Advancing protein language models with linguistics: a roadmap for improved interpretability

Mai Ha Vu[1,C], Rahmad Akbar[2,☯], Philippe A. Robert[2,☯], Bartlomiej Swiatczak[3,☯], Geir Kjetil Sandve[4,*], Victor Greiff[2,*], Dag Trygve Truslew Haug[1,*,C]

[1] Department of Linguistics and Scandinavian Studies, University of Oslo, Norway
[2] Department of Immunology, University of Oslo and Oslo University Hospital, Norway
[3] Department of History of Science and Scientific Archeology, University of Science and Technology of China, China
[4] Department of Informatics, University of Oslo, Oslo, Norway
[☯] Equal contribution
[*] Joint supervision
[C] Correspondence: m.h.vu@iln.uio.no, d.t.t.haug@ifikk.uio.no

Abstract

Deep neural-network-based language models (LMs) are increasingly applied to large-scale protein sequence data to predict protein function. However, being largely blackbox models and thus challenging to interpret, current protein LM approaches do not contribute to a fundamental understanding of sequence-function mappings, hindering rule-based biotherapeutic drug development. We argue that guidance drawn from linguistics, a field specialized in analytical rule extraction from natural language data, can aid with building more interpretable protein LMs that have learned relevant domain-specific rules. Differences between protein sequence data and linguistic sequence data require the integration of more domain-specific knowledge in protein LMs compared to natural language LMs. Here, we provide a linguistics-based roadmap for protein LM pipeline choices with regard to training data, tokenization, token embedding, sequence embedding, and model interpretation. Combining linguistics with protein LMs enables the development of next-generation interpretable machine learning models with the potential of uncovering the biological mechanisms underlying sequence-function relationships.

