
evoBPE: EVOLUTIONARY PROTEIN SEQUENCE TOKENIZATION

Burak Suyunu*

Department of Computer Engineering
Boğaziçi University
İstanbul, Türkiye
burak.suyunu@std.bogazici.edu.tr

Özdeniz Dolu*

Department of Computer Engineering
Boğaziçi University
İstanbul, Türkiye
ozdeniz.dolu@bogazici.edu.tr

Arzucan Özgür

Department of Computer Engineering
Boğaziçi University
İstanbul, Türkiye
arzucan.ozgur@bogazici.edu.tr

ABSTRACT

Recent advancements in computational biology have drawn compelling parallels between protein sequences and linguistic structures, highlighting the need for sophisticated tokenization methods that capture the intricate evolutionary dynamics of protein sequences. Current subword tokenization techniques, primarily developed for natural language processing, often fail to represent protein sequences' complex structural and functional properties adequately. This study introduces evoBPE, a novel tokenization approach that integrates evolutionary mutation patterns into sequence segmentation, addressing critical limitations in existing methods. By leveraging established substitution matrices, evoBPE transcends traditional frequency-based tokenization strategies. The method generates candidate token pairs through biologically informed mutations, evaluating them based on pairwise alignment scores and frequency thresholds. Extensive experiments on human protein sequences show that evoBPE performs better across multiple dimensions. Domain conservation analysis reveals that evoBPE consistently outperforms standard Byte-Pair Encoding, particularly as vocabulary size increases. Furthermore, embedding similarity analysis using ESM-2 suggests that mutation-based token replacements preserve biological sequence properties more effectively than arbitrary substitutions. The research contributes to protein sequence representation by introducing a mutation-aware tokenization method that better captures evolutionary nuances. By bridging computational linguistics and molecular biology, evoBPE opens new possibilities for machine learning applications in protein function prediction, structural modeling, and evolutionary analysis.

Keywords Tokenization · Protein Sequence · BPE · Protein Domain · Evolution

1 Introduction

Recent advancements in computational biology have drawn insightful parallels between protein sequences and linguistic structures, introducing novel methods for protein sequence analysis. Natural language processing (NLP) techniques have emerged as transformative tools for unraveling complex protein sequence patterns, conceptualizing amino acid chains as a form of biological language [Heinzinger et al., 2019, Nambiar et al., 2020, Rives et al., 2021, Elnaggar et al., 2021, Ofer et al., 2021, Brandes et al., 2022, Lin et al., 2023, Elnaggar et al., 2023].

Central to these computational approaches is the critical preprocessing step of tokenization—a method of splitting sequences into meaningful units that fundamentally shapes computational analysis. Besides character-based and k-mer-based tokenization, subword tokenization methods, traditionally developed for natural language texts, have also been

*These authors contributed equally to this work.

used for tokenizing protein sequences [Bepler and Berger, 2021, Tan et al., 2023, Dotan et al., 2024, Ieremie et al., 2024]. Subword tokenization methods have proven successful in NLP when handling rare words and improving model efficiency [Sennrich et al., 2016]. However, these methods present unique challenges when applied to protein sequences. Unlike textual languages, proteins have complicated structural properties and complex long-range dependencies that often cannot be adequately captured by traditional tokenization methods [Suyunu et al., 2024].

Byte-pair encoding (BPE) [Sennrich et al., 2016], a popular subword tokenization technique, exemplifies these limitations. Typically used in NLP, BPE begins by treating each character as a distinct token and iteratively merges the most frequent token pairs until reaching a predefined vocabulary size. However, this approach does not inherently respect the domain-specific boundaries and biological distinctions critical to protein sequence analysis [Suyunu et al., 2024]. Accurately representing protein sequences is challenging due to their intricate, evolutionarily dynamic nature. Proteins exhibit complex evolutionary relationships where functionally similar sequences diverge through mutations—a level of complexity often underestimated by conventional tokenization techniques.

Previous studies have underscored the limitations of traditional tokenization strategies in capturing the structural and functional properties of protein sequences [Vig et al., 2020, Suyunu et al., 2024]. Transformer-based language models have shown immense potential for advancing protein sequence representation and prediction [Brandes et al., 2022, Lin et al., 2023, Elnaggar et al., 2023]. For example, Rao et al. [2020] and Lin et al. [2023] demonstrated that deep learning models can reveal fundamental evolutionary and structural insights in protein sequences, emphasizing the need for advanced tokenization approaches.

The core motivation for our research lies in recognizing that protein sequences are dynamic, evolutionarily driven systems with intricate interconnections among sequence variants. We propose evoBPE, a novel tokenization method that integrates evolutionary mutation patterns into sequence segmentation. The innovation of evoBPE lies in its use of biologically informed substitution principles to evaluate token mutations. EvoBPE transcends commonly used frequency-based methods by leveraging established substitution matrices, offering a mutation-aware framework that captures evolutionary relationships among protein sequences.

The proposed method makes several significant contributions to protein sequence analysis. First, it introduces a biologically informed approach to sequence segmentation that preserves relationships among evolutionarily related proteins. Second, evoBPE establishes a tokenization framework capable of tracing the “genealogy” of tokens, offering more profound insights into the evolutionary trajectories of protein sequences. Third, it performs better domain conservation and mutated sequence embedding similarity analysis, outperforming BPE across various vocabulary sizes.

By bridging computational linguistics and molecular biology, evoBPE offers a robust tool for protein sequence representation, enabling advancements in function prediction, structural modeling, and evolutionary analysis. This biologically informed tokenization approach opens new possibilities for processing and understanding protein sequences.

2 Materials and Methods

2.1 Dataset

For this study, we utilized protein sequences from the human taxonomy (Taxon ID: 9606) obtained from UniProtKB and directly downloaded from the UniProt website. The decision to focus specifically on the human taxonomy, rather than employing a broader or randomly sampled dataset, was driven by several factors: First, proteins within the human taxonomy share more common characteristics than a random set of proteins from diverse species. This situation increases the likelihood of identifying biologically meaningful tokens during tokenization. While this advantage applies to both our proposed evoBPE method and standard BPE, it offers a controlled environment for evaluating the strengths and weaknesses of each approach. Second, focusing on human proteins simplifies the validation and interpretation of identified tokens. In a dataset comprising proteins from diverse taxa, tracing the origin of a potentially meaningful token becomes challenging, as it may correspond to multiple unrelated functional or structural regions. By narrowing the scope to human proteins, we mitigate this complexity, enabling a more precise assessment of the biological relevance of the tokens identified by the tokenization methods.

Despite its advantages, the UniProtKB dataset contains significant redundancy, with many protein groups exhibiting high sequence similarity. This redundancy skews tokenization algorithms towards overrepresented proteins, generating tokens that are hundreds of amino acids in length. To address this issue, we used the UniRef50 dataset instead of the complete UniProtKB dataset. Using UniRef50, we minimized the overrepresentation of evolutionarily similar sequences and obtained a more diverse and representative dataset.

