Grenoble-INP-ENSIMAG

NATIONAL SCHOOL OF COMPUTER SCIENCE AND APPLIED MATHEMATICS OF GRENOBLE, FRANCE

INTERNSHIP REPORT ASSISTANT ENGINEER

CONDUCTED AT

IBM RESEARCH LAB ZÜRICH, SWITZERLAND

Language Modeling on Enzyme-Catalyzed Reactions Unsupervisedly Recover Active Sites

---

KWATE DASSI LOIC

2ND YEAR, INFORMATION SYSTEM ENGINEERING, GRENOBLE-INP-ENSIMAG

IBM Research Lab Zürich,
Switzerland
Säumerstrasse 4
8803 Rüschlikon

Academic Supervisor:
DERBAL Moufida

Supervisors:
Dr. Teodoro LAINO
Dr. Matteo MANICA

March 26, 2022

NATIONAL SCHOOL OF COMPUTER SCIENCE AND APPLIED MATHEMATICS OF GRENOBLE, FRANCE

Abstract

Grenoble-INP
ENSIMAG

IBM Research Lab Zürich, Switzerland

Master's Degree in Information System Engineering

Language Modeling on Enzyme-Catalyzed Reactions Unsupervisedly Recover Active Sites

by KWATE DASSI Loic

We present in this work a Transformer-based language model adapted on enzymatic bio-catalyzed reactions which independently captures the knowledge of the functional sites of the catalyst of the reaction. Training language modeling on reaction SMILES complemented with amino acid sequence information let us define a score based on the attention values to detect regions of the enzyme that matches the active site. This work is relevant for AI explainability and paves the way for an extensive application of language models for enzyme design and reaction optimization in bio-catalyzed chemical processes. We leverage the expressiveness of the Transformer-based models to understand the substrate-enzyme interactions in the enzyme-catalyzed reactions by training the BERT model with the self-supervised learning tasks Masked Language Modeling (MLM) and n-gram MLM on bio-catalyzed reactions and analyzing the attention matrix of the embedded reactions considered as the representation of the mapping graph between the reactants and enzymes. The unsupervised active site detection road map enabled us to recover 31.51% of experimental active regions in accordance with the Protein-Ligand Profiler Interaction (PLIP) and 67.77% of active sites according to the Protein Database Family Pfam.

Keywords: Enzyme, Catalyst, Transformer, SMILES, Protein, Ligand, Language Model, Active Site, BERT, PLIP, Pfam, Residue, PDB, Substrate, Reactant, Product, Docking

Contents

List of Figures

List of Tables

List of Abbreviations

BERT	Bidirectional Encoder Representations from Transformers
NLP	Natural Language Processing
NLU	Natural Language Understanding
MLM	Masked Language Modeling
SMILES	Simplified Molecular Input Line Entry System
PLIP	Protein Ligand Interaction Profiler
PFAM	Protein Families Database
BPE	Byte Pair Encoding
BOW	Bag of Words
LSTM	Long-Short Term Memory

I Context and Problematic

Enzyme engineering is attracting considerable attention on an account of the fact that enzymes are involved in almost all the natural and synthesized chemical reactions, especially in bio-catalyzed reactions. In bio-catalytic processes, natural catalysts, such as enzymes, are used to speed up chemical reactions on organic compounds with the least admissible activation energy. A significant part of enzymes is proteins. Hence, the approaches used in protein engineering –Functional understanding of proteins, protein folding, and protein design– can be tailored to address problems encountered in enzyme engineering. Protein enzymes, when used towards the acceleration of organic reactions, have to be bound with the organic compounds in specific locations on the protein, these regions are entitled to the active site of the protein. It is therefore clear that the functional understanding of the protein is a crucial part of the way to facilitate the interaction between organic elements and enzymes. The two main representations of protein widely used are the amino acid sequence, and the three-dimensional structure –Protein Data Bank (PDB)– (Berman, Henrick, and Nakamura, 2003). The functional analysis of protein remains a great problem in life science due to several which the main are the following: the average length of protein and protein folding. Because proteins are generally lengthy, it is indeed cumbersome to proceed with them by hand and the computational treatment is very expensive. The functional specificity of protein is heavily tied to its three-dimensional structure. Thus, use the PDB structure in the analysis is, of course, the right way to go. But due to the shortage of 3D structures and the difficulty of the protein folding, analyze the 1D dimensional structure, namely the amino acid sequence, enlighten other approaches of the protein understanding solely based on sequence information. Organic elements, generally the reactants and the products in bio-catalyzed reactions, are by and large represented in SMILES (Weininger, 1988) strings format.

Considering the sequence representation of the reactants, the enzymes, and the products as a language of the description of the chemical processes under the hood, we can indeed, adapt the current methods used in natural language understanding to address the active site recognition. The last decade has witnessed a surge of compelling architectures and paradigms of the artificial processing of natural language. The Transformer-based models, thanks to their expressiveness, achieve the state-of-the-art in many tasks on GLUE (Wang et al., 2019).

In line with this, the main problem that will lay the groundwork is: how to tune the natural language understanding methods to break down the chemical grammar in bio-catalyzed reactions, and therefore, draw out the active regions of proteins?

II Host Institution and Course Expectations

The welcome Institution of our course is IBM Research Lab, Zurich, Switzerland, led by Dr. Alessandro Curioni. IBM Research Zurich is leading several domains and provide services in eras where the major are the following: Quantum Computing, Hybrid Cloud Solution,

Cyber Security, Accelerated Discovery. We worked in the Accelerated Discovery Department conducted by Dr. Teodoro Laino and under the supervision of Dr. Matteo Manica, a research staff member in the Accelerated Discovery Group. As the AI positively impacts chemistry, one of the main goals of the Accelerated Discovery group is to increase productivity of research and development processes and foster the construction of long-term solution.

As it was stated in the internship description, the assignments of the course are the following:

- In this internship, the student will work on extending the models for forward and backward reaction prediction on SMILES to additional chemical representations/ languages including macromolecules, such as, polymers (Lin et al., 2019) and proteins (Filipavicius et al., 2020). This will be achieved by expanding the existing transformer architectures to handle multiple reaction classes and text-based chemical descriptors using information contained in public databases.
- The candidate will work at all the pipeline components: from data collection and preparation to model training and inference/deployment
- A successful internship will result in a short publication (workshop paper or research report) on a multilingual chemical model, that will represent a milestone of capital importance in the application of deep learning to scientific discovery.

III Introduction

Considerable attention has been focused on ligand-protein enzyme interaction prediction due to the wide range of applications it induces such as drug repurposing, drug discovery, molecular docking, and so forth. In bio-catalyzed chemical processes, it is essential to understand how substrates interact with the enzyme to optimize the necessary energy for that reactions to happen. A successful accomplishment of these downstream tasks required an understanding of how small molecules interact with proteins and where that reactions happen, that is, the active domains of the protein where the inter-linkages are effectively carried out. Likewise the rise of interest in protein active site recognition, the last five years have witnessed a breakthrough in the development of methods to understand the human natural language. Particularly, the advent of the Transformer (Ashish et al., 2017) marked a significant milestone in the development of the computational methods for language processing. The Transformer-based models developed to date progressively provide a meaningful contextual mapping of the encoded sentence and in many cases yield a good performance on downstream tasks built on top of that representation. In this work, we shaped the active site detection as a language-based task thus took advantage of the compelling Transformer-based model to provide an insightful latent representation of the enzymatic-catalyzed reaction in terms of mapping between the reactants and enzymes.

To learn the language of the enzymatic chemical processes, the underlying grammar that withstands the interactivity between reactant and enzymes, we trained the transformer-based models Albert (Zhenzhong et al., 2019) and BERT (Devlin et al., 2019) with the self-supervised learning task masked language modeling (MLM) on the mixture of bio-catalyzed and organic reactions. After the MLM training, the attention matrix known in this case as the mapping representation between the reactant and the enzyme was analyzed to extract the relevant inter-linkages regarding their attention weights. The characteristics of language modeling are not well understood and language modeling has not been dealt with in-depth in chemistry. Despite the interest in protein active domain prediction, no one as far as we know has studied yet the ability of language models to grasp the chemistry grammar of the bio-catalyzed reactions. We evaluated the ability of the trained models to recover without supervision, the active regions of protein while merely trained on chemical reactions. The evaluation was carried out on two main data sources. The first concerns the protein annotations from Pfam (Mistry et al., 2020), here we evaluated the capacity of the models to recognize the annotated protein active site. The second concerns the ligand-protein interactions of co-crystallized complexes given by PLIP (Adasme et al., 2021), we assessed the aptness of the models to capture the three-dimensional inter-activities between the ligand and protein with only one-dimensional data namely the SMILES representation of the reactants and the amino acid sequence of the enzyme. Our work aims to extend the current knowledge of language models to the understanding of the chemistry grammar under the hood, the main contributions of this work are the following :

- Outlined a new language model-based unsupervised approach for the protein active site recognition.
- Provided a bio-catalyzed reaction knowledge encoder for downstream tasks.

