TechRxiv
Powered by IEEE

# ProteinChat: Towards Achieving ChatGPT-Like Functionalities on Protein 3D Structures

This paper was downloaded from TechRxiv (https://www.techrxiv.org).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

24-05-2023 / 26-05-2023

CITATION

Guo, Han; Huo, Mingjia; Zhang, Ruiyi; Xie, Pengtao (2023): ProteinChat: Towards Achieving ChatGPT-Like Functionalities on Protein 3D Structures. TechRxiv. Preprint. https://doi.org/10.36227/techrxiv.23120606.v1

DOI

10.36227/techrxiv.23120606.v1

# ProteinChat: Towards Achieving ChatGPT-Like Functionalities on Protein 3D Structures

**Han Guo***, **Mingjia Huo***, **Ruiyi Zhang**, **Pengtao Xie**

UC San Diego *Equal contribution

## Abstract

The study of proteins is critical in various scientific disciplines, but understanding their complex structure-function relationships remains challenging. Recent advancements in large language models (LLMs) have demonstrated their ability to comprehend task-specific knowledge, suggesting the potential for specially trained ChatGPT-like systems to accelerate protein research. In this work, we introduce ProteinGPT, a prototype model aimed at learning and understanding protein 3D structures. ProteinGPT enables users to upload proteins, ask questions, and engage in interactive conversations to gain insights. The ProteinChat system consists of three main components: a composite encoder block, a projection layer, and an LLM. The protein undergoes encoding to form a protein embedding, which is then projected to conform with the LLM. The LLM combines user questions with the embedding to generate informative answers. To train ProteinChat, we curated the RCSB-PDB Protein Description Dataset, comprising 143,508 protein-description pairs from publicly available sources. By leveraging the capabilities of ProteinGPT, researchers can potentially expedite their investigations into protein functionalities and structures, benefiting areas such as drug development and therapeutics, food science and nutrition, and various aspects of our lives. This initial step lays the foundation for further exploration and utilization of the ChatGPT-like system in protein research. The code is available at `https://github.com/UCSD-AI4H/proteinchat` and the dataset can be downloaded from `https://drive.google.com/file/d/1AeJW5BY5C-d8mKJjAULTax6WA4hzWSON/view?usp=share_link`.

## 1 Introduction

Proteins play a critical role in biological processes, and understanding the structure, function, and interactions of proteins is crucial to the advancement of biological and biomedical research. However, traditional protein engineering involves manual and often laborious processes of studying proteins through laboratory experiments and extensive literature reviews. This can be time-consuming and requires specialized knowledge and resources. The field is progressively adopting computational methods, such as those used in protein structure prediction [5], but there is still a significant and urgent need for tools that can intuitively process and extract valuable insights from protein structures.

This technical report introduces the concept of applying ChatGPT-like capabilities to protein 3D structures, which could be a powerful tool that could greatly advance our understanding of biology and accelerate the development of new treatments for diseases. This AI-based protein query system is capable of understanding protein 3D structures and answering various questions about these proteins, and can be beneficial in several ways:

- **Accelerating Research**. Such a system would make it easier for researchers to find relevant information about specific proteins, their functions, structures, interactions, mutations, and related diseases. This could significantly speed up research in fields such as drug discovery, molecular biology, and genetics.

- **Improving Understanding of Protein Function**. Proteins are the workhorses of the cell, and understanding their function is crucial for understanding how life works at a molecular level. A system that can provide accurate, detailed information about protein function would be a valuable tool for educators and students.
- **Personalized Medicine**. In the field of personalized medicine, understanding the patient's unique protein structures can guide the development of personalized treatments. A system that can quickly and accurately provide this information could revolutionize the way we approach healthcare.
- **Drug Discovery**. The process of drug discovery involves understanding how potential drug molecules interact with target proteins in the body. An AI system for protein queries can help identify potential drug targets and predict the effects of drug candidates.
- **Disease Diagnosis and Treatment**. Many diseases are associated with changes in protein structure or function. Understanding these changes can help in the diagnosis and treatment of diseases. For example, a system like this could potentially provide insights into the protein-related mechanisms behind cancer, neurodegenerative diseases, and other conditions.
- **Data Analysis**. The number of known proteins and the amount of data related to each of them is immense. Manual analysis of this data would be time-consuming and challenging. An AI system could quickly analyze this data and provide useful insights.