To extract human protein sequences from the UniRef datasets, we applied the following procedure: For each cluster, if the representative protein belonged to the human taxonomy, it was directly selected as the cluster’s representative

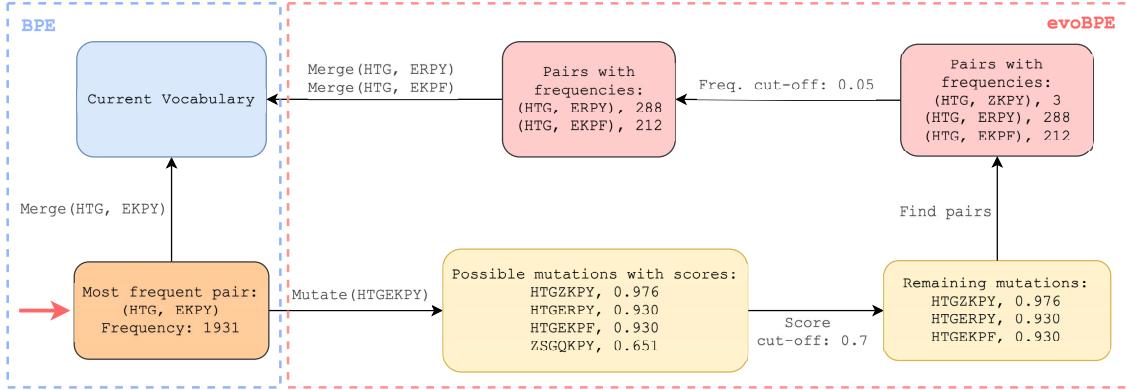


Figure 1: Diagram describing an example of a single iteration of evoBPE. While BPE would add only the token generated by merging the most frequent pair, evoBPE also generates mutations of that token and adds them if they meet the criteria.

sequence. In cases where the representative protein was not of human origin, we searched the cluster for human proteins and selected one to represent the cluster. We excluded isoforms to ensure dataset uniformity. This approach provided comprehensive coverage of human proteins while preserving the clustering characteristics of the UniRef datasets.

In addition to protein sequences, we incorporated domain information to utilize in the pre-tokenization step and experiments. Protein domains are distinct structural or functional units that fold independently and often recur across different proteins with sequence variations. Domain annotations were primarily obtained from InterPro [Blum et al., 2024], a comprehensive database integrating multiple sources. We utilized The Encyclopedia of Domains (TED) [Lau et al., 2024] as a supplementary source, a recent resource that consolidates domain annotations from three independent methods applied to AlphaFold-predicted structures. TED was particularly valuable for cases where InterPro did not provide annotations.

2.2 Pre-tokenization

In NLP, pre-tokenization is crucial in breaking down text into manageable and semantically meaningful units. For instance, in languages like English, whitespace characters and punctuation provide natural boundaries, enabling effective segmentation of text into words. This pre-tokenization step improves the semantic coherence of the resulting tokens by aligning them with linguistic boundaries.

Protein sequences lack such inherent segmentation cues, posing a unique challenge for pre-tokenization. Traditional approaches often treat entire protein sequences as single units, akin to processing a long, unbroken word. While straightforward, this approach risks losing the structural and functional granularity essential for biological insights.

We adopt a pre-tokenization strategy segmenting protein sequences based on domain boundaries to address this. Analogous to semantic units in NLP, domains serve as biologically meaningful boundaries, allowing us to pre-tokenize sequences into segments that reflect functional and structural realities. While pre-tokenizing a protein sequence, when discrepancies or overlaps arose between domain annotations from different InterPro sources for the same protein, we selected the domain set covering the most amino acids without overlaps.

2.3 evoBPE

We propose a novel tokenization approach, evoBPE, which builds upon the standard BPE algorithm by introducing a mutation-driven enhancement. The pseudo-code for EvoBPE is outlined in Algorithm 1. In each iteration of the standard BPE process, evoBPE not only merges the most frequent pair in the dataset but also generates additional candidate pairs using biologically informed mutations. evoBPE simulates evolutionary mutations of the most frequent pair to create candidate pairs via employing substitution matrices. For each amino acid in the selected pair, the algorithm identifies substitution amino acids with non-negative scores from the substitution matrix. The algorithm then generates candidate sequences from these substitutions and evaluates them based on their pairwise alignment scores against the original pair. Sequences exceeding a predetermined alignment score threshold are considered for further processing.

The algorithm evaluates all possible splits of these candidate sequences within the dataset. Candidate pairs that appear in the dataset with a frequency above a specified threshold relative to the most frequent pair are incorporated into the vocabulary. This approach ensures that only biologically relevant and sufficiently frequent mutations are included in the final vocabulary.

An iteration of evoBPE training is best explained with an example taken from an actual training process of evoBPE: Consider the example described in Figure 1. At iteration i , the most frequent pair in the dataset is (HTG, EKPY) with a frequency of 1931. In traditional BPE training, after this pair is merged and added into the vocabulary, iteration $i + 1$ would start. However, evoBPE, on top of adding HTGEKPY, generates 4 more candidate mutations for it (this number is exponentially more in practice): HTGZKPY, HTGERPY, HTGEKPF, and ZSGQKPY. Alignment score cut-off parameter 0.7 shaves off the candidate mutations with low scores and ZSGQKPY gets eliminated at this step. The remaining mutations are seemingly plausible but to add them to the vocabulary, there should be token pairs in the dataset such that when they are merged, they generate these mutated token strings. After finding suitable pairs for each of the three mutations, due to the frequency cut-off parameter (0.05), pairs that have a frequency less than $1931 * 0.05 = 96.55$ are eliminated. This removes the pair (HTG, ZKPY) as it only has a frequency of 3. After all the elimination steps, the remaining pairs are (HTG, ERPY) and (HTG, EKPF). They are merged, and the tokens HTGERPY and HTGEKPF are added to the vocabulary.

The mutation process follows a hierarchical structure that mirrors biological evolutionary relationships. When a frequently occurring pair yields viable mutations, we designate it as a “parent” sequence. The mutations generated from this parent are termed “child mutations” or “siblings” in relation to each other. The complete set of a parent and its child mutations constitutes a “family.” This familial nomenclature provides a clear organizational structure and reflects the evolutionary relationships between related sequences.

Furthermore, since BPE constructs its vocabulary through the iterative merging of previously added tokens, we can trace the lineage of mutation families throughout the tokenization process. This genealogical tracking capability provides valuable insights into the evolutionary patterns and relationships between different token families, potentially offering a deeper understanding of the protein sequence patterns and their variations.