1 Introduction

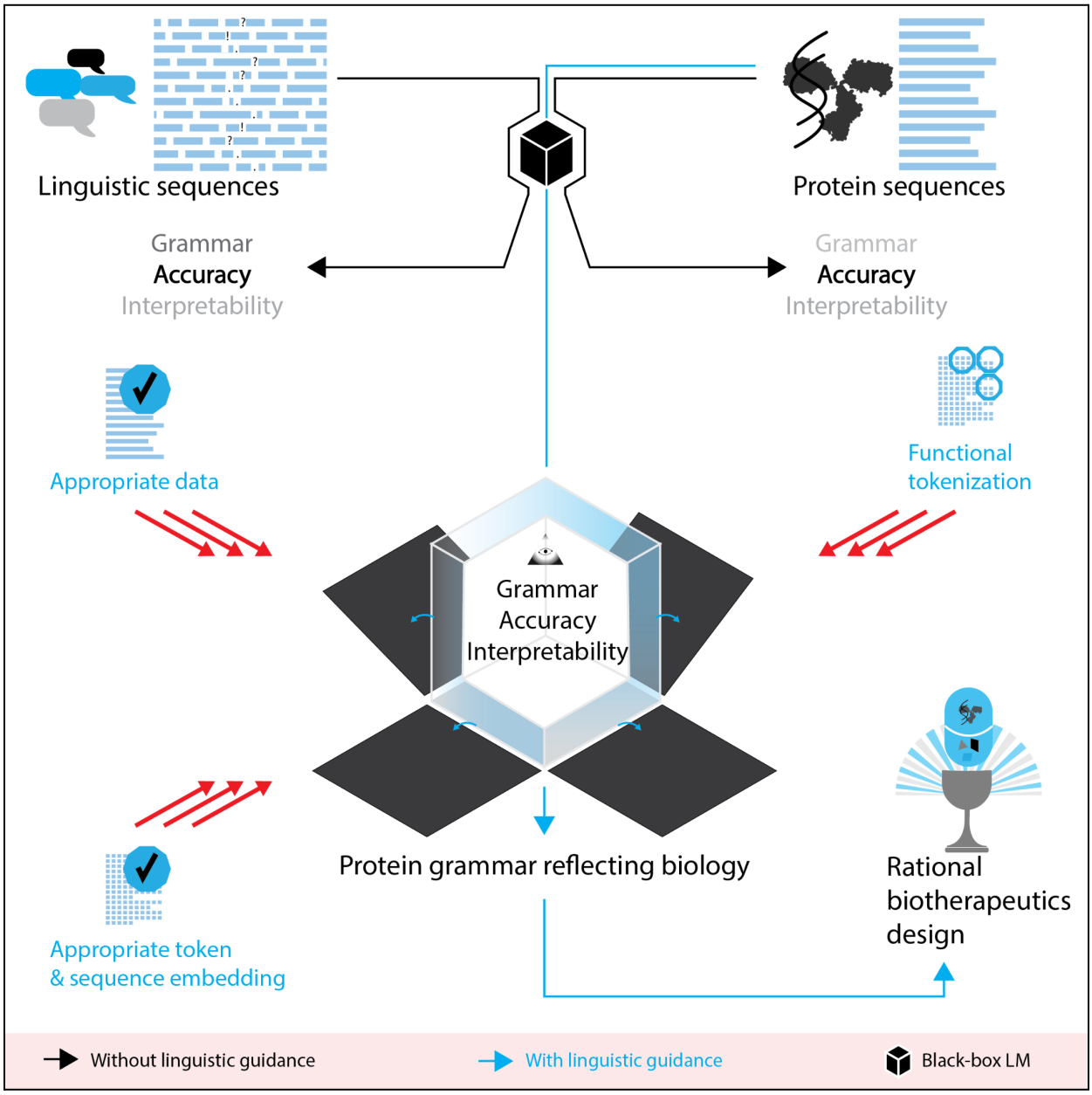

Figure 1 | **Graphical abstract. Advancing protein language models with linguistics: a roadmap for improved interpretability.** A direct application of LMs to protein sequences without any linguistic guidance in the design yields an opaque black-box model. While this protein model might perform with high Accuracy (defined as high performance on target task), it is unlikely to contain relevant protein Grammar (i.e., a generalization of proteins that matches biological reality, similar to natural language grammars as generalizations of linguistic sequences) and it remains low on Interpretability (i.e., a degree to which human users can understand the model and extract rules from it). In comparison, linguistic sequence models are more likely to learn a Grammar that matches independent linguistic analysis even without explicit linguistic guidance because linguistic data already contains structural indicators of basic linguistic units (e.g., punctuation, space), while protein sequence data does not. Even so, linguistic sequence models remain low on Interpretability without linguistic guidance, as domain knowledge is necessary to guide rule extraction from the model. An additional challenge that protein sequence modeling faces compared to linguistic sequence modeling is the absence of larger context beyond sequence. To remedy the disadvantages, in this Perspective we argue for linguistics-guided domain knowledge incorporation (appropriate pre-training data selection, tokenization, token and sequence embedding) into protein LMs. Namely, a thorough linguistic examination of natural language LM design can inform biologically appropriate protein LM design and can yield interpretable, glass-box (transparent) LMs with a protein grammar that reflects biological domain knowledge. Extracting this protein grammar would facilitate rational biotherapeutics design.

A growing number of studies apply machine learning tools called Language Models (LMs) (e.g., BERT (Devlin et al. 2019), RoBERTa (Y. Liu et al. 2019) and GPT (T. B. Brown et al. 2020)) to model biological sequence data (Bepler and Berger 2021; Ofer, Brandes, and Linial 2021; Alley et al. 2019; Brandes et al. 2022; Elnaggar et al. 2021; Heinzinger et al. 2019; Rao et al. 2019; B. L. Hie, Yang, and Kim 2022; Unsal et al. 2022; Rives et al. 2021; Meier et al. 2021; Y. Wang et al. 2019; M. Xu et al. 2022; Nijkamp et al. 2022). LMs are a primary tool in Natural Language Processing (NLP), a subfield of computer engineering applied to natural language, and they can be described as probability distributions over sequences of tokens (e.g., characters, words, subwords); alternatively, they might be called sequence models.

In contrast, linguistics, a field that studies natural language, uses primarily iterative and analytical methods to attain its research goal of describing natural language with rules that are understandable to humans (interpretable) and can transparently explain seen data as well as predict unseen data (explanatory). Linguistics has become increasingly irrelevant to current high capacity LM design for the goals of improving performance on various NLP tasks such as machine translation, text summary, or information retrieval (Naseem et al. 2021; Lin et al. 2021); in fact, studies suggest that LM performance improves with increase in model scaling and available data (Kaplan et al. 2020; Rae et al. 2022).

The success of natural language LMs without reliance on linguistics is due to the fact that linguistic data conforms to the distributional semantics hypothesis: the idea that words that share similar contexts (other words in the sentence, usually positioned close-by) will have similar meaning (Firth 1957). Because many natural language orthographies are encoded with built-in symbols that indicate linguistic structure, such as space, punctuation and capitalization, explicit linguistic knowledge is often unnecessary to obtain a generalization over linguistically meaningful units (i.e., a Grammar, see Figure 1), and so the semantic meaning of individual words can be approximated via purely statistical distribution. For a successful adoption of LMs to protein sequences, it is important to ensure that distributional semantics is applicable to proteins, and for that, it is necessary to determine the biological units of meaning in protein sequences (i.e., ‘protein words’). Because protein sequences do not have built-in symbols to indicate structure, a more analytical, ‘linguistic’ approach to finding patterns equivalent to ‘protein words’ is needed.

Furthermore, high-performing protein LMs without additional linguistically informed guidance do not guarantee the extraction of interpretable, biologically relevant sequence-function rules (Figure 1). Multiple analyses have benchmarked protein LM performance on questions related to structure and function (Bepler and Berger 2021; Ofer, Brandes, and Linial 2021; Vig et al. 2021; Villegas-Morcillo et al. 2021; Unsal et al. 2022; Rao et al. 2019; M. Xu et al. 2022) and showed that protein LMs can perform remarkably well on these questions. However, current protein LMs by design are opaque black-box models that are unfit to directly provide a protein grammar, i.e., interpretable, rule-based characterizations of sequence-function relationships. An example of such rule-based characterizations would be a detailed list of sequence patterns (e.g., a list of [interdependent] [gapped] k-grams) that are predictive of various biological functions (A. J. Brown et al. 2019). The lack of known sequence-function rules is a current bottleneck in rational biotherapeutic drug design (Akbar et al. 2022). To be truly useful for biological research, the LM must have verifiably learned rules that reflect biological reality.