Developing a ChatGPT-like system for proteins poses significant challenges that need to be addressed. Firstly, a major hurdle is the encoding of proteins. Unlike sequential and well-structured text data, proteins are complex molecules with diverse structures and functions. Consequently, developing a methodology to represent proteins in a format compatible with GPT models becomes crucial. Secondly, a substantial amount of protein-related information exists in unstructured formats, such as scientific literature. Extracting meaningful insights from such data necessitates sophisticated natural language processing capabilities to effectively handle the complexities and nuances of scientific texts. Thirdly, training such a system requires large datasets encompassing protein structures and their associated literature. However, generating or compiling such datasets is a challenging task due to the proprietary nature of much of this information and the vast diversity within the biological space. Addressing these challenges will be crucial in the development of a successful ChatGPT-like system tailored for proteins. Overcoming the encoding, natural language processing, and dataset limitations will pave the way for an intuitive and comprehensive platform for protein-related research and analysis.

This technical report presents our initial endeavor to empower ChatGPT-like capabilities for three-dimensional protein structures through the development of a prototype system called ProteinChat. Inspired by ChatGPT, ProteinChat operates on the same principles. Users can upload protein 3D structures (e.g., PDB format, widely used for files containing atomic coordinates) and pose diverse questions regarding them. ProteinChat adeptly engages in multi-turn, interactive conversations to provide comprehensive answers to these inquiries. By leveraging ProteinChat, users can gain valuable insights and explore the intricacies of protein structures in an interactive and intuitive manner.

The ProteinChat system comprises three key components: a GVP-Transformer block [4], a projection layer, and a large language model. To begin, the GVP-Transformer block receives the 3D structure of a protein as input and effectively learns a representation that captures translation-invariant and rotation-equivalent features. This enables the system to extract crucial information from the protein structure. The projection layer further encodes the protein embeddings generated by the GVP-Transformer block, aligning them with the requirements of the LLM. Finally, the LLM leverages the composite representation of the encoded protein, along with the users' prompt related to the current protein, to generate insightful and contextually relevant answers. Through the collaborative functioning of these components, ProteinChat enables seamless interactions and facilitates accurate responses based on the protein's structural information and user queries. For our project, we specifically utilize the Vicuna-13b LLM [2] and a pre-trained GVP-Transformer from ESM-IF1 [4] as the protein encoder block. During the training phase, we freeze the protein encoder block and the LLM, solely focusing on training the projection layer. To train the ProteinChat model, we construct the RCSB-PDB Protein Description Dataset, which pairs protein structures with corresponding textual descriptions, from the publicly available RCSB-PDB database.

The major contributions of this technical report are as follows:

- We propose the ProteinChat, a prototype ChatGPT-like model that is capable of engaging Q&A with users and generating text descriptions for proteins based on their 3D structures.
- We introduce the RCSB-PDB Protein Description Dataset. This dataset contains 143,508 protein entries. Each entry consists of a protein 3D structure represented in backbone atomic coordinates and an abstract that describes this protein.
- To the best of our knowledge, ProteinChat is the first work that aims to harness the power of LLM to study proteins. Our work can be easily generated to other domains by changing the encoder and input data.

## 2 Related Works

**Protein Science and Engineering** is increasingly being influenced by computational approaches, especially with artificial intelligence. Large-scale protein structure prediction, as demonstrated by DeepMind's AlphaFold [5], is a prime example of how AI is revolutionizing protein research. Meta proposed Evolutionary Scale Modeling (ESM) [7], including inverse folding (ESM-IF1) [4], as Transformer-based protein language models for structure prediction. The main goal of ESM-IF1 is to predict a protein sequence from its backbone atom coordinates, and in ProteinChat, we leverage the GVP-Transformer of ESM-IF1 to encode proteins from their 3D structures. However, generating or predicting protein properties based on protein 3D structures using LLMs is still a missing area, and our research made preliminary efforts towards bridging this gap by training both on the protein 3D structures and texts.

**Large Language Models (LLMs)** have shown impressive accomplishments in natural language processing tasks [3, 1]. Such models are typically trained on an extensive, diverse corpus, which includes resources like Wikipedia, web texts, and books. Two strategies often adopted to tailor LLMs for a specific task involve either training a new language model on a domain-specific database or adapting general LLMs for a specific task, the latter being our focus. For instance, scientific language models have been created based on a corpus of scientific knowledge. An example is Galactica by Meta AI [8], which explicitly modeled the sequences and SMILES (instead of the 3D structure in our work) of proteins with scientific literature, and make the model explain properties of the sequences. Further, the development of publicly accessible LLMs, including multi-modal variants, has allowed for the understanding of information in other modalities beyond text. We introduce three of them in the following, and our model is built upon these open-source projects.