The outline of this is the following: in the next section we review the literature in active site detection, followed by the methodology, the experiments led and the results obtained, the interpretation of these results, and a possible outlook of the active site recognition.

IV Related Work

There is a considerable amount of literature on data-driven protein-ligand binding site prediction, protein engineering, and molecular mapping in chemical reactions.

Protein-Ligand Binding Site Prediction. Identify the binding sites on proteins is important for a functional understanding of those proteins. Most of the enzymes are proteins; To optimize the level of the bio-catalyzed reaction energy for a more stable reaction, recognize the active region on protein becomes a crucial part of the quest. (Liang et al., 2006; Krivák and Hoksza, 2015) introduced a binding site prediction method using a scoring function that assigns a ligandability score to the binding pockets, then predicts the active regions based on the assigned scores. The surge of Machine Learning, in particular the Convolutional Neural Network, has not left biochemistry unarmed. Indeed, several works using the 3D feature analysis of the protein were conducted to identify the druggable regions. (Stepniewska-Dziubinska, Zielenkiewicz, and Siedlecki, 2020; Aggarwal et al., 2021; Kozlovskii and Popov, 2020; Simonovsky and Meyers, 2020) used a fully 3D convolutional

neural network to analyze the three-dimensional feature of the protein-ligand complex then extract the binding pockets. Besides the three-dimensional structure analysis of protein, sequence-based functional assessment is also holding information on the amino acid alignment. (Harrison et al., 2015) introduced a Naïves Bayes Classifier based on the sequence information of the protein to predict whether each amino acid residue is active.

Protein Engineering. Several approaches based on the protein structure information have been developed to apprehend the functional properties of proteins. Likewise, the Protein-Ligand interaction understanding with the convolutional neural network, (Gligorijević et al., 2021) used a graph convolutional neural network along with a sequence representation from a pre-trained task-agnostic language modeling for the amino acid residue annotation. Many protein sequences do not have a corresponding three-dimensional representation; the UniProt database (Consortium, 2020) contains about 200M sequence entries. In another hand, the Protein Data Bank contains only about 68K entries. That is, only 0.03% of proteins are annotated. In line with this, understand the mapping between the sequence representation and the 3D structure of proteins became crucial to have a full comprehension of their functional features. (Ingraham et al., 2019) led a protein design task by lining up a graph-based model followed by a Transformer-based generative model. Proteins can be viewed as a sequence of amino acid residues. Hence, the methods developed to process the natural language can be readily used to understand the underpinning morphology of proteins. (Brandes et al., 2021; Elnaggar et al., 2020) tailored the masked language modeling self-supervised training of variant Transformer-based model on protein sequences, therefore, provided pre-trained protein encoders for downstream tasks. As the set of proteins is growing without a consistent annotation alongside, (Madani et al., 2020) leveraged the language modeling ability of Transformer-based models to generate protein based on taxonomy, molecular function, and cellular component.

V Fundamentals in Natural Language Understanding

The work presented here lies at the junction of the natural language understanding (NLU) and chemistry. Before diving straight in the heart of the work done, it is worth to introduce the knowledge of natural language understanding that withstands the results obtained. NLU encloses all the set of methods developed to artificially process natural language – text, voice – and understand the logic and the information embedded according to the end business goal. In this section, we will present the preliminaries ranging from the tokenization, embedding methods to attention-based models and the contextual mapping.

V.1 Tokenization

The natural language is the most widely used means of communication, it appears to be intrinsically rooted in the human thought such that we, as humans, do not even realize how we proceed and understand information conveyed by the natural language. Having a complexity ranges from the syntactic to the semantic level, different methods have been undertaken to crumble the text into understandable units. The tokenization is the process by which a bulk text or voice is broken into pieces so-called tokens (Guo, 1997). Depending on

the understanding level at which we operate, we have different ways to tokenize corpora. In the next sections, we will progressively present the character-level, the word-level, the hybrid-level, and gradient-based tokenization.

Character-Level Tokenization

Tokenize a given corpus at the character-level is the most easiest way to break it down. Each language leans on its alphabet which is the set of the letter – refers to the smallest language unit – used to build up the words of that language. The units of the alphabet are often called bytes, from which the character-level gets its name of byte-level tokenization. The vocabulary size of the that setting is the size of the alphabet units, which are relatively small in general, that's, the size do not exceed 256 (the Khmer alphabet, from Cambodian, is the longest alphabet in the world with 74 letters). Several NLP researches, operating at the character level have been done including self-attention-based language modeling (Al-Rfou et al., 2018), character-level representation for semantic parsing (Noord, Toral, and Bos, 2020). The firgure 1 presents a use case of this type of tokenization, each letter is fed to model in order to been processed according a logical business. Whilst maintaining a small vocabulary size, this tokenization suffers from the lack of the meaningful morphology and more often requires a neural network deep enough to grasp the meaning of the sequence provided.

FIGURE 1: **Character-level tokenization**

Word-Level Tokenization

According to the Cambridge dictionary, the word is a combination of letters with a meaning, it is a single unit of language that has a meaning and can be spoken of written [1]. Each language holds a dictionary containing the spoken and written words of that language. Given a language subject to NLP-based research, the corresponding dictionary is generally added a *UNK* word which refers to the unknown words –based on what is in the dictionary– that will be encountered during the pro-processing of texts. The word-level tokenization mainly consists in splitting a given corpus into a list of words on which it is made of, preserving the orders of that words as it is presented in the figure 2.

Sub-Word Level Tokenization

The inconveniences of the character and word-level splitting fostered the conception of hybrid-level tokenization. First, The character-level tokenization suffers from the lack of

[1] https://dictionary.cambridge.org/dictionary/english/word

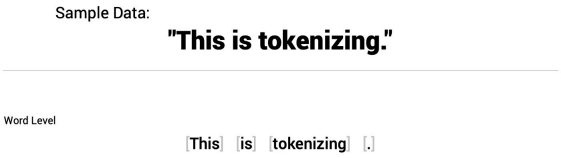


FIGURE 2: **World-Level Tokenization**

meaningfulness of the units issued after tokenizing, that is, the resulting letters of the splitting do not individually bear enough information on the syntactic structure and the semantic meaning of the whole tokenized corpus. This drawback consequently burdens the processing of the corpus and requires very deep or strong architectures. Secondly, the word-level, whilst splitting texts based on the space separating the words, does not take into account the underlying syntactic structure of words which sometimes gives credit to their meaning. Toward addressing the foregoing drawbacks, (Gage, 1994) introduced a data-drive compression algorithm, Byte Pair Encoding (BPE), that statistically builds a sub-word vocabulary based on the appearance frequency of the sub-words on the corpus on which the tokenizer is trained. This hybrid tokenization will be used in our work to build up the vocabulary language of proteins. The BPE is an iterative algorithm that creates the sub-word vocabulary by joining tokens pair by pair based on their frequency –which must be greater or equal than a given threshold– until no fusion is possible under the frequency-based conditions. A succinct presentation of the BPE algorithm can be found on that blog [2] and concrete example is presented in the figure 3

FIGURE 3: **Sub-Word Level Tokenization**

V.2 Word Embedding

Once tokenized a given corpus, the result has to be transformed in a format so that a neural network can understand its meaning and process it accordingly. A wide range of machine learning methods developed so far support only numerical computations, hence only process numerical vectors. It, therefore, becomes mandatory to find a way to map the tokens to a vector representation. The word embedding is a mapping between token and vector representation. The mapping approaches developed to date fall into the following classes: static and dynamic word embedding.

[2] http://www.pennelynn.com/Documents/CUJ/HTML/94HTML/19940045.HTM

Static Word Embedding

The simplest way to perform the word embedding is to build a look-up table which contains the tokens of the vocabulary associated with a fixed vector representation –trainable or not– then, use that mapping table to transform a given corpus into matrix (the collection of the row-vector or column-vector representations of the corpus's tokens). The most pre-eminent static word-embedding used are the following: the singular value decomposition (SVD), the skip-gram and bag-of-word (BOW) model, Glove.

The SVD word embedding involves two processes. The first step consists in constructing the co-occurrence matrix. The second step concerns the singular value decomposition of the the co-occurrence matrix from which the word-embedding mapping will be drawn. Let consider V, $|V|$ as the vocabulary holding n tokens. The co-occurrence matrix $L \in \mathbb{R}^{n \times n}$ is matrix such that $L_{i,j}$ is the number of times the token pair (i, j) appears in the large corpora on which the search pair algorithm will iterate through. Once the L matrix built, it is then decomposed using the SVD. So, we have following representation $L = U\Sigma V^T$. Based on that decomposition several approaches can be used to build up the word-embedding, the simplest way is to add up the matrix U and V^T

The Skip-gram and Bag-of-word models are trainable construction in which the word-embedding mapping matrix is optimized to a given task so-called mask language modeling (MLM). Given a corpus, the MLM is a task which consists in, firstly, randomly masking some tokens in the corpus and secondly, predict that by forward pass through as model or predictor. The skip-gram consists in the prediction of the context of a given word, that is, the masked words surrounding that word, whereas the BOW consists in leveraging the context to predict a target word. The figure 4 summarizes the two definition introduced.