Lastly, linguistics-inspired domain knowledge incorporation can aid with other particular challenges that stem from the differences between protein and linguistic sequence data (Figure 1). Since currently available protein sequence data undersamples the potential sequence space and since there is no comprehensive knowledge of sequence-function mapping rules, there is no guarantee that the available data contains all relevant domain information, especially for more specialized language models (e.g., 10^9 publicly available immune receptor sequences (Tobias H. Olsen, Boyles, and Deane 2022) vs. $>10^{14}$ possible sequences (Greiff et al. 2017; Elhanati et al. 2015)). In comparison, natural language corpora are more easily verifiable for whether they contain data illustrating all relevant linguistic rules for well-studied languages, due to pre-existing linguistic knowledge. In NLP, models of under-resourced languages especially benefit from more rule-based linguistic priors compared to languages that have an abundance of data available (Kutuzov and Kuzmenko 2019; Y. Pan et al. 2020; Schwartz et al. 2020). Also, protein sequences do not have a larger context from which to infer their meaning (i.e., their function), as protein sequences are essentially unordered strings in an organism. In contrast, in natural language corpora, a sentence is found in a larger context of other sentences, which can aid significantly in inferring the meaning of the sentence. In the absence of larger context, protein sequences have an even higher need for domain-based structural analysis of a sequence in isolation in order to determine its overall function (Jumper et al. 2021). This lack of larger ‘textual’ context for protein sequences also leads to inherently more limited and smaller data compared to linguistic corpora.

In this Perspective, we examine multiple aspects of the deep LM pipeline, which typically consists of three main parts once the appropriate training data has been selected (Rogers, Kovaleva, and Rumshisky 2021; Mielke et al. 2021; Naseem et al. 2021): (1) pre-processing (e.g., tokenization), which is the first step in both (2) pre-training and (3) fine-tuning (Figure 2). Specifically, we discuss pre-training data selection, tokenization, token embedding, sequence embedding, and model interpretation (Figure 2). For each aspect, we draw from the ways linguistics can influence LM design and state the original linguistic motivation behind their implementation, describe the ways current protein LMs have fallen short of considering these motivations, and suggest alternative choices for improving LMs into more appropriate models for protein sequences. Our suggestions are meant to point at future exploratory research directions to address current LM-research-related challenges.