**LLaMA** [9] is a general LLM model with 65 billion parameters developed by Meta AI. This model is trained on publicly accessible datasets including English CommonCrawl, C4, Github, Wikipedia, Gutenberg Project, ArXiv and Stack Exchange. With a size smaller than GPT-3, LLaMA achieves competitive performance on many tasks, including commonsense reasoning, question answering, reading comprehension and code generation. **Vicuna** [2] is an open-source chatbot trained by fine-tuning LLaMA on around 70,000 user-shared conversations with ChatGPT,with training costs as low as around $300.

**MiniGPT-4** [11] is an vision-language model that can process images and generate text interactively.It is built upon Vicuna and Lavis [6] for language and vision tasks. Users can upload their own images and then chat with the model based on those images. The architecture of MiniGPT-4 integrates a single linear projection layer, designed to capture the visual information from a pretrained vision encoder. Vicuna serves as the language decoder in this setup, while a ViT backbone operates as the visual encoder.

## 3 Method

### 3.1 Data Collection

The three-dimensional structures of proteins were sourced from the Research Collaboratory for Structural Bioinformatics Protein Data Bank (RCSB PDB) [1], which includes 204,826 experimentally determined 3D structures. We utilized the data version last updated on May 17, 2023. Of these proteins, 163,635 have a primary publication linked with a PubMed ID. Among those, the PDB

[1] `https://www.rcsb.org/`

format of 143,508 proteins can be extracted with a valid chain using Biotite, a Python library for computational molecular biology [2], and these constituted the main focus of our study. For every protein taken into account, we compiled its 3D atomic coordinates, representing the molecular structure, along with the abstract of corresponding scientific literature to train our model. Table 3 shows some examples of the abstract data.

We also obtained various protein properties in multiple levels from Protein Data Bank. Entry level properties contain molecular weight and count of atoms. Entity level properties include the number of entities and their classification as polymer, non-polymer, branched molecular, or solvent entities. We also collected the statistics in assembly and instance level. Additionally, we collected protein-specific experimental details, such as whether the structure was determined using experimental or AlphaFold, and the methods of experiments. This information is directly requested from Protein Data Bank using its web service APIs. Based on these properties, we manually designed 30 question-answer pairs, asking about numbers, true/false questions, categories, and explanations. An example of Q&A for a specific protein is displayed in Table 2. Note that there are some missing values in the original data corpus, so the designed questions would be slightly different for each protein.

Table 1: Dataset statistics.

| Data Format | Number of Entries | Total Size |
|---|---|---|
| Protein PDB files | 143,508 | 121GB |
| Encoded PT files | | 83GB |
| Abstract TXT files | | 182MB |

Table 2: An example of designed Q&A for a specific protein from the Protein Data Bank.

| Protein PDB | 1KIT.pdb |
|---|---|
| QA Pair 1 | **Q:** How many assemblies does this protein have?<br>**A:** 1. |
| QA Pair 2 | **Q:** How many entities does this protein have?<br>**A:** 3. |
| QA Pair 3 | **Q:** Does this protein contain non-polymer entities?<br>**A:** Yes. |
| QA Pair 4 | **Q:** Does this protein contain polymer entities?<br>**A:** Yes. |
| QA Pair 5 | **Q:** Does this protein contain DNA polymer entities?<br>**A:** No. |
| QA Pair 6 | **Q:** Does this protein contain RNA polymer entities?<br>**A:** No. |
| QA Pair 7 | **Q:** Does this protein contain solvent entities?<br>**A:** Yes. |
| QA Pair 8 | **Q:** Does this protein contain branched entities?<br>**A:** No. |
| QA Pair 9 | **Q:** What is the polymer entity composition for this protein? Please choose from DNA, DNA/RNA, NA-hybrid, NA/oligosaccharide, RNA, heteromeric protein, homomeric protein, oligosaccharide, other type composition, other type pair, protein/NA, protein/NA/oligosaccharide, protein/oligosaccharide and other.<br>**A:** Homomeric protein. |
| QA Pair 10 | **Q:** What is the nucleic acid polymer entity type for this protein? Please choose from DNA (only), DNA/RNA (only), NA-hybrid (only), RNA (only) and others.<br>**A:** Other. |