Our implementation utilizes a max heap map data structure to efficiently store and retrieve pairs and their occurrences, with each heap item maintaining references to all its occurrences in sequences. Searching in the mutation space is very costly because the number of mutations scales exponentially with the length of the sequence. We employ a depth-first search algorithm with pruning to optimize the search for viable mutations. It is possible to search the mutation space starting from mutations with the highest scores, and since the alignment score cut-off has a predefined value, as soon as a branch reaches a score that is less than the cut-off, that branch can be pruned. Even though the run-time is still exponential, this optimization significantly relaxes the time and space complexity of the training.

The primary objective of evoBPE is to capture biologically related sequences that may have diverged due to mutations over time yet still share similar functional or structural roles. By doing so, evoBPE enhances the ability to identify motifs, domains, and other conserved structures.

3 Experiments and Results

3.1 Experimental Setup

UniRef50 human dataset is used as both training and test dataset in all experiments. If it is stated that a given model is trained on the pre-tokenized dataset, it means that it is also tested on the pre-tokenized dataset and both of them are derived from UniRef50 human dataset.

The experiments presented in this section are divided into two main categories: one focusing on preliminary analysis, while the other we consider as the main experiments. In the former category, we provide comparisons and insights on the segmentation behavior and vocabulary statistics of various evoBPE models, as well as an analysis of their adherence to linguistic laws. In the latter, there are two experiments. The first experiment evaluates the performance of evoBPE models by measuring how consistently they segment variants of the same domain. The second experiment provides a comparative analysis on the ESM2 embeddings of sequences that are mutated using evoBPE vocabularies versus random substitutions.

3.1.1 evoBPE Hyper Parameters

Unless stated otherwise, evoBPE models have been trained with an alignment score cut-off of 0.7 and frequency cut-off of 0.05. As an exception, evoBPE model using PAM250 uses the alignment score cut-off value of 0.8 because for smaller values, it allows too many mutations to compute feasibly. BLOSUM62, BLOSUM45, PAM70, and PAM250

Algorithm 1 evoBPE Tokenization Algorithm

Require: Dataset D with protein sequences, Vocabulary size V , Substitution matrix S , Frequency cut-off f_{th} , Alignment score cut-off a_{th}

Ensure: Tokenized vocabulary V_{evoBPE}

- 1: Initialize V_{evoBPE} as set of unique characters in D
- 2: Initialize max-heap H with frequencies of all pairs in D
- 3: **while** $|V_{evoBPE}| < V$ **do**
- 4: Extract most frequent pair $p = (x, y)$ from H
- 5: Add p to V_{evoBPE}
- 6: **for** each amino acid a_i in p **do**
- 7: Find substitutions a_j such that $S(a_i, a_j) \geq 0$
- 8: **end for**
- 9: Generate all substitution sequences p' of p using a_j
- 10: **for** each p' **do**
- 11: Compute alignment score, $score(p, p')$
- 12: **if** $score(p, p') \geq a_{th}score(p, p)$ **then**
- 13: Compute frequency of p' in D , $freq(p')$
- 14: **if** $freq(p') \geq f_{th}freq(p)$ **then**
- 15: Add p' to V_{evoBPE}
- 16: **end if**
- 17: **end if**
- 18: **end for**
- 19: Update H with new pairs and their frequencies
- 20: **end while**

were used as the substitution matrices. For the vocabulary sizes, sizes ranging from 800 to 25600 were used in experiments where all values in between are power of 2 multiples of 800. Mutations for tokens shorter than 3 and longer than 12 are not generated. The latter is due to the time complexity of mutation search, whereas the latter is because most of the 2 letter tokens were already getting included in vocabularies without the mutations.

Alignment score cut-off parameter has a very strong impact on the size of the search space of mutations. The value 0.7 provided a good trade-off between the time complexity and mutational variations in the learned vocabularies. As this cut-off value gets closer to 1, the behavior of evoBPE models starts being very similar to BPE.

Frequency cut-off parameter is selected as 0.05 by trial and error during development phase of the algorithm. For our purposes, it does not make sense to add all possible mutations of a parent, if those mutations have very little frequencies in the training dataset since in that case, the chance of them being just random coincidences rather than being biologically meaningful mutations is higher.

3.2 Preliminary Analysis

3.2.1 General Tokenizer Statistics

A general overview of various statistics of the tokenizers is provided in Table 1.

In all models, the average token length in the vocabulary is significantly higher than the average token length in the test dataset. This is not surprising as the longer tokens are generally added in the last iterations of the training, and therefore, they should have less frequency in the dataset. The application of pre-tokenization before training does not seem to impact the statistics very noticeably. Token lengths of evoBPE models using PAM70 and BLOSUM62 are very similar, but their mutated and parent token percentages are consistently different. PAM70 is more conservative with the number of mutations it allows for each parent and has a higher ratio of parents. The average token lengths of evoBPE models are consistently lower than BPE as they populate the limited vocabulary space with mutations, whereas BPE has space for adding longer and longer tokens since it ignores low-frequency mutations.

3.2.2 Effect of Substitution Matrices

The vocabularies and consequent segmentations resulting from the evoBPE algorithm are influenced by the substitution matrix used to generate the mutations. To better observe to what extent this influence occurs, an experiment has been performed in which we compare the resulting segmentations on the test dataset from various tokenization algorithms.

Table 1: Statistics of BPE and evoBPE models.

Model	pre	sMatrix	vSize	pRatio	pRatio*	mRatio	mRatio*	avg tLength	avg tLength*
BPE	No	-	800	1.0	1.0	0.0	0.0	2.52 ± 0.60	1.95 ± 0.57
BPE	No	-	6400	1.0	1.0	0.0	0.0	3.43 ± 0.71	2.46 ± 0.71
BPE	No	-	25600	1.0	1.0	0.0	0.0	3.94 ± 1.52	2.84 ± 0.94
evoBPE	No	PAM70	800	0.09	0.04	0.49	0.08	2.58 ± 0.62	1.90 ± 0.58
evoBPE	No	PAM70	6400	0.15	0.12	0.79	0.27	3.35 ± 0.68	2.40 ± 0.68
evoBPE	No	PAM70	25600	0.11	0.16	0.87	0.41	3.89 ± 1.01	2.75 ± 0.85
evoBPE	No	BLOSUM62	800	0.06	0.03	0.52	0.09	2.60 ± 0.63	1.89 ± 0.58
evoBPE	No	BLOSUM62	6400	0.10	0.10	0.84	0.30	3.36 ± 0.67	2.40 ± 0.68
evoBPE	No	BLOSUM62	25600	0.07	0.13	0.91	0.44	3.89 ± 0.79	2.75 ± 0.83
BPE	Yes	-	800	1.0	1.0	0.0	0.0	2.51 ± 0.59	1.94 ± 0.56
BPE	Yes	-	6400	1.0	1.0	0.0	0.0	3.4 ± 0.71	2.44 ± 0.71
BPE	Yes	-	25600	1.0	1.0	0.0	0.0	3.90 ± 1.19	2.82 ± 0.91
evoBPE	Yes	PAM70	800	0.08	0.03	0.49	0.08	2.59 ± 0.65	1.89 ± 0.58
evoBPE	Yes	PAM70	6400	0.15	0.12	0.79	0.27	3.35 ± 0.68	2.39 ± 0.68
evoBPE	Yes	PAM70	25600	0.11	0.16	0.87	0.40	3.89 ± 0.95	2.74 ± 0.86
evoBPE	Yes	BLOSUM62	800	0.06	0.03	0.52	0.09	2.6 ± 0.63	1.88 ± 0.59
evoBPE	Yes	BLOSUM62	6400	0.10	0.10	0.84	0.29	3.33 ± 0.65	2.39 ± 0.68
evoBPE	Yes	BLOSUM62	25600	0.07	0.13	0.91	0.44	3.88 ± 0.94	2.73 ± 0.86

