.g., enzymes, membrane receptors, ion channels) crucial for drug discovery. MCC is the primary metric for this evaluation.

We downloaded the benchmark from https://github.com/kansil/PROBE and ran the experiments locally using Protein-STORY embeddings.

For the sake of simplicity we only considered the best performing method agaisnt each metric. To the best of our knowledge, the best performing methods in this benchmark, i.e., PROBE-best are:

- Semantic Similarity
    1. MF : ProtT5-XL
    2. BP : Mut2Vec
    3. CC : PFAM
    4. Avg : Mut2Vec
- Protein Function Prediction

1. MF : Domain-PFP
2. BP : Domain-PFP
3. CC : Domain-PFP
4. Avg : Domain-PFP

- Drug Target Protein

1. Random : ProtT5-XL
2. 50% : ProtT5-XL
3. 30% : ProtT5-XL
4. 15% : ProtT5-XL

Table 5: Results of Zero Shot Protein Search

Query	Hits
Identify extracellular proteins involved in the regulation of blood coagulation that utilize specialized structural modules to bind to membrane phospholipids or other protein mediators.	MFAP4_HUMAN, FBCDA_XENLA, FCN2_MOUSE, FBCD1_HUMAN, FCN2_PIG, FBCD1_XENTR, FCNV3_CERRY, TLLP_PHONI, FGL2_HUMAN, FCN1B_XENLA
Find intracellular signaling proteins containing SH3 domains that localize to the plasma membrane upon phosphorylation to regulate actin cytoskeleton remodeling.	SLA1_SCHPO, SLA1_SCLS1, SLA1_MYCMD, SH3Y1_HUMAN, SH3Y1_RAT, SH3Y1_XENLA, SH3Y1_BOVIN, SH3Y1_MOUSE, SH3Y1_PONAB, LSB3_YEAS7
Identify proteins localized to the nucleolus	NOL6_DROYA, NOL6_DROSI, UTP22_SCHPO, NOL6_HUMAN, UTP22_YEAST, NOL6_DROMO, NOL6_DROWI, NOL6_DROVI, NOL6_DROME, NOL6_DROPE