[2] `https://github.com/biotite-dev/biotite`

| | |
|---|---|
| QA Pair 11 | **Q:** What are the bound nonpolymer components for this protein?<br>**A:** CA. |
| QA Pair 12 | **Q:** What experimental methods were used to determine the structure of this protein? Please choose from EM, Multiple methods, NMR, Neutron, X-ray and other.<br>**A:** X-ray. |
| QA Pair 13 | **Q:** What is the molecular mass (KDa) of polymer and non-polymer entities (exclusive of solvent) for this protein?<br>**A:** 83.12. |
| QA Pair 14 | **Q:** What's the polymer entity type for this protein? Choose from nucleic acid (only), oligosaccharide (only), protein (only), protein/NA, protein/oligosaccharide and other.<br>**A:** Protein (only). |
| QA Pair 15 | **Q:** What are the software programs reported in connection with this protein?<br>**A:** DENZO, TNT. |
| QA Pair 16 | **Q:** Is this protein determined by experimental or computational methods?<br>**A:** Experimental methods. |
| QA Pair 17 | **Q:** How many heavy atom coordinates records does this protein have?<br>**A:** 6560. |
| QA Pair 18 | **Q:** How many hydrogen atom coordinates records does this protein have?<br>**A:** 0. |
| QA Pair 19 | **Q:** How many model structures deposited for this protein?<br>**A:** 1. |
| QA Pair 20 | **Q:** How many polymer monomers does this protein have?<br>**A:** 757. |
| QA Pair 21 | **Q:** How many heavy solvent atom coordinates records does this protein have?<br>**A:** 699. |
| QA Pair 22 | **Q:** How many nucleic acid polymer entities (DNA or RNA) does this protein have?<br>**A:** 0. |
| QA Pair 23 | **Q:** Does this protein have unmodeled polymer monomers?<br>**A:** No. |
| QA Pair 24 | **Q:** Does this protein have hybrid nucleic acid polymer entities?<br>**A:** No. |
| QA Pair 25 | **Q:** When is this protein first published?<br>**A:** 1994. |
| QA Pair 26 | **Q:** What is the terms characterizing the protein?<br>**A:** Hydrolase. |
| QA Pair 27 | **Q:** What is the radiation wavelength in angstroms for this protein?<br>**A:** 1.5418. |
| QA Pair 28 | **Q:** How many intermolecular covalent bonds does this protein have?<br>**A:** 0. |
| QA Pair 29 | **Q:** How many intermolecular metalic bonds does this protein have?<br>**A:** 13. |
| QA Pair 30 | **Q:** Does this protein have cis-peptide linkages?<br>**A:** Yes. |

### 3.2 Model Architecture

ProteinChat consists of a composite encoder block and an LLM decoder block, working synergistically to provide protein-related insights. The composite encoder block combines a graph neural network (GNN) encoder block with a Transformer encoder [10] block. In our implementation, we leverage a pre-trained GVP-Transformer [4] as the composite encoder block, which effectively captures essential features from protein structures. As for the LLM decoder, we employ Vicuna-13B. To bridge the gap between the protein embedding generated by the encoder block and the LLM, we introduce a projection layer. This layer is trained using our dedicated protein dataset, facilitating seamless

Table 3: Examples of the abstract of the primary publication for a specific protein, identified by the protein entry id in Protein Data Bank.