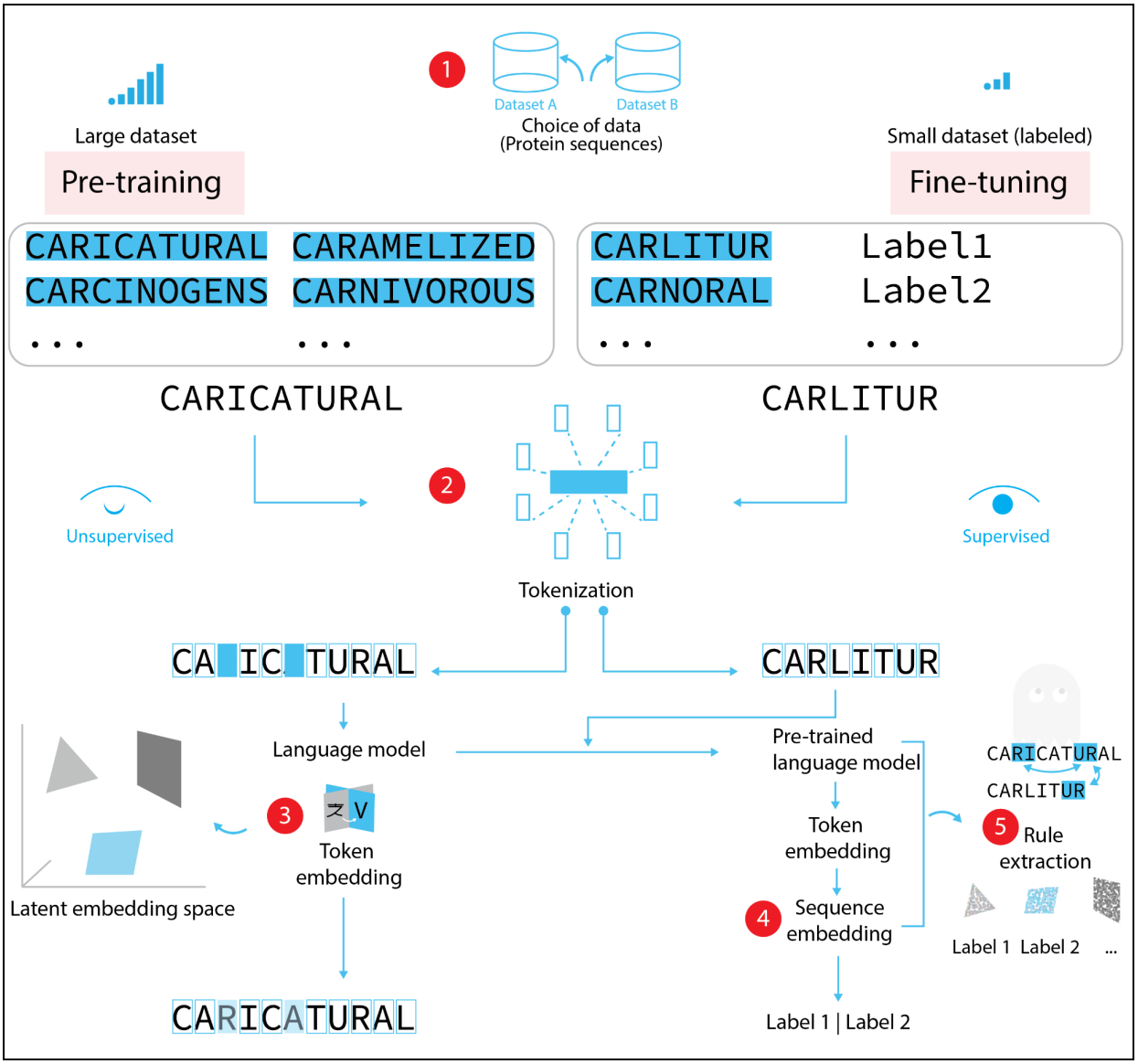

Figure 2 | **Overview of the widely used deep LM pipeline on a protein sequence example**. A ML architecture is pre-trained in a self-supervised manner (Pre-training), independently of the task of interest on large sequencing data (1). Subsequently, the pre-trained model with added layers is trained to perform the task of interest, e.g., classification (Fine-tuning). Fine-tuning can involve tasks that require supervision and hence labeled (e.g., protein function, disease, clinical outcomes) and smaller datasets. Both steps require Tokenization (2) that segments sequences into discrete elements, usually single AAs, due to the lack of task-informed or biologically meaningful tokens. Pre-training assigns a latent embedding to the tokens (3) that represents their contextual usage in the language. The token embedding is leveraged during Fine-tuning, and the sequence embedding is calculated (4) if the fine-tuning task is a form of sequence-based prediction. Interpretation of fine-tuned models, which so far remains in its initial stages, would enable sequence-function rule discovery, such as function-associated long-range sequence dependencies (5).

2 Task-specific pre-training data selection can improve model performance and interpretability

Pre-training is a process in which an LM learns the statistical distribution of a large corpus of data, most typically by identifying missing tokens in a text. Pre-training thus generates a probabilistic model of protein sequence data without explicit supervision, which then can be leveraged for various structural and functional prediction tasks. The probabilistically distributed set of sequences that the LM aims to model is

called the *language*. However, because it is impossible to exhaustively list all sequences of an unboundedly large language, only a representative sample can be given to the model in the form of the pre-training dataset. Data points in the pre-training dataset thus define the language that LMs model: for example, BERT, an LM pre-trained on English Wikipedia and an English book corpus would be a general model of English (Devlin et al. 2019), while BioBERT, which is pre-trained on biomedical texts only (Lee et al. 2020) is a model of biomedical English. Note that in this sense, "language" does not necessarily align with the conventional definition of a natural language, such as Norwegian, Indonesian, or Swahili.

A major challenge for protein LMs is determining a rigorous definition of the *protein language* to be modeled as well as the selection of a pre-training dataset that not only reflects this language but can also be useful in downstream tasks of interest. While pre-training datasets can be unlabeled due to the self-supervised nature of the pre-training task, it is imperative that they contain information that is specifically useful for more data-limited downstream supervised tasks that require labeled data (Devlin et al. 2019) and also suitable for learning extractable rules that reflect genuine domain knowledge. Due to the non-uniform distribution of types of information in datasets, it is difficult to establish a priori how large such datasets should be exactly to be informative. For reference, English language BERT was trained on a corpus of 3×10^9 words (Devlin et al. 2019) and GPT-3 was trained on 4×10^{11} tokens (T. B. Brown et al. 2020). In so far as there is considerable number of studies on the effects and limitations of pre-training data choice on natural language LM behavior (Qiu et al. 2020; Bender et al. 2021; Doddapaneni et al. 2021; Shin et al. 2022; Bender and Koller 2020), a similar investigation into protein LMs is lacking.

Natural language LMs pre-trained on large unannotated linguistic corpora can be applied to various linguistic tasks that may even involve data that is fairly different from the pre-training data due to distributional semantics. Distributional semantics assumes a strong connection between the distributional properties of a token and its semantic meaning (Firth 1957). As long as tokens retain the same semantic meaning in the same contexts across different texts and across different tasks, the tokens should be able to leverage this inferred semantic meaning in fine-tuning datasets that share distributional similarities with the pre-training datasets (Qiu et al. 2020). For example, a pre-trained LM can learn to assign different vector embeddings (i.e., different 'semantic meanings') to the word 'bank' when it follows 'river' (to mean the land alongside a river) compared to its higher frequency meaning of monetary establishment; these learned embeddings can be then transferred and aid fine-tuning for other downstream tasks. Furthermore, distributional semantics allows the sharing of information between tokens with similar meaning: if an LM has learned during pre-training that 'bank' has a similar meaning to 'broker', it can transfer the knowledge it learns about 'broker' during fine-tuning training to 'bank' during fine-tuning testing, even if there is no occurrence of 'bank' in the fine-tuning training dataset. It remains an open question whether distributional semantics is a reasonable assumption for protein sequences when it comes to the functional meaning of protein tokens; that is, whether a given protein token retains the same functional meaning when transferred into a new sequence as long as it is in a similar context.