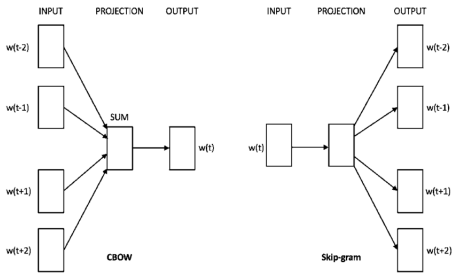


FIGURE 4: **Skip-Gram and Bag of Word Model**
Source [3]

Formally, Consider a vocabulary of size V, $W \in \mathbb{R}^{V \times d}$ the word-embedding matrix with d as the hidden dimension of vector representation of each vocabulary token, $2p + 1$ as the window size of the context in which the token will be masked to be predicted.

Given $S = \{t_i\}_{i=1}^{n}$, a sequence of tokens, k, a randomly selected token in the sequence S, $f : \mathbb{R}^{2p} \to \mathbb{R}^{V}$ and $g : \mathbb{R} \to \mathbb{R}^{2p \times V}$ two derivable multi-variate mappings.

In the skip-gram, the context C_k of a given token t_k is given by the following formula

$$C_k = argmax(g(W[k, :]), axis = 1)$$

In the BOW, the central word t_k of given context is given by the following formula

$$t_k = argmax(f([W[t - p : t - 1, :], W[t + 1 : t + p, :]]), axis = 1)$$

Glove(Pennington, Socher, and Manning, 2014) has been brought in toward improving the performance of the word-embedding on similarity-based tasks. In this work, it's claimed that the statistical occurrence of words in a text bears information about their meaning. Hypothesizing that the ratio between the co-occurrence probabilities between words follows the same propensity as the difference between their corresponding vector representation, they trained a log-bilinear model with weighted-least-squares objective to create the embedding look-up table.

Dynamic Word Embedding

The static word embedding methods struggle with the contextualization of the vector representation provided. The meaning of a given word depends on the context in which it is used. For example, in these two sentences: *"I opened a bank account"* and *"I am standing on the river bank"*, the signification of the word *"bank"* is unique in each case, though the static embedding will always give the same vector representation and will consequently fail to capture the contextual dependency within the sentence embedded. On the way to address that issue, several methods have been introduced to strengthen the understanding of the inter-linkages between the tokens of the sequence treated. These approaches share a common part, that is, the static embeddings. On top of that, several attention-based means have been initiated to construct the vector representation of given token based on the representation of the tokens in its neighborhood. The most prominent approaches are Elmo and the Transformer-based encoders (Peters et al., 2018; Ashish et al., 2017). In the former, the word vectors are the internal states of a deep bi-directional Long-Short Term Memory (LSTM) whereas in the latter, the vector representations are figured out by a self-attention-based network that can also be observed as a consensus-based representation where each token attends the computation of vector representation of the others. In this work we exploit the conveniences of the contextual mapping provide by the Transformer-based model to perform a logical mapping between two spaces, namely, the reactant and the enzyme space in bio-catalyzed reactions..

V.3 Self-Attention and Contextual Mapping

In this section, we present the core concept on which our work relies. It is worth precising that we will not fully develop how the vanilla Transformer works in this section as it is established in the original paper (Ashish et al., 2017). In contrast, we will only present the self-attention mechanism under the same conditions as in the original introduction. More precisely, the notions skipped here are the following: the static embedding, the positional encoding, and the multi-head attention.

a Transformer is a sequence transducer function, that is, given an embedded sequence $S \in \mathbb{R}^{n \times d}$, a passing of S through a Transformer yields another sequence S' with the same shape as S. a Transformer mainly consists of two parts: an encoder and a decoder. Each part consists of a stack of components so-called Transformer-blocks and each Transformer-block is made up of two sub-blocks, that is, the self-attention block and a feed-forward neural network on top of the self-attention. The encoder and the decoder of the Transformer are respectively used for the NLU and the natural language generation (NLG). In regard to the goal of our work (which entirely relies on NLU), we will only present the Transformer encoder architecture. Besides the purposeful difference between the Transformer encoder and

encoder, a glimpse on their architectures points out that, the main difference between them is the mask applied on the self-attention mechanism in the decoder for the auto-regressive text generation, that is, the mask ensures the rightward flow of the information in the generation process (the newly generated tokens do not attend the representation of the previously generated ones). The three main components of the self-attention module are the following: the queries, keys, and values.

In the upcoming writings, we will used the following notation:

- $d_{model}, d_k, d_v, d_{ff} \in \mathbb{N}, b_1 \in \mathbb{R}^{d_f f}, b_2 \in \mathbb{R}^{d_v}$
- $W_K, W_Q \in \mathbb{R}^{d_{model} \times d_k}, W_V \in \mathbb{R}^{d_{model} \times d_v}, W_1 \in \mathbb{R}^{d_v \times d_{ff}}, W_2 \in \mathbb{R}^{d_{ff} \times d_v}$ five linear projectors.
- $Att : \mathbb{R}^{\cdot \times d_{model}} \rightarrow \mathbb{R}^{\cdot \times d_{model}}$ the self-attention mapping
- $FFN : \mathbb{R}^{\cdot \times d_{model}} \rightarrow \mathbb{R}^{\cdot \times d_{model}}, FFN(x) = max(0, xW_1 + b1)W_2 + b_2$ the feed-forward neural network
- $LayerNorm : \mathbb{R}^{\cdot \times d_{model}} \rightarrow \mathbb{R}^{\cdot \times d_{model}}$ the layer normalization

Having a sequence $X \in \mathbb{R}^{n \times d_{model}}$, the first transformation within the Transformer-block is described as follows:

$$Q = XW_Q, K = XW_K, V = XW_V$$

$$X' = Softmax(\frac{QK^T}{\sqrt{d_k}})V, X' = X + LayerNorm(X').$$

The second transformation step concerns the passing through the feed-forward network.

$$X' = X' + FFN(X')$$

It is important to highlight that this transformation does not involve the token mixing like the previous one, it only modifies the second dimension of the sequence.

VI Methodology

The extraction of active sites in proteins consists in determining the regions on these proteins where a given substrate can be bound. We address the site active site recognition by reshaping the task into an NLP-based frame in which the site identification consists in the analysis of the attention matrix given a bio-catalyzed reaction. We hypothesized that, train a Transformer-based model with the unsupervised learning tasks MLM and n-gram MLM will inherently capture the inter-activities between the reactants and enzymes. After training the BERT model on reactions with the MLM and n-gram MLM, we performed the mapping between the reactants' atoms and the enzyme's tokens then, compared our active sites with the experimentally determined active sites from PLIP (Adasme et al., 2021) and finally, performed the molecular docking and compared the binding free energy of the complexes obtained with the predicted and the experimental active sites.

VI.1 Formal Definition of the Problem

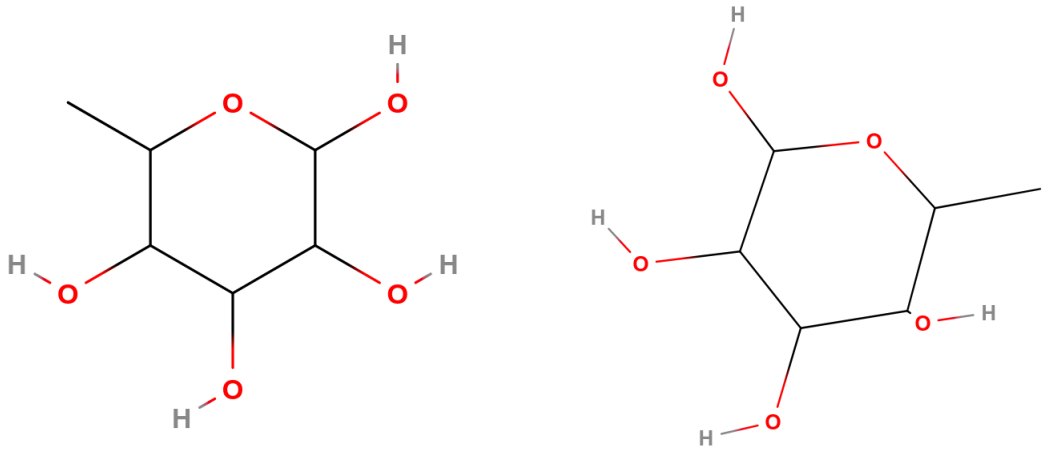
A formal reshaping of the assigned task is essential to understand how we implemented our end-to-end pipeline of active site discovery. The dataset used to train the model is

made up of reaction rows. Each reaction row consists of the concatenation of the SMILES strings of reactants, the amino acid sequence of the enzyme, and the SMILES strings of products. The reactant representation is separated from the amino acid sequence with a pipe "|" symbol and the amino acid sequence is separated from the product representation with the redirection symbol "»". One example of a reaction row is showed in the figure 5

```
C[C@@H]1O[C@@H](O)[C@@H](O)[C@H](O)[C@@H]1O |
METPQTGYQVQSYKIPVKRYCQTLDLRDSPELIAEYRKRHSETEAWPEILAGIREVGILE
MEIYILGTRLFMIVETPVDFDWDTAMARLNTLPRQQEWEEYMAIFQQAAPGMSSAEKWKP
MERMFHLYNT >> C[C@@H]1O[C@H](O)[C@@H](O)[C@H](O)[C@@H]1O
```