| Protein ID | Abstract of the Primary Publication |
|---|---|
| 2KI9 | We detail the structure and dynamics of a synthetic peptide corresponding to transmembrane helix 6 (TMH6) of human cannabinoid receptor-2 (hCB2) in biomembrane-mimetic environments. The peptide's NMR structural biology is characterized by two alpha-helical domains bridged by a flexible, nonhelical hinge region containing a highly-conserved CWFP motif with an environmentally sensitive, Pro-based conformational switch. Buried within the peptide's flexible region, W(258) may hydrogen-bond with L(255) to help stabilize the Pro-kinked hCB2 TMH6 structure and position C(257) advantageously for interaction with agonist ligands. These characteristics of hCB2 TMH6 are potential structural features of ligand-induced hCB2 activation in vivo. |
| 9INS | Two localized monovalent cation binding sites have been identified in cubic insulin from 2.8 A-resolution difference electron density maps comparing crystals in which the Na+ ions have been replaced by Tl+. One cation is buried in a closed cavity between insulin dimers and is stabilized by interaction with protein carbonyl dipoles in two juxtaposed alternate positions related by the crystal dyad. The second cation binding site, which also involves ligation with carbonyl dipoles, is competitively occupied by one position of two alternate His B10 side chain conformations. The cation occupancy in both sites depends on the net charge on the protein which was varied by equilibrating crystals in the pH range 7-10. Detailed structures of the cation binding sites were inferred from the refined 2-A resolution map of the sodium-insulin crystal at pH 9. At pH 9, the localized monovalent cations account for less than one of the three to four positive counterion charges necessary to neutralize the negative charge on each protein molecule. The majority of the monovalent counterions are too mobile to show up in the electron density maps calculated using data only at resolution higher than 10 A. Monovalent cations of ionic radius less than 1.5 A are required for crystal stability. Replacing Na+ with Cs+, Mg++, Ca++ or La+++ disrupts the lattice order, but crystals at pH 9 with 0.1 M Li+, K+, NH4+, Rb+ or Tl+ diffract to at least 2.8 A resolution. |
| 2M7P | The peptide hormone relaxin is showing potential as a treatment for acute heart failure. Although it is known that relaxin mediates its actions through the G protein-coupled receptor relaxin family peptide receptor 1 (RXFP1), little is known about the molecular mechanisms by which relaxin binding results in receptor activation. Previous studies have highlighted that the unique N-terminal low density lipoprotein class A (LDLa) module of RXFP1 is essential for receptor activation, and it has been hypothesized that this module is the true "ligand" of the receptor that directs the conformational changes necessary for G protein coupling. In this study, we confirmed that an RXFP1 receptor lacking the LDLa module binds ligand normally but cannot signal through any characterized G protein-coupled receptor signaling pathway. Furthermore, we comprehensively examined the contributions of amino acids in the LDLa module to RXFP1 activity using both gain-of-function and loss-of-function mutational analysis together with NMR structural analysis of recombinant LDLa modules. Gain-of-function studies with an inactive RXFP1 chimera containing the LDLa module of the human LDL receptor (LB2) demonstrated two key N-terminal regions of the module that were able to rescue receptor signaling. Loss-of-function mutations of residues in these regions demonstrated that Leu-7, Tyr-9, and Lys-17 all contributed to the ability of the LDLa module to drive receptor activation, and judicious amino acid substitutions suggested this involves hydrophobic interactions. Our results demonstrate that these key residues contribute to interactions driving the active receptor conformation, providing further evidence of a unique mode of G protein-coupled receptor activation. |

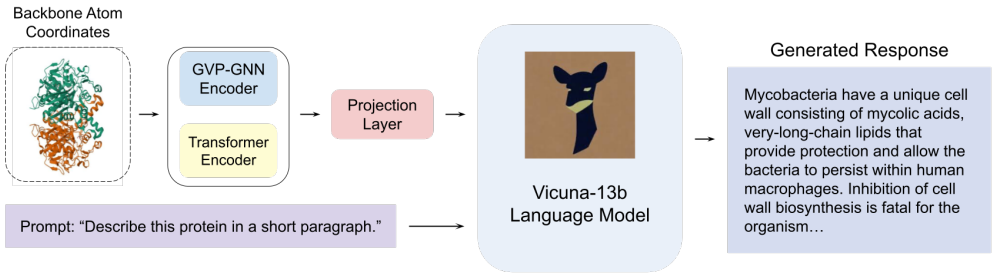


Figure 1: An overview of the ProteinChat framework.

integration between the encoded protein representation and the LLM. The output of the projection layer serves as soft prompts, and we design auxiliary prompts to guide the LLM in generating textual descriptions for the input protein. For our training setup, we freeze both the composite encoder block and the LLM decoder, solely focusing on training the projection layer. This strategy ensures that the pre-trained models retain their learned knowledge while fine-tuning the projection layer aligns the protein embeddings with the LLM's requirements.

### 3.2.1 Composite Encoder Block

**GVP-GNN**. We use Geometric Vector Perceptron (GVP) to represent protein graphs and GNN to perform graph embedding. Protein 3D backbone atoms are represented as proximity graphs and each amino acid corresponds to a node in the graph. Scalar-valued note features $s$ and vector-valued note features $V$ are generated. Simply put, GVP learns geometric vectors and scalars using two separate linear transformations, followed by nonlinear layers $\sigma, \sigma^+$. To achieve rotation-invariant property, additional normalization and re-scaling are utilized. The features are updated as follows:

$$s \leftarrow \sigma(W_m(l_{L_2}(V), s) + b), \tag{1}$$

$$V \leftarrow \sigma^+(l'_{L_2}(V)) \odot l'(V). \text{ (row-wise multiplication)} \tag{2}$$

Here $l$ and $l'$ are linear layers, and $L_2$ means applying row-wise L2 norm. With protein graphs defined above, GNN will update the representation of each node by aggregating features from its neighboring nodes. Therefore, after $k$ rounds, the embedding $h_v^k$ of node $v$ contains information on its k-hop neighbors. Formally, GNN performs the following updates:

$$h_v^k = g(h_v^{k-1}, AGG(\{h_u^{k-1}, u \in \mathcal{N}(v)\})). \tag{3}$$

Here, $AGG$ represents a function that aggregates information from neighboring nodes, and $g$ is a function that combines information from the node feature of the previous layer and neighboring nodes. $\mathcal{N}(v)$ indicates all neighboring nodes for node $v$. To acquire the representation vector for the entire graph $G$, a pooling function $f$ extracts information from all node features at the final round $K$:

$$h_G = f(\{h_v^K, v \in G\}), \tag{4}$$

where $f$ represents a permutation-invariant function, such as averaging.