Even in NLP, defining the modeled language is nontrivial and it must align with the final downstream task goals. For example, several domain specific natural language LMs have been developed to capture domain-specific token meaning without interference from more general English, such as SciBERT for scientific texts (Beltagy, Lo, and Cohan 2019) and BioBERT for biomedical texts (Lee et al. 2020). An

analogous protein LM would be one that is specific to a particular type of protein, such as the antibody-specific LMs AntiBERTa (Leem et al. 2021), AntiBERTy (Ruffolo, Gray, and Sulam 2021), and AbLang (Tobias Hegelund Olsen, Moal, and Deane 2022).

An opposite strategy is to use the most general and largest possible dataset for pre-training, while answering questions that only target a subset of those sequences. For example, LMs pre-trained on multiple natural languages have been leveraged for monolingual tasks, cross-linguistic tasks such as machine translation, and zero-shot learning, which in the multilingual LM context is defined as fine-tuning the LM on labeled data in a source language, but then test on a different target language (Doddapaneni et al. 2021). The pre-training data for multilingual LMs usually include unlabeled data from both the source and target languages (Doddapaneni et al. 2021).

Multilingual LMs are of particular interest for protein LMs, because they open up the possibility of leveraging an LM pre-trained on all available millions of protein sequences to perform tasks relevant to only a small subset of those sequences (e.g., antibody receptors) that by themselves would not be large enough for learning embeddings in a self-supervised manner. Furthermore, multilingual LMs could potentially be applied to zero-shot or few-shot learning problems in biology too, where source and target datasets differ during the fine-tuning phase. However, studies in NLP demonstrated that the performance of multilingual LMs remains limited compared to monolingual LMs; their performance correlates with the size of relevant language training data either in the pre-training or fine-tuning phase (Conneau et al. 2020; Agerri et al. 2020; C.-L. Liu et al. 2020; Lauscher et al. 2020), and with the similarity between the source language and the target language in the case of zero-shot learning (de Vries, Wieling, and Nissim 2022).

In the most extreme zero-shot learning cases, LMs have shown capability for knowledge transfer between drastically different types of data (Kao and Lee 2021; Krishna, Bigham, and Lipton 2021). Kao and Lee pre-train their model on linguistic data, then fine-tune on proteins in one instance, and music on another. Since it is highly implausible that for example LMs pre-trained on natural language would learn patterns that reflect genuine and scientifically useful domain knowledge in biology or music (Kao and Lee 2021), LM performance alone cannot be a reliable criterion for pre-training data selection, especially in these cases, and a priori study of the problem and sequence distribution are necessary.

Based on current results in NLP, it is likely that LMs that are pre-trained on all available protein sequences will be most appropriate for downstream tasks that predict general features of proteins, such as secondary structure, amino acid contact in the structure, and stability (Rao et al. 2019). To answer questions that are specific to only certain types of proteins, such as antibody affinity maturation or epitope prediction, which are only applicable to antibody sequences, specialized antibody LMs are likely to perform better (Leem et al. 2021; Tobias H. Olsen, Boyles, and Deane 2022; Ruffolo, Gray, and Sulam 2021; Ruffolo, Sulam, and Gray 2022; Shuai, Ruffolo, and Gray 2021; Ostrovsky-Berman et al. 2021): for example, AntiBERTa outperforms ProtBERT, a general protein model (Elnaggar et al. 2021) on a number of antibody-specific questions (Leem et al. 2021).

In order to choose the appropriate pre-training data that can contribute to true scientific insights, there thus needs to be careful rational consideration whether it contains information transferrable to the downstream task, more empirical study to determine the viability of different types of pre-training data

for various fine-tuning tasks compared to randomly generated nonsense data, and more available large datasets for specialized types of proteins. The last two points may be addressed with computational simulations (Robert et al. 2021; Marcou, Mora, and Walczak 2018; Weber et al. 2020; Morris, White, and Crowther 2019), which can generate arbitrarily large datasets with a priori defined rules to test different approaches.