FIGURE 5: **Example of Bio-Catalyzed Reaction**

- **C[C@@H]1O[C@@H](O)[C@@H](O)[C@H](O)[C@@H]1O** refers to the reactant (figure 6a)
- **METPQTGYQVQSYKIPVKRYCQTLDLRDSPELIAEYRKRHSETEAWPEILA GIREVGILEMEIYILGTRLFMIVETPVDFDWDTA MARLNTLPRQQEWEEYMAIFQQAAPGMSSAEKWKPMERMFHLYNT**
- **C[C@@H]1O[C@H](O)[C@@H](O)[C@H](O)[C@@H]1O** refers to the product (figure 6b)



(A) **Example of Reactant**

(B) **Example of Product**

FIGURE 6: **Reactant and Product**

After the tokenization of the foregoing reaction, we obtained a list of tokens (presented in the figure 7) referring to the set of the understandable units that will be processed by the model.

['C_', '[C@@H]_', '1_', 'O_', '[C@@H]_', '(_', 'O_', ')_', '[C@@H]_', '(_', 'O_', ')_', '[C@H]_', '(_', 'O_', ')_', '[C@@H]_', '1_', 'O_', '|', 'ME', 'TP', 'QT', 'GYQ', 'VQ', 'SY', 'KIP', 'VK', 'RY', 'CQ', 'TLDL', 'RD', 'SP', 'EL', 'IAE', 'YR', 'KRH', 'SET', 'EAW', 'PEI', 'LAGI', 'RE', 'VGIL', 'E', 'ME', 'IY', 'IL', 'GT', 'RLF', 'MI', 'VE', 'TP', 'VD', 'FD', 'WD', 'TA', 'MA', 'RLN', 'TLP', 'RQ', 'QEW', 'EEY', 'MA', 'IF', 'QQ', 'AAP', 'GMS', 'SA', 'EK', 'W', 'KP', 'ME', 'RMF', 'HLY', 'NT', '>>', 'C_', '[C@@H]_', '1_', 'O_', '[C@H]_', '(_', 'O_', ')_', '[C@@H]_', '(_', 'O_', ')_', '[C@H]_', '(_', 'O_', ')_', '[C@@H]_', '1_', 'O_']

FIGURE 7: **Tokenized Reaction**

Mask Language Modeling

The masked language modeling training in this work is intended to build a model strong enough to get one's hand on the underlying grammar that sustains the underground chemical processes in bio-catalysis. The consensus-based representation of tokens in self-attention-based models, complemented with the MLM training is theoretically and sufficiently strong to construct a suitable contextual mapping to represent the encoded sequence (Yun et al., 2020). In the following writings, we present first, a formal definition of the MLM and secondly, the settings used in our experiments.

MLM Definition.
Our definition of MLM directly incorporates BERT as the Transformer-based model used. Let us consider $S = [s_1, ..., s_n]$, a sequence of n tokens; $BERT : \mathbb{N}^{\cdot} \rightarrow \mathbb{R}^{\cdot \times d_{model}}$, a transducer function representing the model BERT; $[MASK]$, a token that belongs to the language vocabulary and used to represent the masked tokens; $f : V^{\cdot} \rightarrow \mathbb{N}^{\cdot}$, a bijective mapping from the vocabulary to the set of natural numbers that binds each to token to its corresponding index in the set of vocabulary index, $g : \mathbb{R}^{\cdot \times d_{model}} \rightarrow \mathbb{R}^{\cdot \times |V|}$, a linear mapping designed to map the latent presentation space of sequence to the vocabulary space in order to facilitate the prediction of the masked tokens. The important steps of the MLM traning are the following:

- Choose the masking probability p (0.15 in our case). The probability to mask a given token under the binomial distribution.
- Mask the entry sequence with the forenamed probability and keep the indices of the the masked tokens. As the result, we get a sequence S' that contains $\lceil p|S| \rceil$ of masked tokens and M the indices of the tokens $[MASK]$.
- Transform the masked sequence S' to its analogous sequence of the index, named X in our settings.
- A forward pass of the sequence X through the Transformer model, in that case BERT, that gives a hidden representation $X' \in \mathbb{R}^{n \times d_{model}}$ of X
- A forward pass of X' through the linear projector g, yielding a sequence $X' \in \mathbb{R}^{n \times |V|}$. The second dimension the tensor X' could be viewed a the logits that will be fed to a smooth function to draw out the probability distribution of the masked tokens over the vocabulary.
- The final step contains the application of smoother on the aforesaid logits, the computation of the loss value through an objective function and finally the optimization of that loss value. The commonly used smoother function is $softmax$ that gives a probability distribution of the predicted tokens over the vocabulary space. The following

equations summarize the final step of MLM training.

$$X' = softmax(X', axis = 1)$$

$$loss = -\sum_{i \in M} log\ X'[i, X_i]$$

The value $loss$ is optimized and the parameters of the model $BERT$ are updated appropriately.

A bird's eye view on the proteins in our data reveals that, based on the sequence information, the active site on proteins are more locally than sparsely distributed over the sequence. The skewed distribution of the active regions led us to the local feature analysis on the protein-side while keeping the broad feature investigation for the reactants and products. This reason motivated our choice of mixing the MLM and n-gram MLM training. Concretely, we sparsely masked the tokens of the reactants and products and densely masked the tokens of the proteins, strictly speaking:

- we randomly selected p of reactants and products' tokens then masked them
- we chose $n(n = 3)$ the size of the masking, randomly the chose the centers of the ball in which the tokens will be mask. Roughly speaking, we chose $\left\lceil \frac{p|S_p|}{n} \right\rceil$ centroids and S_p refers to the sequence of amino acid residues.

The figure 8 presents an example of the masked reactions under the aforementioned conditions.

['C_', '[MASK]', '1_', 'O_', **'[MASK]'**, '(_', 'O_', ')_', '[C@@H]_', '(_', 'O_', ')_', '[C@H]_', '(_', 'O_', ')_', '[C@@H]_', '1_', **'[MASK]'**, 'I', 'ME', 'TP', 'QT', 'GYQ', 'VQ', 'SY', 'KIP', 'VK', 'RY', 'CQ', 'TLDL', 'RD', 'SP', 'EL', 'IAE', 'YR', 'KRH', 'SET', 'EAW', 'PEI', 'LAGI', **'[MASK]', '[MASK]', '[MASK]'**, 'ME', **'[MASK]', '[MASK]', '[MASK]'**, 'RLF', 'MI', 'VE', 'TP', 'VD', 'FD', 'WD', 'TA', 'MA', 'RLN', 'TLP', 'RQ', 'QEW', 'EEY', 'MA', 'IF', 'QQ', **'[MASK]', '[MASK]', '[MASK]', '[MASK]', '[MASK]'**, 'KP', 'ME', 'RMF', 'HLY', 'NT', '>>', 'C_', '[C@@H]_', '1_', 'O_', '[C@H]_', '(_', 'O_', ')_', '[C@@H]_', '(_', 'O_', ')_', '[C@H]_', '(_', **'[MASK]'**, ')_', '[C@@H]_', '1_', **'[MASK]'**]

FIGURE 8: **Masked Reaction**

Active Site Definition

Active sites on enzymes are the regions where substrates can be bound and undertake a chemical reaction. The principal approaches used to determine the 3D structures of protein are the X-ray protein crystallography and the nuclear magnetic resonance (NMR). Having the three-dimensional coordinates of each atom of a given protein, it becomes obvious to observe the clustering of active regions using the Euclidean distance in that 3D space as the clustering metric. Two pillars that foster the sequence-side analysis of protein are the following: transform a protein sequence in its 3D shape remains a great problem in the protein engineering in the sense that a given sequence could have multiple representation depending on the functional properties expected, and there is no a direct binding between the residues in sequence information and the residues in the 3D structure. By dint of the non-intrinsic binding between sequence and three-dimensional structure residues, the local leaning of

active sites in 3D space is not well observed with the sequence information. Owing to the non-locality distribution on sequence, we defined the active region of a given protein as the sequence of the active segments of the that protein. Formally, consider $S = [r_1, ..., r_n]$ a sequence of n amino acid residues, the active region A_S of S will be defined as follows:

$$A_S = \{(a_i, b_i)\}_{i=0}^{m}$$

(a_i, b_i) and m refer respectively to the boundary indices of the active segment i and the number of active segments on the protein.