**Transformer Block**. We employ 8 Transformer encoding blocks [10] to encode the graph embedding from GVP-GNN to form a composite embedding as inputs to the LLM.

### 3.2.2 Large Language Models (LLMs)

Recent LLMs typically employ the Transformer decoder [10] to model the conditional generation probability $p_\theta(n_i|n_{<i})$ in the language model. This involves applying a multi-head self-attention block to the input context tokens and using a position-wise feedforward network to calculate the generative distribution over output tokens. Given the context vector of tokens, the token generation probability is computed as follows:

$$\begin{aligned} h_0 &= N_{i-1}W_e + W_p \\ h_l &= transformer\_block(h_{l-1})(1 \le l \le m) \\ p(n_i|N_{i-1}) &= Softmax(h_m W_e^T) \end{aligned} \tag{5}$$

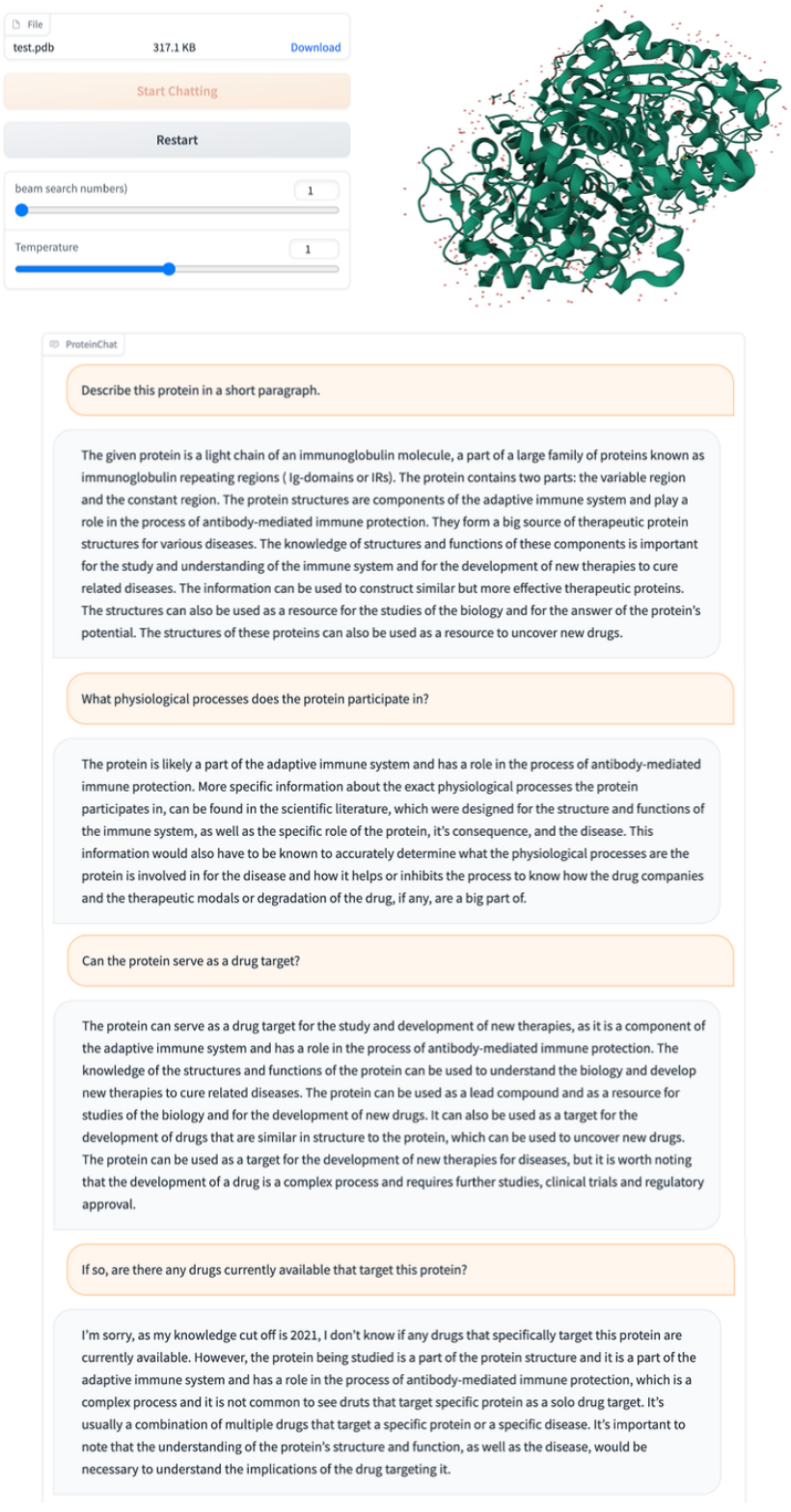


Figure 2: A demonstration of the conversation between human and the ProteinChat

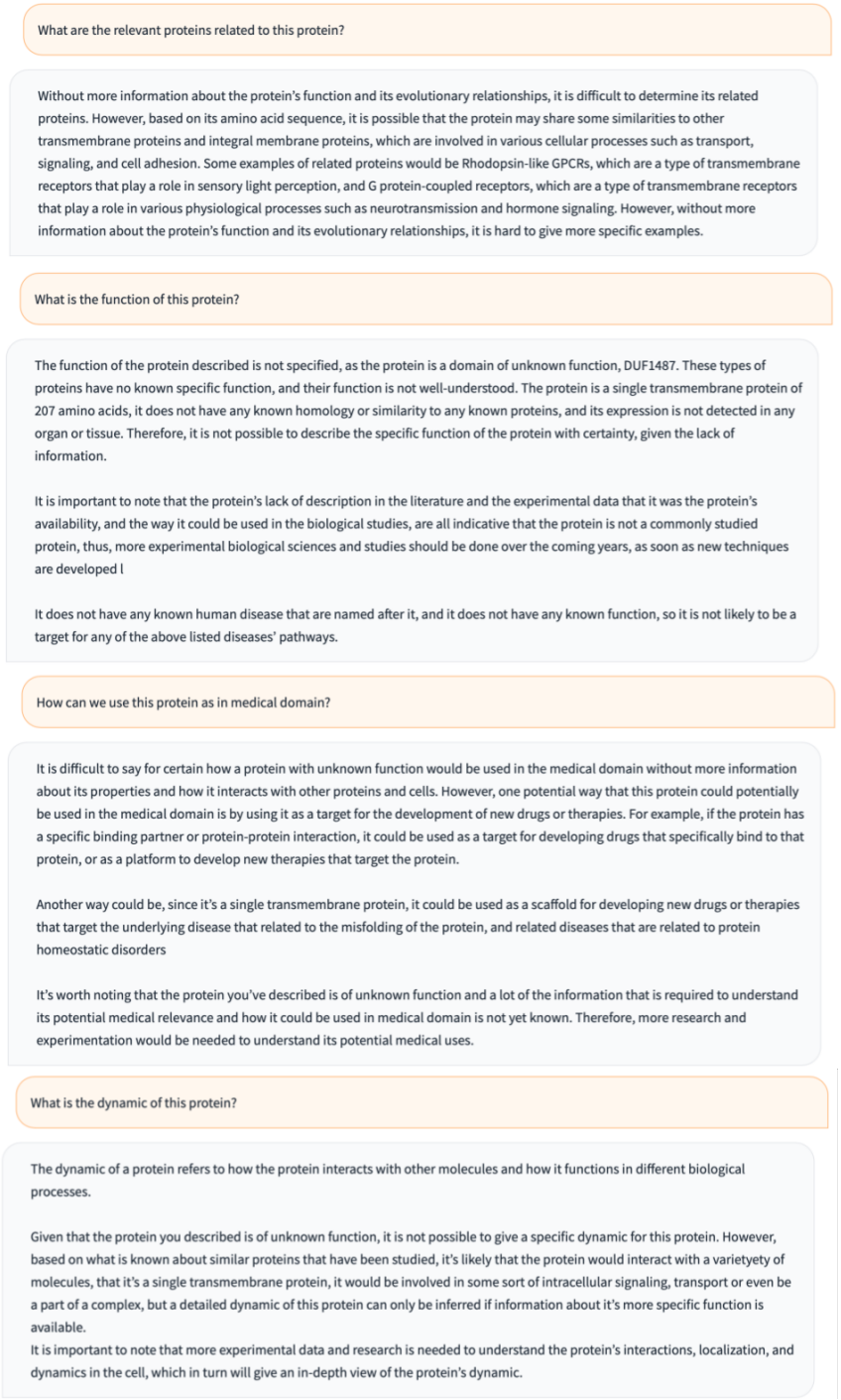


Figure 3: A demonstration of the conversation between human and the ProteinChat

where $N_{i-1} = (n_1, n_2, ..., n_{i-1})$ denotes the context vector of tokens, $m$ refers to the number of layers, $W_e$ denotes the token embedding matrix, and $W_p$ represents the positional embedding matrix.