Variables include pre-tokenization usage (pre), substitution matrix (sMatrix), vocabulary size (vSize), and metrics: parent token ratios (pRatio, pRatio*), mutated token ratios (mRatio, mRatio*), and average token lengths (avg tLength, avg tLength*)

Note: Asterisk (*) indicates the metric is calculated from the test dataset after running the model instead of from the vocabulary itself.

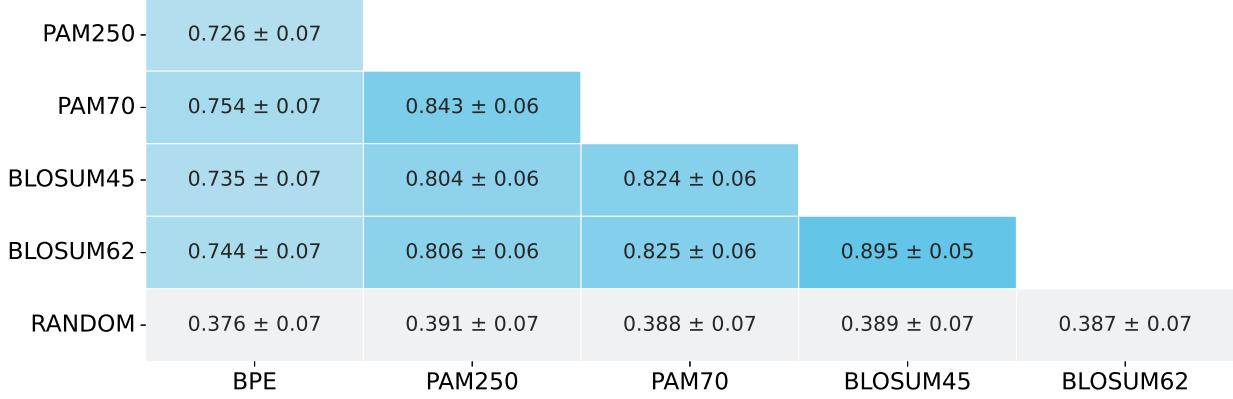


Figure 2: Average Dice coefficient between RANDOM, BPE, and evoBPE tokenizers (vocabulary size = 25600)

evoBPE (which has been trained using various substitution matrices such as BLOSUM62 or PAM70), BPE, and RANDOM models were used as tokenizers. To evaluate the similarity between the segmentations produced by these tokenizers, the Dice-Sørensen coefficient was used as the primary metric.

The Dice-Sørensen coefficient is commonly used in segmentation tasks in computer vision and NLP. It is a form of F1 measure when the terms are stated as a “prediction task”. It calculates the similarity between two sets. Given a protein sequence S and two tokenizers T_1 and T_2 , let $T_1(S)$ and $T_2(S)$ represent the segmentations generated by these tokenizers. These representations, expressed as sets of indices marking token boundaries, are unique to each segmentation because the indices correspond to the same sequence, and the tokens are non-overlapping. The Dice coefficient between these sets is calculated as:

$$Dice(T_1(S), T_2(S)) = \frac{2 * |T_1(S) \cap T_2(S)|}{|T_1(S)| + |T_2(S)|} \quad (1)$$

RANDOM refers to a simple procedure in which the sequence is segmented at random positions such that in the resulting segmentation, the mean and the standard deviation of the token lengths throughout the whole dataset match that of the tokenizer that is being compared to the RANDOM. This is achieved by sampling the “next token length” from a Gaussian distribution with these mean and standard deviation values until the end of the sequence is reached.

In Figure 2, all tokenizers are compared pairwise using a fixed vocabulary size of 25600. For each sequence in the dataset, the Dice coefficient is calculated between the token boundaries generated by the two tokenizers, and the values are averaged across all sequences.

RANDOM comparisons in Figure 2 serve as a baseline for the Dice coefficient metric. A random segmentation with the same token length distribution as the tokenizer under comparison achieves an agreement of approximately 0.38. Although it is possible for Dice coefficient value to be 0 or near 0, it requires more extreme divergence in the behavior of the two tokenizers. We also calculated dice coefficient between two different segmentations by the same RANDOM tokenizer and it also yielded a value of approximately 0.38. At vocabulary size 25600, we see from Table 1 that both evoBPE and BPE tokenizers segment the sequences in the test dataset into tokens of length approximately 2.75 ± 0.8 . This, combined with the comparisons with RANDOM tokenizer, suggests that any two segmentations with such token length distributions would yield a Dice coefficient value of at least 0.38, except for some edge cases.

In Figure 2, we see that, although evoBPE tokenizers agree with each other more than they agree with BPE, those Dice coefficient values are still around 0.8 in most cases. This implies that the substitution matrix that was used in the training of the evoBPE had a rather significant effect on the resulting segmentation behavior of the tokenizer.

3.2.3 Adherence to Linguistic Laws

Linguistic laws provide a framework for understanding structural patterns in language, and when applied to protein sequences, they offer valuable insights into the underlying organizational principles [Suyunu et al., 2024]. In this experiment, we examined how different tokenization methods conform to key linguistic laws, specifically Zipf’s law and Brevity law.

Zipf’s law states that the frequency of a particular element is inversely proportional to its rank [Zipf, 1949]. To observe this law, we plotted the frequency of each token as a function of its frequency rank on a log-log scale, where an ideal line has a slope of -1 . Figure 3 illustrates the slopes of Zipf’s law plots for different tokenization methods across varying vocabulary sizes. Our analysis revealed that the BPE method demonstrated closer alignment with Zipf’s law than evoBPE. This deviation can be attributed to the fundamental modification in token frequency distribution introduced by evoBPE. Despite incorporating low-frequency mutation tokens, we observed that Zipf’s law graph did not exhibit extreme divergence, suggesting the robustness of our proposed method.