Active Site Extraction Algorithm

The extraction algorithm entirely relies on the analysis of the attention matrix after embedding a reaction through the Transformer model. It is noteworthy to step back in the self-attention equations and directly point the component subject to our analysis during the site extraction. In the next writings we will keep the same terminology as in the section V.3 and assume that the model BERT is already trained under the conditions specified in the section VI.1. Consider S, $(|S| = n = r + m + p)$ as a reaction r, m and p refer respectively to the length of the reactant, the enzyme, and the protein. As it is stated in (Ashish et al., 2017), each Transformer-block performs the self-attention computation and therefore produces an attention matrix Att determined by:

$$Att = Softmax(\frac{QK^T}{\sqrt{d_k}}) \in \mathbb{R}^{n \times n}$$

Since we were interested in the mapping between the reactant and the enzyme space to be able to draw the active regions of the enzymes, we extracted the sub-matrices which hold information regarding the binding between the reactant and the enzyme. More concretely, we took out a matrix $P \in \mathbb{R}^{r \times m}$ obtained by adding up the following two sub-matrices: $Att[1:r, 1:m]$ and $Att[r+1:r+1+m, 1:r]^T$, that is,

$$P = Att[1:r, 1:m] + Att[r+1:r+1+m, 1:r]^T$$

The figure 9 presents an instance of the attention matrix extracted from the initial attention-matrix for a given reaction. The algorithm 1 used the draw out the active sites is a consensus-based algorithm in which each reactant's atom has k votes to choose the best-bound enzyme's tokens. The chosen enzyme's tokens are gathered and entitled to the active region of the enzyme.

Algorithm 1 Active Site Extraction

```
1: procedure EXTRACTION(P ∈ ℝ^{r×m}, k)
2:     active_site = set()
3:     for i in 1..r do
4:         line = P[i]
5:         for j in argmax(line, k) do
6:             active_site.add(j)
7:     return active_site
```

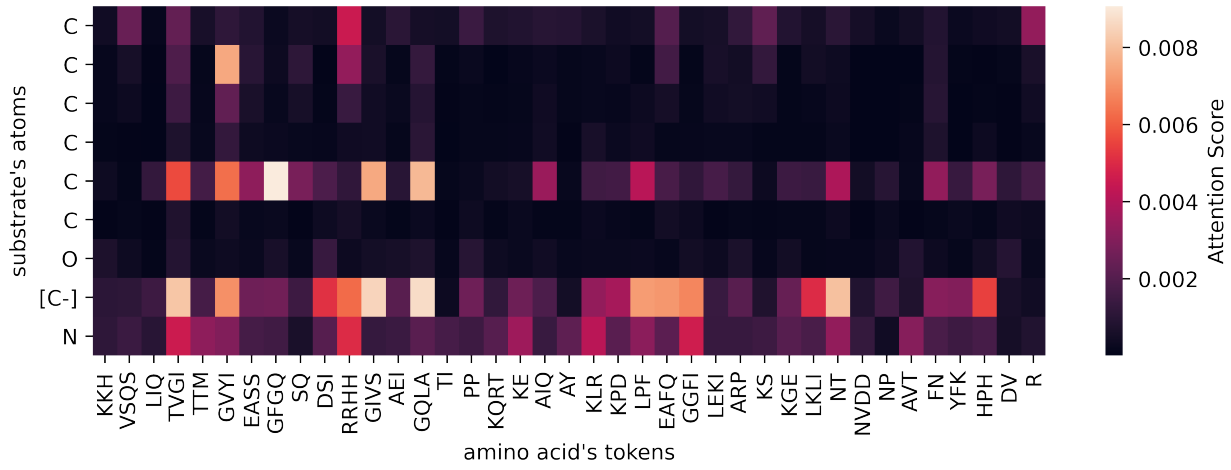


FIGURE 9: **Reactant-Enzyme attention**

VI.2 Data Preparation

A great part of the data was already collected by the host institution at the beginning of the internship. Although the data preparation is not the main point of the course, we will succinctly described the acquisition process in the following words:

Training Data Preparation

The training data consists of a mixture bio-catalyzed and organic reactions where an enzymatic reaction comprised the reactants, an enzyme, and the products, and an reaction is made up of the concatenation of the SMILES strings of reactants and products separated by the redirection symbol "»". The organic reactions were collected from USPTO (BHAVEN, 2011) whilst the enzymatic ones were collected from different sources with much more complex approach.

The enzyme commission (EC) number is numerical nomenclature of enzymes based on the type of chemical reactions they accelerate. The preliminary processes of the dataset collection, consisting in the gathering of the enzymatic reaction complemented with the EC number, was conducted as the same as initiated in (Probst et al., 2021). The enzymatic reactions with the EC numbers were collected from different sources including: Rhea (Alcântara et al., 2012), BRENDA (Schomburg, Jäde, and Schomburg, 2002), PathBank (Wishart et al., 2020), and MetaNex (Ganter et al., 2013). Once the catalyzed reactions with the EC numbers were downloaded, the EC number for each reaction was substituted by all the enzyme proteins within that EC number family. The amino acid sequence of proteins were collected from UnitProt (Consortium, 2016). The figure 10 shows the EC number distribution in dataset used to lead our experiments.

Test Data Preparation

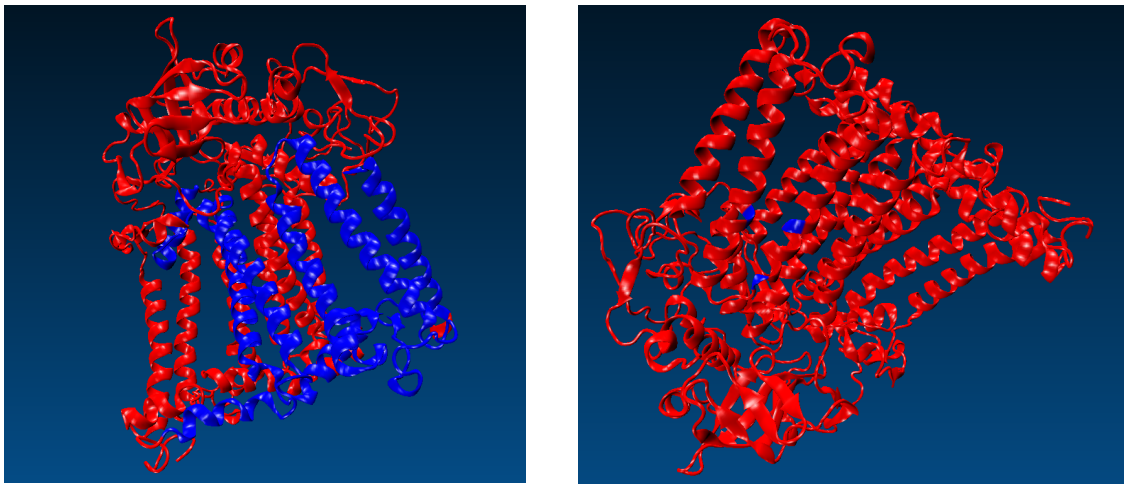
As we independently evaluated the aptness of Transformers models to unsupervisedly recognize active sites on proteins, we need a test bed from that learning experience. Two approaches were used to construct the test set. The first concerns the interaction annotation

FIGURE 10: **Enzyme Commission number distribution**
The distribution of samples at EC-levels 1 (corresponding to enzyme classes) and 2, as well as EC-levels 2 and 3 of oxidoreductases (class 1), transferases (class 2), hydrolases (class 3), lysases (class 4), isomerases (class 5), ligases (class 6), and translocases (class 7). Source (Probst et al., 2021)

of the co-crystallized ligand-protein complexes using the Protein-Ligand Interaction Profiler (PLIP) (Krivák and Hoksza, 2015), the second is about the annotation of amino acid sequences using the sequence alignment algorithm over different protein families to check the preserved active domains; the external tool used to perform that sequence alignment was The Protein Families Database (Pfam) (Mistry et al., 2020).