3 Linguistically-guided tokenization motivates a search for meaningful biological units in protein sequences

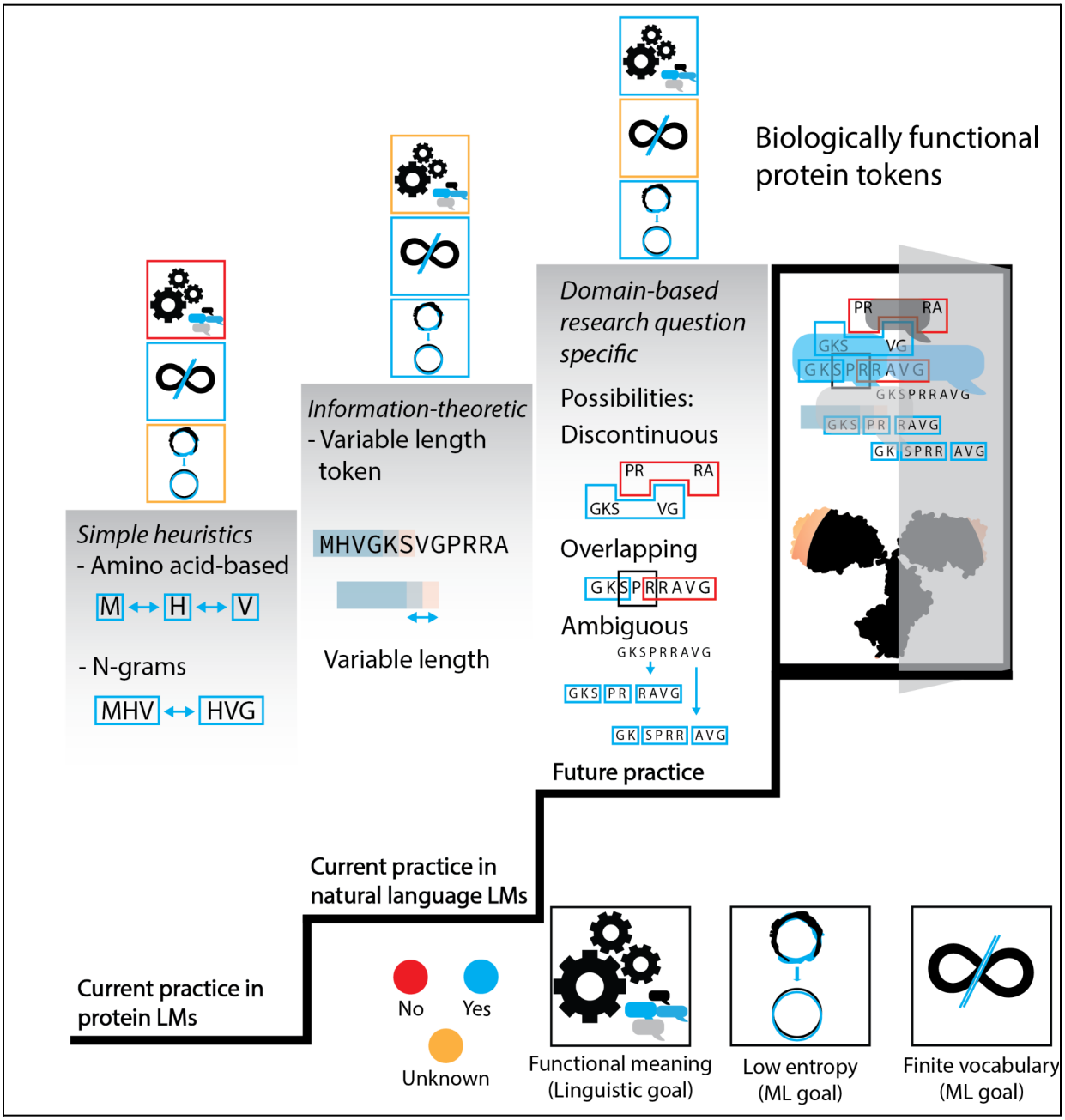

Figure 3 | **Advancing protein sequence tokenization from currently popular simple heuristics to complex methods that**

would generate biologically functional protein tokens akin to linguistically sound tokens in natural language. Tokenization methods must balance three distinct goals. Linguistically sound tokens should atomically map to well-defined, abstract functional meaning. Technical constraints in ML necessitate that the generated set of possible tokens is finite and small in number (finite vocabulary), and that tokenization yields a LM with low entropy for a fixed vocabulary size (low entropy). Current practice in protein LMs is to use simple heuristics that result in tokens based on single amino acids or n-grams. While such simple heuristics yield a finite and small vocabulary, they do not map to functional meaning and it is unclear how low the generated LM entropy is. Information-theoretic tokenization methods are one step more complex, and are currently popular in natural language LMs. They also result in a finite, though larger vocabulary than simple heuristics do, and they generate low entropy LMs, but it is still unclear whether they would map to functional meaning in proteins. Finally, the most complex method is domain-based tokenization that is specific to a research question. The tokens yielded with this method map to well-defined functional meaning, but might potentially result in an arbitrarily large, practically non-finite vocabulary. They should still generate low entropy LMs. It is yet to be seen how domain-based tokens manifest, but they might be discontinuous, overlapping, and they might be ambiguous, meaning that there might be multiple possible segmentations for a given sequence. Domain-based tokenization is closest to biologically sound protein tokens akin to linguistically defined tokens in natural language.

Just as understanding a language requires knowledge of its basic vocabulary, processing sequence data requires identification of its discrete information units. Breaking down unstructured sequence data into its basic building blocks or tokens, whether through domain knowledge or through a tokenization algorithm, is therefore a fundamental step in an LM pipeline (Figure 2). Tokenization in NLP serves computational goals and ideally it can also fulfill linguistic goals. From a computational perspective, tokenization is useful because it reduces data sparsity: it enables the representation of unseen sequences as a combination of already seen tokens drawn from a finite vocabulary, with the trade-off that it results in longer encoding for each sequence (Mielke et al. 2021). Thus to fulfill the computational goals, tokenization needs to create a relatively small, finite but exhaustive vocabulary that avoids out-of-vocabulary tokens. Furthermore, tokenization is preferably unsupervised and results in an LM with low information entropy distribution for a given vocabulary size, which ensures lower perplexity (the ability for a model to predict a sample) (P. F. Brown et al. 1992; J. Xu et al. 2021) (Figure 3). From a linguistic point of view, the main criteria is that tokens should correspond to morphemes: atomic units carrying abstract *meaning* that cannot be inferred from the characters alone. Such meaningful tokens have been traditionally derived through careful manual linguistic analysis, and necessitate an ever-expanding open vocabulary as natural languages constantly admit new words.