Brevity law suggests that frequently used tokens tend to be shorter [Zipf, 1949, Torre et al., 2019]. The evoBPE vocabulary, enriched with mutation-based tokens, exhibited a tendency towards relatively shorter tokens, while BPE generated a more uniform vocabulary in terms of token length shown in Figures 4a and 4b. This difference does not imply that evoBPE fails to adhere to Brevity law but rather highlights the nuanced approach of BPE in token length representation.

Notably, while evoBPE did not improve upon BPE’s adherence to linguistic laws, it achieved comparable compliance across most metrics despite its biological modifications to the fundamental frequency-based token selection. This suggests that evoBPE’s incorporation of evolutionary information preserves the underlying linguistic structure of protein sequences while potentially capturing additional biological relationships.

3.3 Experiments

3.3.1 Domain Conservation Analysis

Protein domains are structural units that maintain their function and structure across different proteins despite evolutionary variations in sequence and length. We designed an experiment to evaluate how different tokenization methods handle these evolutionary variations by measuring the similarity of tokenization patterns across domain variants (i.e., the variants of the same domain occurring in different proteins). Our hypothesis was that evoBPE, with its evolutionary-aware approach, would demonstrate more consistent tokenization patterns across domain variants compared to standard BPE.

Methodology: We extracted domain sequences from our protein dataset and grouped them by their Interpro annotations. To control for length variations, we created paired subgroups within each domain group, allowing for a maximum length difference of 5 amino acids between pairs. Domains without suitable length-matched partners were excluded from the analysis. For standardization, we first performed pairwise sequence alignment on the domain pairs

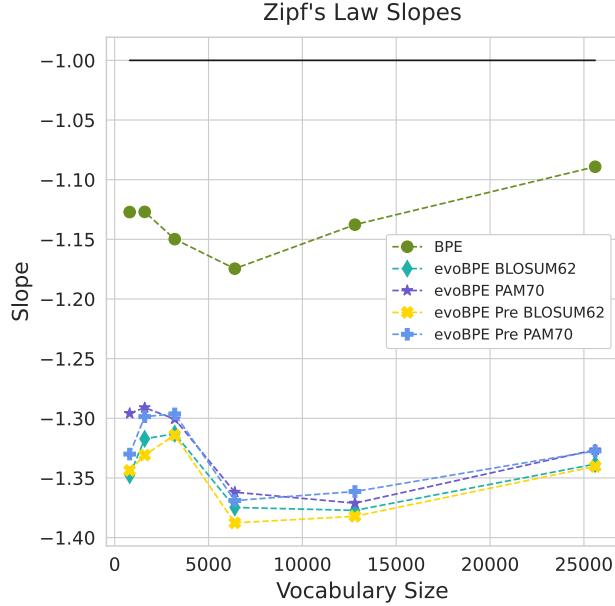


Figure 3: The slope values for Zipf's law plots of BPE and various evoBPE models. The Pre keyword in the naming means pre-tokenization is applied. -1 is the ideal slope value.

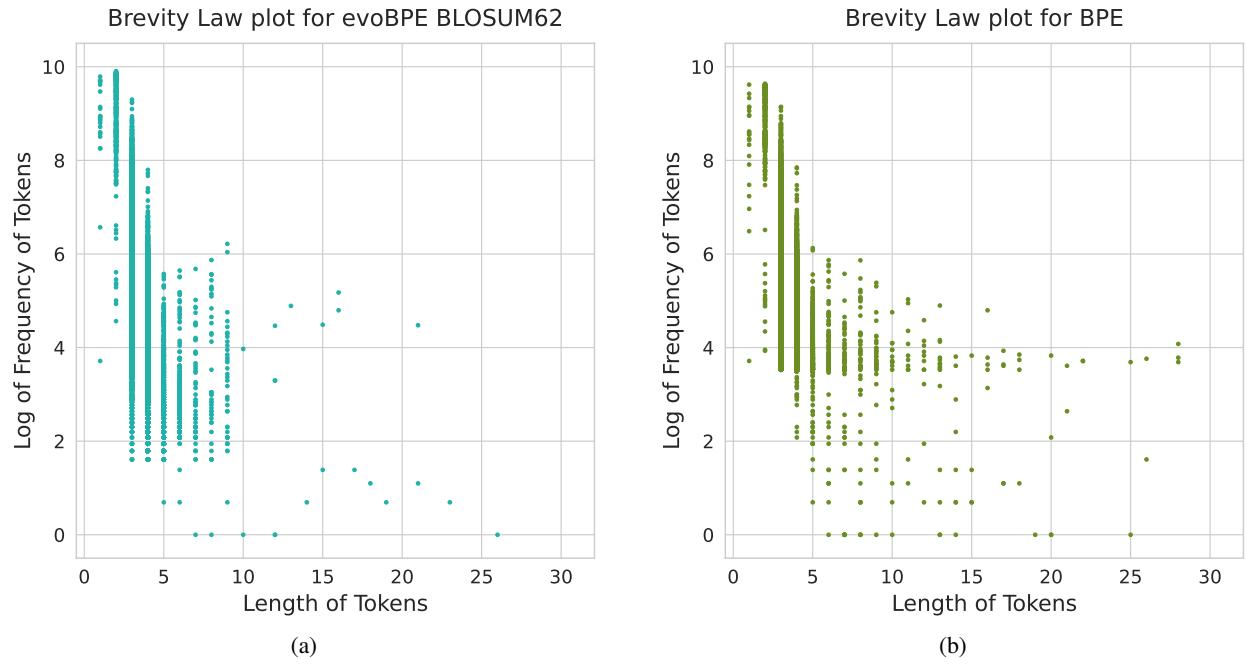


Figure 4: Brevity law plots of evoBPE BLOSUM62 and BPE for the vocabulary size of 25600.

to ensure a consistent format for comparisons. We then tokenized each domain in the paired sets. The tokenization patterns were compared by calculating the Dice coefficient between token sets for each domain pair. To assess overall tokenizer performance, we computed mean Dice coefficients across all pair groups. This metric serves as our primary performance indicator, where higher Dice coefficients suggest better preservation of tokenization patterns across evolutionary variants and stronger capture of biological relationships. You can see the diagram of the method in Figure 7, where we apply the procedures through IPR000328 Interpro domain annotation.

Results: The analysis revealed that evoBPE methods consistently outperformed standard BPE by a significant margin while showing minimal variation among different evoBPE models, as seen in Figure 5. At smaller vocabulary sizes, we observed higher and more closely clustered scores across methods, which can be attributed to shorter tokens and more similar vocabularies. As vocabulary size increased, the performance gap between evoBPE models and BPE widened, suggesting that evoBPE successfully incorporated biologically relevant tokens into its vocabulary. This performance differential suggests that evoBPE’s mutation-aware approach better captures the underlying biological patterns in protein domains. The consistent superiority of evoBPE across different vocabulary sizes supports our hypothesis that incorporating evolutionary information through substitution matrices enhances the tokenizer’s ability to handle sequence variations in related proteins.