### 3.2.3 Aligned Protein-Text Generation

To further assist the training of the model with protein-text pairs, we leverage the special token prompt following the conversational format of Vicuna-13b:

$$\begin{aligned} &\textbf{Q:} < Protein >< ProteinStructure >< /Protein >< Instruction > \\ &\textbf{A:} < Desc > \end{aligned} \tag{6}$$

In this prompt, $< ProteinStructure >$ is a soft prompt that symbolizes the protein 3D structure encoded by the linear projection layer. $< Instruction >$ serves as a directive sentence, prompting the LLM to generate descriptions for the protein, such as "Introduce the protein in a short paragraph." During the training stage, $< Desc >$ is replaced with descriptive text from publicly available description of the protein to train the linear projection layer. In the testing stage, $< Desc >$ remains empty, and the model is expected to generate descriptive text for the given protein structure.

## 4 Results

Figure 2 presents example conversations between the user and ProteinChat on a protein that does not appear in the training set of ProteinChat. It can be seen that ProteinChat possesses the ability to answer various questions that are not covered during the training process, such as "what physiological processes does the protein participate in?", "can the protein serve as a drug target?", "what are the relevant proteins related to this protein?", with coherent and human understandable explanations. We will perform a systematic quantitative evaluation by collaborating with biologists.

## 5 Limitations and Future Work

While ProteinChat harnesses the impressive capabilities of LLMs, it is essential to address an important limitation related to potential language hallucination. The model has the potential to generate confident yet unjustified protein descriptions, which could misguide researchers in their investigations. To overcome this challenge, it becomes crucial to train the model using a larger corpus of high-quality aligned protein-text pairs. Rigorous data preprocessing strategies should be implemented to ensure the validity and reliability of the training data. Furthermore, incorporating feedback from domain experts is crucial for improving the system. By introducing reinforcement learning techniques, ProteinChat could learn from expert guidance and iteratively refine its responses over time. This iterative learning process would enhance the accuracy and reliability of the system's predictions. In future work, it is imperative to address these limitations and focus on refining ProteinChat to enhance its reliability and accuracy. This includes not only expanding the training dataset but also incorporating mechanisms to validate the generated protein descriptions. By continuously refining the system and actively seeking input from experts, ProteinChat can evolve into an even more valuable tool for protein research, providing reliable and insightful information to support scientific investigations.

## 6 Conclusions

In this study, we introduce the ProteinChat framework, a ChatGPT-like system to respond to queries and generate textual explanations of proteins based on their 3D structures. In addition, we construct and curate 143K protein-text pairs from publicly available RCSB-PDB dataset with carefully designed Q&A pairs. ProteinChat bridges the gap between multi-modal large language models (LLMs) and protein studies. By leveraging the power of LLMs, ProteinChat has the potential to empower researchers to delve into the complexities of proteins, uncover hidden insights, and gain a deeper understanding of their structures and functions.

## References

[1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

[2] W.-L. Chiang, Z. Li, Z. Lin, Y. Sheng, Z. Wu, H. Zhang, L. Zheng, S. Zhuang, Y. Zhuang, J. E. Gonzalez, I. Stoica, and E. P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] C. Hsu, R. Verkuil, J. Liu, Z. Lin, B. Hie, T. Sercu, A. Lerer, and A. Rives. Learning inverse folding from millions of predicted structures. *ICML*, 2022.

[5] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

[6] D. Li, J. Li, H. Le, G. Wang, S. Savarese, and S. C. H. Hoi. Lavis: A library for language-vision intelligence, 2022.

[7] A. Rives, J. Meier, T. Sercu, S. Goyal, Z. Lin, J. Liu, D. Guo, M. Ott, C. L. Zitnick, J. Ma, and R. Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *PNAS*, 2019.

[8] R. Taylor, M. Kardas, G. Cucurull, T. Scialom, A. Hartshorn, E. Saravia, A. Poulton, V. Kerkez, and R. Stojnic. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*, 2022.

[9] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[10] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *NIPS*, 2017.

[11] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.