The annotation of sequences with the protein-ligand profiler was already carried out by the welcome institution. The annotation with Pfam was realized by designing a sub-system

interacting with the xml API of Pfam [4]. By this end, we annotated about 60K protein sequences. A noteworthy difference between the two annotation processes is the dependency on the substrate, that is, PLIP's annotation uses the protein-ligand complex for the annotation and consequently needs the substrate to detect the binding regions on proteins whereas the Pfam's annotation is completely substrate-free, it only leans on the overlapping between a given sequence and the preserved domains of different protein families. Although the two approaches undertaken diverge on the substrate dependency perspective, the annotated active regions follow the same rules as it is mentioned in the section VI.1. Figures 11a and 11b show a protein annotation from Pfam and PLIP respectively, the blue part in both represents the active regions and the red one represents the remainder. Knowing that PLIP actually uses the three-dimensional representation to tag the ligand-protein intern-linkages, it will be further consider as the ground-truth for the assessment and Pfam's annotations will be consider as the baseline though.



(A) **Pfam's Annoation** (B) **PLIP's Annotation**

FIGURE 11: **Protein Annotation**

VI.3 Evaluation

We developed a two-folds evaluation to analyze the unsupervised binding site predictions of our model. The first is an alignment-based evaluation and the second an assessment based on the analysis on the free energy of binding of ligand-protein complexes.

Alignment-Based Evaluation

Because of the use of sequence information to train the model BERT to understand language of catalyzed reaction, we initiated a sequence-based evaluation to test the ability of the model to capture the interactions with substrates at the sequence unit level (residue level). In this context, the main statistical indicators used were the following: the overlapping score, the false positive rate, the token recovery.

[4] https://pfam.xfam.org/help#tabview=tab11

- **Overlapping Score**. It measures the matching at the residue level between the experimentally found active segments and the predicted ones. Considering the formal definition of the binding segment defined in VI.1, calculating the overlapping score is turned to the measurement of the overlapping between a list of intervals. Strictly speaking, let consider S, $|S| = n$ a sequence of residues, the overlapping score between the predicted active region $A = \{(a_{1i}, b_{1i})\}_i^n$ and the ground-truth $B = \{(a_{2i}, b_{2i})\}_i^m$ is defined as follows:

$$overlap\ score = \frac{\sum_{i}^{n}\sum_{j}^{m} max(0, min(b_{1i}, b_{2j}) - max(a_{1i}, a_{2j}))}{\sum_{i}^{m}(b_{2i} - a_{2i})}$$

- **False Positive Rate**. It quantifies the failures of the model. In our configurations, a false positive segment refers to a predicted segment that doesn't overlap any experimentally determined active segment. The next formula defines the false positive rate:

$$False\ positive\ rate = \frac{\sum_{i}^{n}(b_{1i} - a_{1i})\mathbb{1}_{\bigwedge_{j=1}^{m}[a_{1i},b_{1i}]\cap[a_{2j},b_{2j}]=\emptyset}}{\sum_{i}^{n}(b_{1i} - a_{1i})}$$

- **Token Recovery**. The last statistical indicator we used, reflects the fraction of the enzyme's token that was utilized in the prediction. This indicator is significant because it regulates the token expansion as the number of votes for each reactant's atom increases. The following is the definition of token recovery:

$$token\ recovery = \frac{|predicted\ tokens|}{|enzyme's\ tokens|}$$

Free Energy-Based Evaluation

The major limitation of the sequence-based evaluation is that it doesn't take into consideration the three-dimensional location of the annotated residues. A non-overlapping on the sequence-side could still be a valid active region based on the 3D distance to the actual binding sites. Once the ligand has found a conformation to be bound to a given protein, a fair means to evaluate the goodness of the pose (ligand-protein complex) is to measure its binding affinity. The common method used to assess the binding affinity is the evaluation of the free energy of the pose. In the way of broadening our understanding of the predictions, we performed the molecular docking with Autodock Vina (Eberhardt et al., 2021; Trott and Olson, 2010) and computed the free energy of the poses using the two configurations: (1) the centre of the binding box was computed by averaging the coordinates of the atoms in the predicted active sites. (2) the centre of the binding box was computed by averaging the coordinates of the atoms in the experimental active sites.

VII Experiments and Results

The data preparation procedure is reminiscent of what was done in (Probst et al., 2021). On top of the ECREACT dataset, we replaced the Enzyme Commission numbers with all the amino acid sequences of proteins within their classes drew from the UnitProt database. The EC numbers were balanced within the dataset to assure that each type of reaction, according to the EC classification, is significantly represented. The emerging dataset was split into the following parts for the training:

- **Training Set** : ∼7M of reactions.
- **Validation Set** : ∼4K of reactions.
- **Test Set**: ∼4K of reactant-enzyme pair in co-crystallized form.

We trained the model BERT with the MLM and n-gram MLM using the following configuration:

- Batch size : 4096
- Optimization Algorithm : LAMB, with 20928 optimization steps.
- Learning rate scheduler: ReduceOnPlateau with 0.7 as the reduction factor.

TABLE 1: Overlapping score, false positive rate, and token recovery

	Overlap Score	False Positive Rate	Token Recovery
Random Model	4.98%	84.20%	11%
Pfam	24.01%	78.01%	13.09%
BERT-base	28.98%	75.56%	9.35%
RXNAAMapper (ours)	**31.51%**	**66.63%**	11.1%

The sequence-based evaluation was carried out in two steps. The first concerned the evaluation of the ability of the model to recover the active regions whilst considering the ground truth as the sites predicted by Pfam (using sequence alignment over different protein families to check the preservation of the domain). The second referred to the evaluation of the faculty of the model to recognize the sequence segment of the experimental binding sites determined with PLIP. In the first evaluation configuration, we were able to recover 67.7% of the active regions. table 1 summarizes the result obtained in the second configuration, the description of each approach undertaken is given as follows:

- RXNAAMapper corresponds to our model built on the contextual mapping of the reactant-enzyme pair provided by BERT.
- BERT-base refers to the basic pretrained version of BERT of linguistic corpora
- Pfam symbolizes the active site provided by the sequence alignment algorithm
- Random Model, to be sure that we capture insightful information on reactions in terms of the mapping between the reactant and enzyme space, we built up a random model to design such as baseline to outperform. Following that way, we uniformly randomly sampled tokens using the same token recovery as the RXNMapper then, entitled these tokens to the active regions predicted by the random model.

The evaluation based only on the sequence information does not provide insights regarding three-dimensional location. To reinforce our understanding of the active regions predicted. We performed a protein-ligand docking with the predicted active regions and the ground truth using the docking tool Autodock Vina (Eberhardt et al., 2021; Trott and Olson, 2010) then compared the binding free energy of the two configurations. Given the three-dimensional structure of a ligand and a protein, the 3D coordinates of the centre of the binding box, and the size of the binding box, the protein-ligand docking consists in finding the best conformation of the ligand on the protein within the binding box. The best conformation there refers to the configuration of the ligand which has the lowest free energy. As we needed the centres of the binding box in the two configurations namely: the predicted active regions and the experimentally found active sites, we averaged the coordinates of all the atoms within each region and we set 50 Angstrom as the size of the docking box. At the end of the molecular docking, we compared the binding free energy of the two settings, figure 12 reports visual information regarding the free energy of the predicted and ground

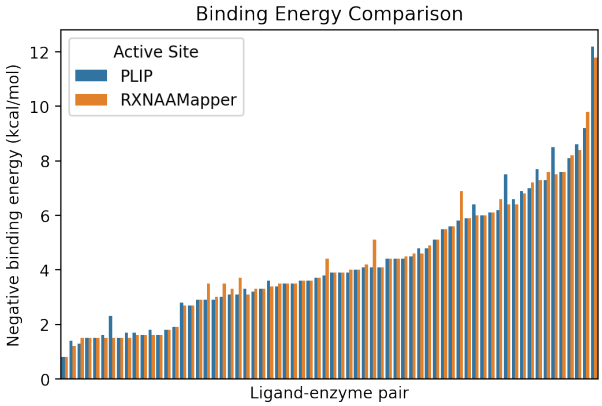


FIGURE 12: **Docking Energy**

truth active sites. Figure 13 presents a study case where we compare the binding site annotation using the reaction mapper we built (RXNAAMapper referring to our model) against the sequence alignment. As is has been pointed out in this instance, our RXNAAMapper is more accurate than the sequence overlap algorithm (Pfam). The predictions are more densely distribution rather than sparsely like with Pfam.