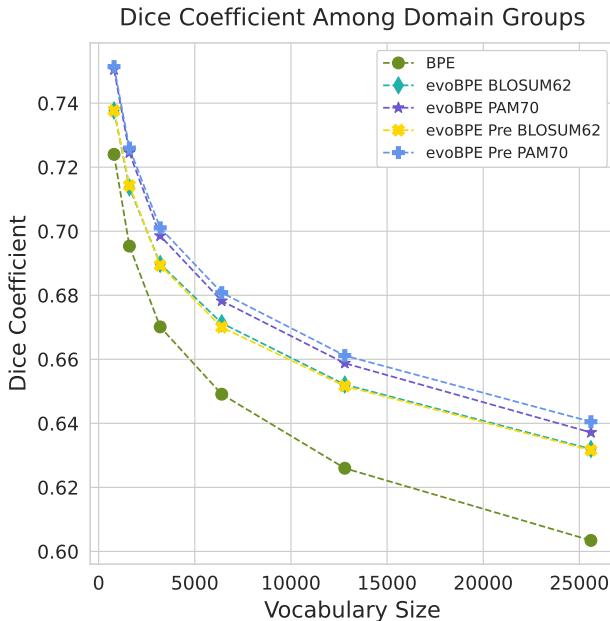


Figure 5: Dice Coefficient among domain groups for BPE and various evoBPE models. The Pre keyword in the naming means pre-tokenization is applied.

3.3.2 ESM-2 Embedding Similarity Analysis for Mutations

ESM-2 [Lin et al., 2023] is a state-of-the-art protein language model that learns contextual representations of proteins through self-supervised learning on millions of protein sequences. The model’s embeddings capture both structural and evolutionary information of proteins, demonstrating remarkable performance in various downstream tasks such as structure prediction and function annotation. In this experiment, we utilized the 650M parameter version of ESM-2 to evaluate the biological relevance of mutation-based token replacements in evoBPE. This experimental design allows us to quantitatively assess whether mutation-based token replacements better preserve the biological properties of the original sequence compared to arbitrary substitutions despite the relative rarity of mutation tokens in the training data.

Methodology: The experimental procedure consisted of generating and comparing three versions of each sequence: original, mutated, and alternative. The original version is the as-is version of the sequence in the dataset. To generate the mutated version, we first tokenized each sequence in the dataset with evoBPE to identify parent and mutation tokens. Then, we replaced parent tokens with their first child mutation and mutation tokens with their first sibling mutation. Tokens outside mutation families remained unchanged. To establish a baseline for comparison, we generated alternative sequences by introducing random amino acid substitutions at positions corresponding to the mutation-based changes. These alternative substitutions were constrained to exclude both the original amino acid and its mutation-based replacement while ensuring the resulting tokens existed in the evoBPE vocabulary. We employed ESM-2 to

generate embeddings for each sequence variant, then retrieved the embeddings where amino acid substitutions occurred. The embedding similarities between the original-mutated pairs and original-alternative pairs were quantified using cosine similarity metrics. This process was repeated across the entire dataset to obtain statistically robust results. You can see the diagram showing an example of the ESM-2 embedding similarity analysis process in Figure 8.

Results: Analysis of the results reveals several key insights. Across all models and vocabulary sizes, mutated sequences consistently show higher similarity to the original sequences compared to alternative sequences, as shown in Figure 6. While higher similarity scores at lower vocabulary sizes are expected due to fewer amino acid modifications, a particularly noteworthy observation emerges as vocabulary size increases. Despite the increasing number of modifications reflected in alternative sequences, mutation sequences maintain high similarity scores, leading to a widening gap between mutated and alternative sequence similarities. This demonstrates the quality and effectiveness of mutation tokens added through the evoBPE algorithm at larger vocabulary sizes. Comparing different evoBPE models, we can attribute the variations in similarity scores to the specific sequence modifications made by each tokenizer. While differences are subtle, focusing on the gap between mutated and alternative sequences for each tokenizer reveals that PAM70 outperforms BLOSUM62, indicating its ability to generate higher-quality mutations (Figure 6b). We don't observe a particular advantage of pre-tokenization for this experiment.

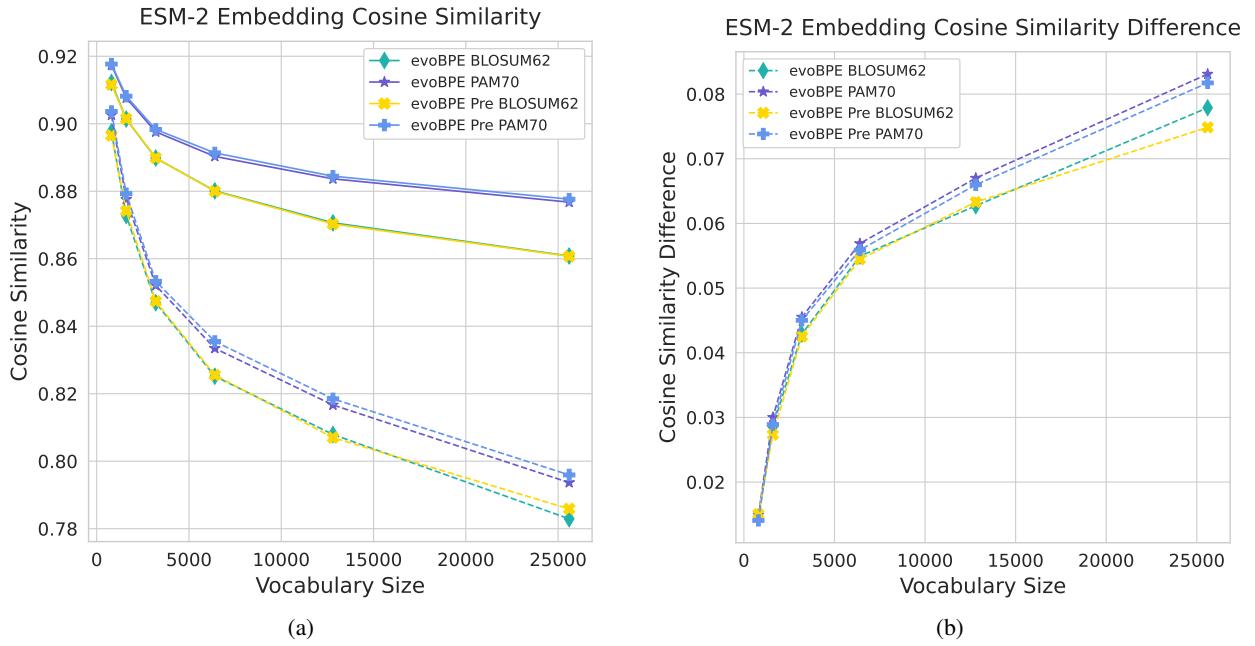


Figure 6: Plots for the cosine similarity analysis on ESM-2 embeddings for BPE and various evoBPE models. The Pre keyword in the naming means pre-tokenization is applied. 6a Continuous lines and dashed lines represent mutated-original and alternative-original cosine similarity scores, respectively. 6b Shows the cosine similarity difference between mutated-original and alternative-original scores.