In current NLP practice, tokenization methods are a matter of trade-offs between ML requirements and linguistic criteria, and thus can range from simple heuristics such as space-delimited tokenization to more information-theoretic, data-driven methods such as Byte-Pair-Encoding (BPE) (Gage 1994) and its varieties, to hand-crafted tokens derived from linguistic analysis (Mielke et al. 2021). Developing tokenization algorithms is an active field of research within NLP, and the performance of different tokenization methods has been extensively studied (Mielke et al. 2021; Pinter 2021). Results show that there is no single optimal tokenization strategy, as best practices depend on the intended task, the language, and available data (Hofmann, Pierrehumbert, and Schütze 2021; Kutuzov and Kuzmenko 2019; Pinter 2021; Mielke et al. 2021; J. Xu et al. 2021).

For protein LMs, similarly extensive investigations of diverse tokenization strategies and their effects on LM performance are lacking, because there is no biologically informed tokenization equivalent to linguistically informed tokenization in NLP, and the biological rules for assembling protein building blocks mapped to semantic meaning, defined here as specific, abstract functions, remain unknown below the protein domain level. To build more robust protein LMs, we argue for more extensive benchmarking

studies of different tokenization methods and for more effort directed into defining ground truth, biologically informed sequence tokens that reflect discrete protein functional groups (Figure 3). Given the limited size of protein sequence data, especially for certain types of proteins, we believe that defining hand-crafted, domain-informed tokens is necessary for building an effective LM, similarly to how linguistically guided tokenization leads to better results for under-resourced languages (Kutuzov and Kuzmenko 2019; S. J. Pan and Yang 2010; Schwartz et al. 2020).

Currently the most popular protein tokenization methods remain at the simplest level, as they are either amino acid-based (Alley et al. 2019; Heinzinger et al. 2019; Elnaggar et al. 2021; Littmann et al. 2021; Madani et al. 2021; Villegas-Morcillo et al. 2021; Brandes et al. 2022) or k-gram-based (typically 3-grams) (Asgari and Mofrad 2015; Ostrovsky-Berman et al. 2021; Yang et al. 2018) (Figure 3). The estimated entropy rate with amino acid-based tokenization for a very small sample of protein sequences is 2.4-2.6 (Strait and Dewey 1996), while the estimated entropy rate for English based on character-based tokenization is 0.6-1.75 (Shannon 1951; P. F. Brown et al. 1992). Information theoretic measures such as entropy rate should be consistently reported for any new protein LMs so that they can be compared to natural language LMs, but these measures are often missing. Even if the performance of protein LMs with current tokenization methods is high, significant divergence in entropy rate between protein and natural language LMs indicates a possibility for better alternative tokenization methods.

When it comes to validating the meaning of the tokens gained from simple tokenization, both amino acid and k-gram token embeddings cluster along physicochemical properties, which are individual amino acid properties (Alley et al. 2019; Asgari and Mofrad 2015; Ostrovsky-Berman et al. 2021). Thus, based on the published results, for the purposes of finding biological tokens with more abstract function, neither amino acid nor k-gram tokenizations are satisfactory. There needs to be further research to determine whether these simple tokens cluster along more abstract biological behavior, such as more global properties of the protein itself, but it is likely that only more complex tokens can reflect complex biological functions.

Instead of using simple amino acid or n-gram tokenization, a more sophisticated method is to extract tokens of variable size using data-driven, information-theoretic algorithms; this type of tokenization method is the most popular in current NLP applications (Mielke et al. 2021). Only a few studies applied these types of algorithms to protein sequences (Devi, Tendulkar, and Chakraborti 2017; Asgari, McHardy, and Mofrad 2019; Y. Wang et al. 2019; Brandes et al. 2022; Szymborski and Emad 2022), and their effectiveness in protein LMs remain mixed compared to simpler tokenization methodologies (Y. Wang et al. 2019; Brandes et al. 2022; Szymborski and Emad 2022). Given that in NLP, new information-theoretic tokenization algorithms are actively developed (Gage 1994; Kudo and Richardson 2018; Mielke et al. 2021; Pinter 2021), there are still numerous unexplored options to implement and test in protein LMs.

Furthermore, in previous studies biologically sound tokens were either validated in terms of their performance on a downstream task (Devi, Tendulkar, and Chakraborti 2017) or in terms of similarity to experimentally verified motifs (Asgari, McHardy, and Mofrad 2019). Even so, the specific functional meaning of the generated tokens remains undefined, and thus these tokens do not reach the equivalent standard for linguistically sound natural language tokens, which should have clear mapping to specific meaning. It is also known from studies in NLP that information-theoretic algorithms, though widely used due to their efficiency, are not reliable for finding linguistically sound tokens (Hofmann, Pierrehumbert,

and Schütze 2021; Hofmann, Schütze, and Pierrehumbert 2022). In the absence of well-defined, biologically meaningful protein tokens, a truly informative testing of tokenization algorithms remains impossible.