VIII Discussion

The first step analysis of the results pointed in table 1 shows that the model BERT independently trained with self-supervised learning tasks can apprehend the mapping between the reactant and the enzyme which is important in finding the active regions of that enzyme protein. We were able to unprecedentedly recover 31.51% of experimental binding sites. Knowing that these ground truth binding sites were determined with the 3D structure of the ligand-protein complexes using PLIP, the results obtained are a significant signal that besides the sequence information grasped by our model, it is also able to implicitly learn the layout of the active regions in the three-dimensional structure. A willingness to fully understand the prediction led us to the authentication with molecular docking. The result of

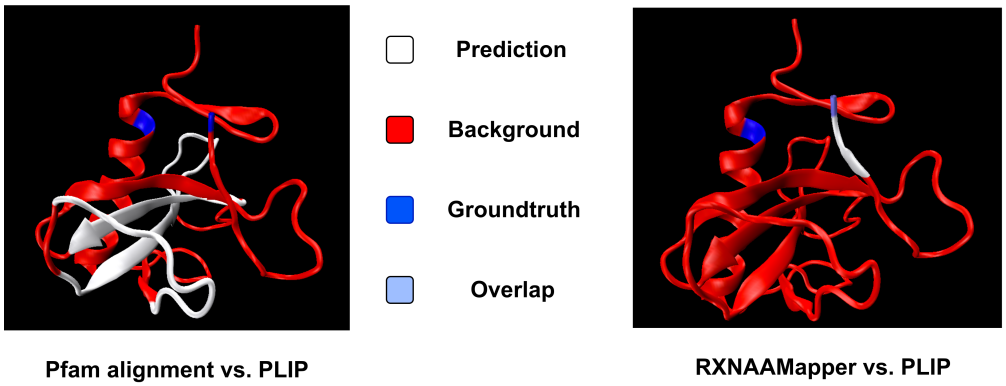


FIGURE 13: **Comparing active site predictions in 3D**
Comparison of the prediction from Pfam alignments (left) and RXNAAMapper (right) using PLIP as a groundtruth.

the ligand-protein docking confirms our hypothesis stating that the mappings between reactants and enzymes are unconditionally learned with MLM-based training of self-attention-based models. Following the reverse grade scale of the energy in figure 12, the greater the energy, the better found the active site. Indeed, our RXNAAMapper performs on par with PLIP (used to draw out the ground truth active regions) and in some situations outperforms it.

IX Personal Assessment

This section provides a personal evaluation of our internship, we will mainly point out the major difficulties encountered the means undertaken to overpass them. Overall, we state that the internship was a lovely experience and was conducted under the thorough supervision of my mentor Dr Matteo Manica and my closest manager Dr Teodoro Laino. They supported me both in the work environment and my personal life. The foremost expectation I had before starting the course was to lead it face-to-face, due to the pandemic, COVID-19, which severely hit the world and negatively impact the daily life at business, we were obliged to undertake the training remotely with some days onsite for the sake of the employees' health. This, therefore, represents the main difficulty I encountered during our internship. Working with like-minded people, the main skills I developed during my internship are the following:

- **Large-Scale Data Management**. The datasets used to train our models were extremely large (∼200Gb). Hence, the training was not lead as usual with small-scale data and we, therefore, developed methods and attitudes to optimally manage the data size taking into account the infrastructure limitation (GPU memory, RAM, Disk, Network bandwidth)

- **Literature in Drug Discovery**. Chemistry is not the main focus of my studies but by leveraging the knowledge acquired in past experiences, namely my previous internship, on drug discovery, I deepen my understanding of how catalyzed reactions work and how we can build computer-aided systems to withstand bio-catalyst processes.

For instance, the computer-aided approach is based on language modelling for active site detection.

- **Presentation Skills**. The internship was sprinkled with events wherein I had to briefly summarize the project I was working on to someone not in the same team. The major events are the following: poster session, group presentation, and scrum meeting.

During our course, we worked with the scrum methodology wherein the milestones were set progressively to achieve the main goal, consisting in designing an attention value mapper for the enzymatic active site recognition. The workflow of our work is summarized in the following Gantt diagram. 14

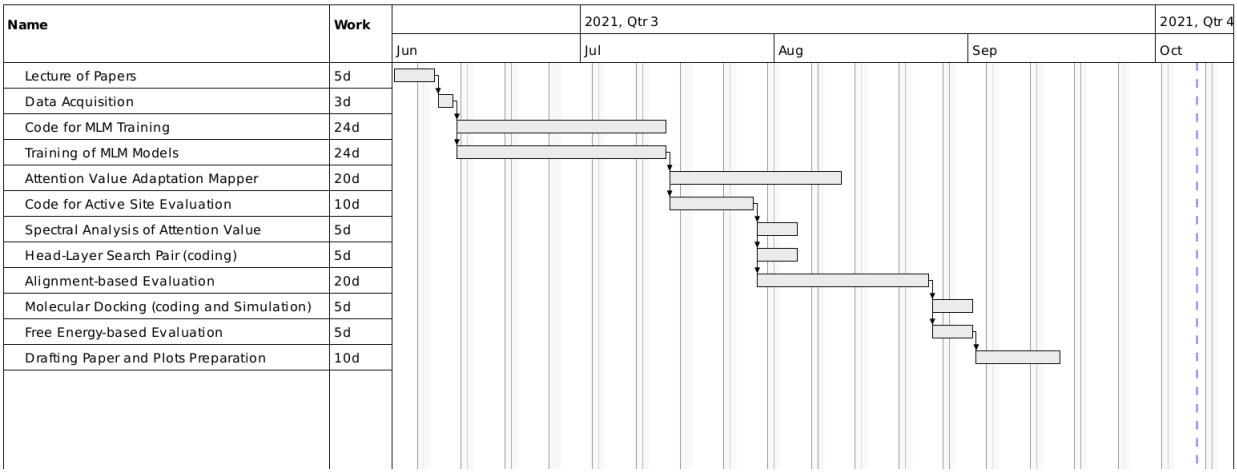


FIGURE 14: **Gantt Diagram**

X Conclusion

Use only sequence information of reaction to finding out the active sites of proteins, enzymes involved in the reactions represents a significant milestone towards breaking down the 3D structure dependency of the methods in accelerated discovery to date. In this work we presented an active site recognition approach that entirely relies on the self-attention mechanism of Transformer models, trained with the self-supervised learning tasks MLM and n-gram MLM on the mixture of bio-catalyzed and organic reactions, which was able to recover one-third of the experimental sites. A double-checking of the predictions showed us that the predictions of sites obtained binding energy commensurable with the experimental sites one. This signal effectively confirms that the method built can recognize the active regions of the amino acid sequence of proteins.

Regarding the expectations of the internship which are in a nutshell, design a reactant amino acid mapper and contribute to a research paper on the topic, we can state that, according to the results obtained, the expectations are reached and a contribution has been made on a research paper under reviews at the NeurIPS workshop AI for Science. Albeit this noteworthy achievement has been reached, tremendous eras remain uncovered in the

accelerated discovery. One road map that could be undertaken in line with the work we have done, is the study of the syntax morphology of protein for a better functional understanding. Indeed, in our work, we used a fixed tokenization process (which is free from the gradient optimization) to split the protein sequence into tokens, thus we implicitly lose information on the morphology of protein.

Bibliography

Adasme, Melissa F et al. (May 2021). "PLIP 2021: expanding the scope of the protein–ligand interaction profiler to DNA and RNA". In: *Nucleic Acids Research* 49.W1, W530–W534. ISSN: 0305-1048. DOI: 10.1093/nar/gkab294. eprint: https://academic.oup.com/nar/article-pdf/49/W1/W530/38841758/gkab294.pdf. URL: https://doi.org/10.1093/nar/gkab294.

Aggarwal, Rishal et al. (2021). "DeepPocket: Ligand Binding Site Detection and Segmentation using 3D Convolutional Neural Networks". In: *Journal of Chemical Information and Modeling*. PMID: 34374539. DOI: 10.1021/acs.jcim.1c00799. eprint: https://doi.org/10.1021/acs.jcim.1c00799. URL: https://doi.org/10.1021/acs.jcim.1c00799.

Al-Rfou, Rami et al. (2018). *Character-Level Language Modeling with Deeper Self-Attention*. arXiv: 1808.04444 [cs.CL].

Alcântara, Rafael et al. (Jan. 2012). "Rhea–a manually curated resource of biochemical reactions." In: *Nucleic Acids Research* 40.Database issue, pp. D754–60. DOI: 10.1093/nar/gkr1126. URL: https://hal.inria.fr/hal-00746861.

Ashish, Vaswani et al. (2017). "Attention Is All You Need". In: *CoRR* abs/1706.03762. arXiv: 1706.03762. URL: http://arxiv.org/abs/1706.03762.

Berman, H., K. Henrick, and Haruki Nakamura (2003). "Announcing the worldwide Protein Data Bank". In: *Nature Structural Biology* 10, pp. 980–980.

BHAVEN (2011). *USPTO Patent and Citation Data*. Version V4. DOI: 10.7910/DVN/SJPHLG. URL: https://doi.org/10.7910/DVN/SJPHLG.

Brandes, Nadav et al. (2021). "ProteinBERT: A universal deep-learning model of protein sequence and function". In: *bioRxiv*. DOI: 10.1101/2021.05.24.445464. eprint: https://www.biorxiv.org/content/early/2021/05/25/2021.05.24.445464.full.pdf. URL: https://www.biorxiv.org/content/early/2021/05/25/2021.05.24.445464.