4 Conclusion and Discussion

The evoBPE method represents a significant advancement in protein sequence tokenization, bridging the critical gap between computational linguistics and molecular biology. By introducing an evolutionary mutation-aware approach to sequence segmentation, our research offers a novel perspective on how protein sequences can be more intelligently parsed and analyzed.

The key innovation of evoBPE lies in its ability to transcend traditional frequency-based tokenization methods. Unlike standard BPE, which treats protein sequences as static linguistic structures, our approach acknowledges the dynamic, evolutionarily driven nature of protein sequences. By leveraging established substitution matrices like BLOSUM62 and PAM70, evoBPE generates token mutations that capture the nuanced biological relationships underlying protein sequence variations.

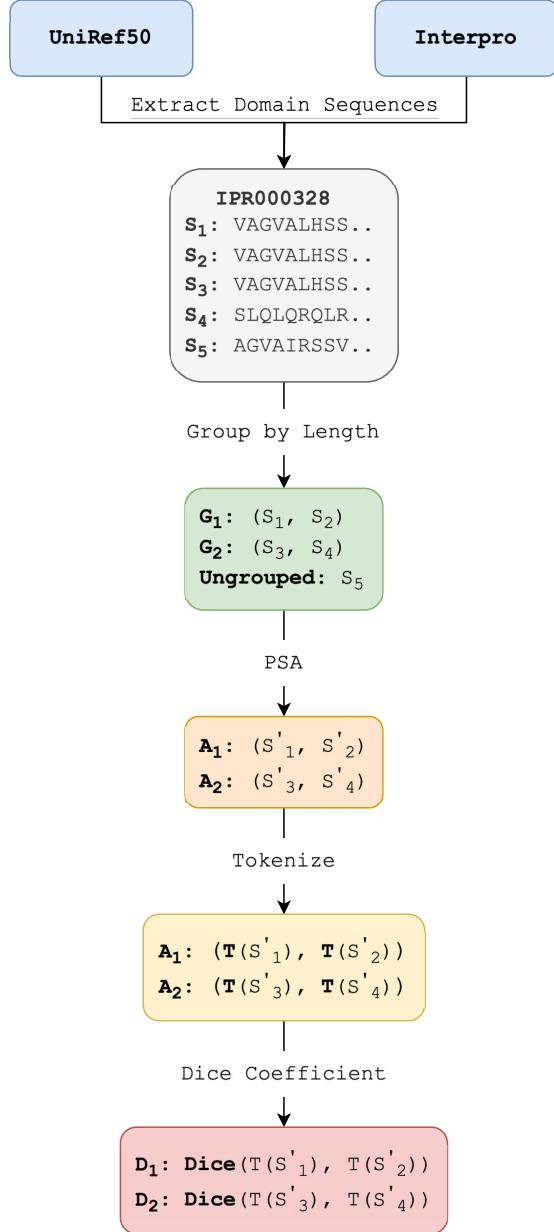


Figure 7: Diagram for the domain conservation analysis methodology applied through IPR000328 Interpro domain annotation. First, we extract IPR000328 domain sequences from the corresponding proteins. Then, we pair sequences by their lengths, allowing a maximum length difference of 5 amino acids where sequence S_5 is left out. We apply pairwise sequence alignment (PSA) to paired sequences and tokenize the aligned versions. Finally, we calculate the Dice coefficient between tokenized pairs.

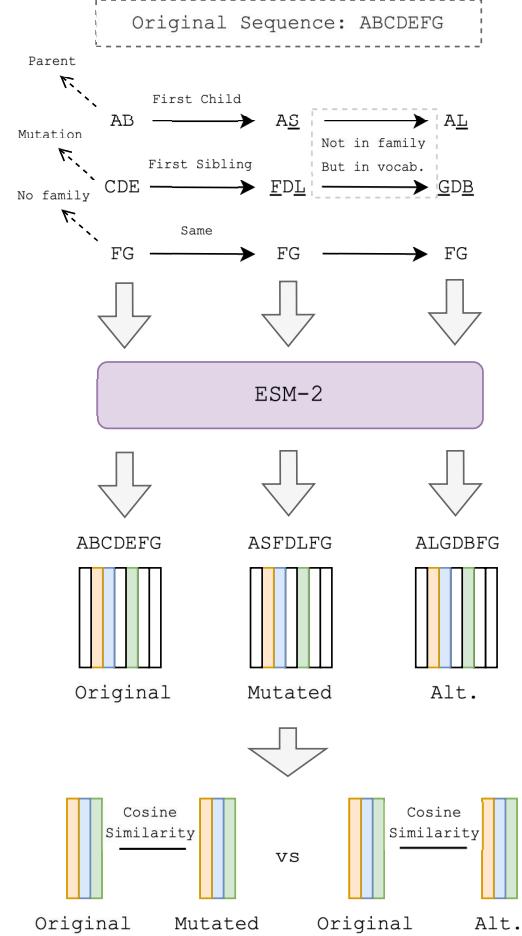


Figure 8: Diagram showing an example for ESM-2 embedding similarity analysis process. First, the original sequence is tokenized. Parent and mutation tokens are identified. The mutated sequence is generated by swapping the parent tokens with their first child and the mutation tokens with their first sibling. Amino acids where a substitution has occurred are underlined. In order to generate the alternative sequence, random substitutions are explored at these underlined positions. The generated token belongs to the vocabulary but does not belong to the same family as the original token. Meaning that AL is in the vocabulary but is not in the same family as AS and AB. The same is true for GDB. Embeddings for original, mutated, and alternative sequences are computed. Only the embeddings of the amino acids at the underlined positions are taken. The cosine similarity of these embeddings is calculated between original and mutated sequences as well as between original and alternative sequences.

Our experimental results demonstrate the method’s robust performance across multiple analytical dimensions. The domain conservation analysis revealed that evoBPE consistently outperforms standard BPE, particularly as vocabulary size increases. This suggests that our mutation-aware approach more effectively captures the underlying biological patterns in protein domains. The widening performance gap at larger vocabulary sizes indicates that evoBPE successfully incorporates biologically relevant tokens that reflect evolutionary relationships.

The ESM-2 embedding similarity analysis provided further validation of our approach. Mutated sequences generated through evoBPE maintained higher similarity to original sequences compared to random substitutions. This finding is particularly significant, as it quantitatively demonstrates that our mutation-based token replacements preserve critical biological properties more effectively than arbitrary amino acid substitutions.

While our method did not dramatically improve adherence to linguistic laws compared to BPE, comparable compliance suggests that the incorporation of evolutionary information does not disrupt the fundamental linguistic structure of protein sequences. This balanced approach represents a nuanced contribution to the field, showing that biological insights can be integrated without fundamentally altering established computational linguistic principles.

Our attempt to apply pretokenization by segmenting the sequences at known domain boundaries did not produce noticeable effects in our experiments and analysis. Therefore, it is yet unclear as to how effective this method might prove to be in the general task of tokenizing protein sequences.

The research also highlights the potential of interdisciplinary approaches in computational biology. By drawing parallels between linguistic tokenization and protein sequence analysis, we demonstrate how methodological innovations from one domain can provide transformative insights in another. The evoBPE method is not merely a technical improvement but a conceptual bridge that reimagines protein sequences as dynamic, evolving linguistic systems.

Future research directions could explore expanding the method to broader taxonomic datasets, investigating its performance across different protein families, and integrating more sophisticated evolutionary models. The current implementation provides a promising framework for more biologically informed sequence representation, with potential applications in protein function prediction, structural modeling, and evolutionary analysis.

In conclusion, evoBPE represents a significant step towards more intelligent and biologically sensitive tokenization methods. By embedding evolutionary mutation principles into sequence segmentation, we offer a powerful tool that promises to enhance our understanding of protein sequence complexity and variation.

5 Competing interests

No competing interest is declared.

6 Author contributions statement

B.S., Ö.D., and A.Ö. conceived and designed the study. B.S. and Ö.D. implemented the algorithms, conducted the experiments, and wrote the manuscript. B.S., Ö.D., and A.Ö. analyzed the results and reviewed the manuscript.

7 Acknowledgments

This work is supported by ERC grant (LifeLU, 101089287). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

8 Data Availability

All data underlying this work, including source code, is available at <https://github.com/boun-tabi-lifelu/evolutionary-substitution>

References

- T. Bepler and B. Berger. Learning the protein language: Evolution, structure, and function. *Cell systems*, 12(6):654–669, 2021.
- M. Blum, A. Andreeva, L. Florentino, S. Chuguransky, T. Grego, E. Hobbs, B. Pinto, A. Orr, T. Paysan-Lafosse, I. Ponamareva, G. Salazar, N. Bordin, P. Bork, A. Bridge, L. Colwell, J. Gough, D. Haft, I. Letunic, F. Llinares-

- López, A. Marchler-Bauer, L. Meng-Papaxanthos, H. Mi, D. Natale, C. Orengo, A. Pandurangan, D. Piovesan, C. Rivoire, C. A. Sigrist, N. Thanki, F. Thibaud-Nissen, P. Thomas, S. E. Tosatto, C. Wu, and A. Bateman. Interpro: the protein sequence classification resource in 2025. *Nucleic Acids Research*, 53(D1):D444–D456, 11 2024.
- N. Brandes, D. Ofer, Y. Peleg, N. Rappoport, and M. Linial. Proteinbert: a universal deep-learning model of protein sequence and function. *Bioinformatics*, 38(8):2102–2110, 2022.
- E. Dotan, G. Jaschek, T. Pupko, and Y. Belinkov. Effect of tokenization on transformers for biological sequences. *Bioinformatics*, 40(4):btae196, 2024.
- A. Elnaggar, M. Heinzinger, C. Dallago, G. Rehawi, Y. Wang, L. Jones, T. Gibbs, T. Feher, C. Angerer, M. Steinegger, et al. Prottrans: Toward understanding the language of life through self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127, 2021.
- A. Elnaggar, H. Essam, W. Salah-Eldin, W. Moustafa, M. Elkerdawy, C. Rochereau, and B. Rost. Ankh: Optimized protein language model unlocks general-purpose modelling. *arXiv preprint arXiv:2301.06568*, 2023.
- M. Heinzinger, A. Elnaggar, Y. Wang, C. Dallago, D. Nechaev, F. Matthes, and B. Rost. Modeling aspects of the language of life through transfer-learning protein sequences. *BMC bioinformatics*, 20:1–17, 2019.
- I. Ieremie, R. M. Ewing, and M. Nirajan. Protein language models meet reduced amino acid alphabets. *Bioinformatics*, 40(2):btae061, 2024.
- A. M. Lau, N. Bordin, S. M. Kandathil, I. Sillitoe, V. P. Waman, J. Wells, C. A. Orengo, and D. T. Jones. Exploring structural diversity across the protein universe with the encyclopedia of domains. *Science*, 386(6721):eadq4946, 2024.
- Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.
- A. Nambiar, M. Hefflin, S. Liu, S. Maslov, M. Hopkins, and A. Ritz. Transforming the language of life: transformer neural networks for protein prediction tasks. In *Proceedings of the 11th ACM international conference on bioinformatics, computational biology and health informatics*, pages 1–8, 2020.
- D. Ofer, N. Brandes, and M. Linial. The language of proteins: Nlp, machine learning & protein sequences. *Computational and Structural Biotechnology Journal*, 19:1750–1758, 2021.
- R. Rao, J. Meier, T. Sercu, S. Ovchinnikov, and A. Rives. Transformer protein language models are unsupervised structure learners. *Biorxiv*, pages 2020–12, 2020.
- A. Rives, J. Meier, T. Sercu, S. Goyal, Z. Lin, J. Liu, D. Guo, M. Ott, C. L. Zitnick, J. Ma, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021.
- R. Sennrich, B. Haddow, and A. Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, 2016.
- B. Suyunu, E. Taylan, and A. Özgür. Linguistic laws meet protein sequences: A comparative analysis of subword tokenization methods. In *2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 4489–4496. IEEE, 2024.
- Y. Tan, M. Li, P. Tan, Z. Zhou, H. Yu, G. Fan, and L. Hong. Peta: Evaluating the impact of protein transfer learning with sub-word tokenization on downstream applications. *arXiv preprint arXiv:2310.17415*, 2023.
- I. G. Torre, B. Luque, L. Lacasa, C. T. Kello, and A. Hernández-Fernández. On the physical origin of linguistic laws and lognormality in speech. *Royal Society open science*, 6(8):191023, 2019.
- J. Vig, A. Madani, L. R. Varshney, C. Xiong, R. Socher, and N. F. Rajani. Bertology meets biology: Interpreting attention in protein language models. *arXiv preprint arXiv:2006.15222*, 2020.
- G. K. Zipf. *Human behavior and the principle of least effort: An introduction to human ecology*. Addison-Wesley, 1949.