Consortium, The UniProt (Nov. 2016). "UniProt: the universal protein knowledgebase". In: *Nucleic Acids Research* 45.D1, pp. D158–D169. ISSN: 0305-1048. DOI: 10.1093/nar/gkw1099. eprint: https://academic.oup.com/nar/article-pdf/45/D1/D158/23819877/gkw1099.pdf. URL: https://doi.org/10.1093/nar/gkw1099.

— (Nov. 2020). "UniProt: the universal protein knowledgebase in 2021". In: *Nucleic Acids Research* 49.D1, pp. D480–D489. ISSN: 0305-1048. DOI: 10.1093/nar/gkaa1100. eprint: https://academic.oup.com/nar/article-pdf/49/D1/D480/35364103/gkaa1100.pdf. URL: https://doi.org/10.1093/nar/gkaa1100.

Devlin, Jacob et al. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. arXiv: 1810.04805 [cs.CL].

Eberhardt, Jerome et al. (2021). "AutoDock Vina 1.2.0: New Docking Methods, Expanded Force Field, and Python Bindings". In: *Journal of Chemical Information and Modeling* 61.8. PMID: 34278794, pp. 3891–3898. DOI: 10.1021/acs.jcim.1c00203. eprint: https://doi.org/10.1021/acs.jcim.1c00203. URL: https://doi.org/10.1021/acs.jcim.1c00203.

Elnaggar, Ahmed et al. (2020). "ProtTrans: Towards Cracking the Language of Life's Code Through Self-Supervised Deep Learning and High Performance Computing". In: *bioRxiv*. DOI: 10.1101/2020.07.12.199554. eprint: https://www.biorxiv.org/content/early/2020/07/12/2020.07.12.199554.full.pdf. URL: https://www.biorxiv.org/content/early/2020/07/12/2020.07.12.199554.

Filipavicius, Modestas et al. (2020). *Pre-training Protein Language Models with Label-Agnostic Binding Pairs Enhances Performance in Downstream Tasks*. arXiv: 2012.03084 [q-bio.BM].

Gage, Philip (1994). "A new algorithm for data compression". In: *The C Users Journal archive* 12, pp. 23–38.

Ganter, M. et al. (2013). "MetaNetX.org: a website and repository for accessing, analysing and manipulating metabolic networks". In: *Bioinformatics* 29, pp. 815 –816.

Gligorijević, Vladimir et al. (May 2021). "Structure-based protein function prediction using graph convolutional networks". In: *Nature Communications* 12.D1, pp. D412–D419. ISSN: 2041-1723. DOI: 10.1038/s41467-021-23303-9. eprint: https://www.nature.com/articles/s41467-021-23303-9.pdf. URL: https://doi.org/10.1038/s41467-021-23303-9.

Guo, Jin (1997). "Critical Tokenization and its Properties". In: *Comput. Linguistics* 23, pp. 569–596.

Harrison, Paul et al. (Jan. 2015). "Development of a Machine Learning Method to Predict Membrane Protein-Ligand Binding Residues Using Basic Sequence Information". In: *Advances in Bioinformatics*. ISSN: 1687-8027. DOI: 10.1186/s13321-015-0059-5. URL: https://doi.org/10.1155/2015/843030.

Ingraham, John et al. (2019). "Generative Models for Graph-Based Protein Design". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2019/file/f3a4ff4839c56a5f460c88cce3666a2b-Paper.pdf.

Kozlovskii, Igor and Petr Popov (Oct. 2020). "Spatiotemporal identification of druggable binding sites using deep learning". In: *Communications Biology* 3. ISSN: 2399-3642. DOI: 10.1038/s42003-020-01350-0. URL: https://doi.org/10.1038/s42003-020-01350-0.

Krivák, Radoslav and David Hoksza (Apr. 2015). "Improving protein-ligand binding site prediction accuracy by classification of inner pocket points using local features". In: *Journal of Cheminformatics* 7. ISSN: 1758-2946. DOI: 10.1186/s13321-015-0059-5. eprint: https://jcheminf.biomedcentral.com/track/pdf/10.1186/s13321-015-0059-5.pdf. URL: https://doi.org/10.1186/s13321-015-0059-5.

Liang, Shide et al. (Aug. 2006). "Protein binding site prediction using an empirical scoring function". In: *Nucleic Acids Research* 49.D1, pp. D412–D419. ISSN: 1362-4962. DOI: 10.1093/nar/gkl454. eprint: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1540721/. URL: https://doi.org/10.1093/nar/gkl454.

Lin, Tzyy-Shyang et al. (2019). "BigSMILES: A Structurally-Based Line Notation for Describing Macromolecules". In: *ACS Central Science* 5.9, pp. 1523–1531. DOI: 10.1021/acscentsci.9b00476. eprint: https://doi.org/10.1021/acscentsci.9b00476. URL: https://doi.org/10.1021/acscentsci.9b00476.

Madani, Ali et al. (2020). "ProGen: Language Modeling for Protein Generation". In: *bioRxiv*. DOI: 10.1101/2020.03.07.982272. eprint: https://www.biorxiv.org/content/early/2020/03/13/2020.03.07.982272.full.pdf. URL: https://www.biorxiv.org/content/early/2020/03/13/2020.03.07.982272.

Mistry, Jaina et al. (Oct. 2020). "Pfam: The protein families database in 2021". In: *Nucleic Acids Research* 49.D1, pp. D412–D419. ISSN: 0305-1048. DOI: 10.1093/nar/gkaa913. eprint: https://academic.oup.com/nar/article-pdf/49/D1/D412/35363969/gkaa913.pdf. URL: https://doi.org/10.1093/nar/gkaa913.

Noord, Rik van, Antonio Toral, and Johan Bos (2020). *Character-level Representations Improve DRS-based Semantic Parsing Even in the Age of BERT*. arXiv: 2011.04308 [cs.CL].

Pennington, Jeffrey, Richard Socher, and Christopher Manning (Oct. 2014). "GloVe: Global Vectors for Word Representation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, pp. 1532–1543. DOI: 10.3115/v1/D14-1162. URL: https://aclanthology.org/D14-1162.

Peters, Matthew E. et al. (2018). *Deep contextualized word representations*. arXiv: 1802.05365 [cs.CL].

Probst, Daniel et al. (2021). "Molecular Transformer-aided Biocatalysed Synthesis Planning". In: *ChemRxiv*. DOI: 10.26434/chemrxiv.14639007. URL: https://app.dimensions.ai/details/publication/pub.1138314574andhttps://chemrxiv.org/engage/api-gateway/chemrxiv/assets/orp/resource/item/60c75919842e6599a7db4990/original/molecular-transformer-aided-biocatalysed-synthesis-planning.pdf.

Schomburg, Ida, Antje Jäde, and Dietmar Schomburg (Feb. 2002). "BRENDA, Enzyme data and metabolic information". In: *Nucleic acids research* 30, pp. 47–9. DOI: 10.1093/nar/30.1.47.

Simonovsky, Martin and Joshua Meyers (2020). "DeeplyTough: Learning Structural Comparison of Protein Binding Sites". In: *Journal of Chemical Information and Modeling* 60.4. PMID: 32023053, pp. 2356–2366. DOI: 10.1021/acs.jcim.9b00554. eprint: https://doi.org/10.1021/acs.jcim.9b00554. URL: https://doi.org/10.1021/acs.jcim.9b00554.

Stepniewska-Dziubinska, Marta M., Piotr Zielenkiewicz, and Pawel Siedlecki (Mar. 2020). "Improving detection of protein-ligand binding sites with 3D segmentation". In: *Scientific Reports* 10. ISSN: 2045-2322. DOI: 10.1038/s41598-020-61860-z. eprint: https://jcheminf.biomedcentral.com/track/pdf/10.1186/s13321-015-0059-5.pdf. URL: https://doi.org/10.1038/s41598-020-61860-z.

Trott, Oleg and A. Olson (2010). "AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading". In: *Journal of Computational Chemistry* 31.

Wang, Alex et al. (2019). *GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding*. arXiv: 1804.07461 [cs.CL].

Weininger, David (1988). "SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules". In: *Journal of Chemical Information and Computer Sciences* 28.1, pp. 31–36. DOI: 10.1021/ci00057a005. eprint: https://pubs.acs.org/doi/pdf/10.1021/ci00057a005. URL: https://pubs.acs.org/doi/abs/10.1021/ci00057a005.

Wishart, D. et al. (2020). "PathBank: a comprehensive pathway database for model organisms". In: *Nucleic Acids Research* 48, pp. D470 –D478.

Yun, Chulhee et al. (2020). *Are Transformers universal approximators of sequence-to-sequence functions?* arXiv: 1912.10077 [cs.LG].

Zhenzhong, Lan et al. (2019). "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations". In: *arXiv preprint arXiv:1909.11942*.