Finally, an unexplored possibility is building a rule-based tokenizer grounded in domain knowledge for a specific downstream task of interest. Defining biologically sound protein tokens with the criteria that they map to well-defined biological function remains a challenging task, and previous research often resorted to pragmatically defined quantitative measurements, as seen above. In contrast, hand-crafted protein tokens, where the only criteria is that they map to some biological function relevant to a downstream task are less straightforward and more labor-intensive to generate; it necessitates domain-based expertise to hypothesize how such tokens would manifest. Based on known properties of protein structure and function, meaningful protein tokens might be discontinuous, overlapping, and each sequence might need to map to several different tokenization possibilities (Figure 3), unlike natural language tokens in most cases. For example, in the case of antibody sequences, one might define tokens for a given antibody receptor to be multiple different possible paratopes, which are a set of non-continuous amino acids that interact with the recognized antigen (Akbar, Robert, Pavlović, et al. 2021; Robert et al. 2021), and map different antigen specificity to each different paratope tokenization.

If protein tokens are defined to be non-continuous, overlapping and possibly ambiguous, it will also be necessary to employ alternative LM technology to appropriately process them. One naive solution would be to shift from non-continuous tokens to smaller continuous subtokens with learned long-distance dependency between them, but this strategy misses the opportunity of actually identifying meaningful tokens, and conceptually mixes tokenization with long-distance dependency rules at the expense of higher interpretability. Another possibility, employed in NLP, is to re-order the non-continuous tokens (Welleck et al. 2019; Stern et al. 2019) so they become continuous.

In any case, given the amount of expert knowledge necessary to define biologically meaningful tokens, such hand-crafted, function-based tokenization might only be valuable in practice if it is possible to develop an unsupervised algorithm that can tokenize novel sequences. A possibility is to train a tokenizer based on a large number of defined tokens in protein simulations; for example, simulated antibodies in high resolution can give information about interacting and non-interacting segments (Robert et al. 2021), which then can be used to train a tokenizer. Further evaluation would be needed to compare the results from simulated data to the desired ground truth in real world data.

Altogether, there remain many open questions regarding protein tokenization, most importantly, the definition of biologically sound protein tokens akin to linguistically sound tokens for language to serve as ground truth, and comparing the performance of various tokenization methodologies in terms of how well they approximate the ground truth as well as how they might influence protein LM performance on downstream tasks. The fact that proteins contain units that compositionally determine their function (at least at the scale of protein domains (Gimona 2006)) similarly to how linguistic tokens compositionally map to sentence meaning, suggests that analytic, linguistic tokenization methods may be transferable to protein tokenization, given more robust data and investigations.

4 Linguistic considerations require token embeddings that capture protein function for interpretability

For tokens to be used in a deep learning environment, they ideally are numerically represented in a way that reflects the similarities and differences between these units. These representations of tokens, determined by their context of use, are called embeddings, and are represented as vectors in a multidimensional vector space. Token embeddings are initially calculated during pre-training, and then are further refined during fine-tuning. Pre-trained embeddings can be extracted from the hidden layers of the pre-trained LM and then leveraged as input for downstream tasks that use much smaller datasets (Figure 2) (Devlin et al. 2019). The linguistic function of a token embedding is to reflect the relevant linguistic role of the token in the text. In the case of protein tokens, these roles are equivalent to their biological functions or other contextual aspects that cannot be easily captured conceptually due to their non-linearity and complexity. Being able to capture the functional meaning of protein tokens would also improve the interpretability of the model.

Following distributional semantics, by pre-training LMs to predict tokens based on their context in large, unannotated linguistic data, the latent vector representation associated with a given token reflects that token's context and presumably correlates with its lexical meaning. The information captured by the embedding vector is typically validated by showing that synonymous tokens cluster close to each other in the embedding space visualization and have high cosine similarity, or with arithmetic calculations, such as `king-man+woman≈queen` (Mikolov, Yih, and Zweig 2013). Note that this type of validation requires a priori knowledge about token meaning. A weakness of this validation method is also its staticness: there are several ways for two words to be related (e.g., king-queen and king-chief can both be closely related pairs), and so similarity is always along only certain attributes (Schluter 2018). It is thus important to have a well-defined similarity attribute in mind before validating token embeddings. The success of the fine-tuning task can also indicate, albeit only indirectly, the usefulness of the vector embeddings.

For protein LMs, it is uncertain whether distributional semantics is plausible as the working principle behind mapping protein token to functional meaning, that is, whether the biological function of protein tokens is actually encoded by their contexts. To answer this question, there first needs to be a clear definition of ground truth, task-specific protein tokens with known functional meaning. Furthermore, it is likely that different downstream tasks would require different token embedding methods that are altered from current mainstream NLP practice. Therefore, it is necessary to have clear theoretical reasoning about the definition of protein token meaning before designing the appropriate LM pre-training task.

Most current protein LMs directly borrow standard NLP pre-training tasks for token embedding, without further justification or reasoning (Alley et al. 2019; Asgari and Mofrad 2015; Heinzinger et al. 2019; Leem et al. 2021; Ostrovsky-Berman et al. 2021; Elnaggar et al. 2021; B. Hie et al. 2021). Protein LMs have also closely followed the evolution of embedding methods in NLP, moving from non-contextual, rigid token embedding techniques such as word2vec (Mikolov et al. 2013; Asgari and Mofrad 2015; Asgari, McHardy, and Mofrad 2019; Ostrovsky-Berman et al. 2021) where each token has a fixed embedding vector, to contextual token embeddings such as ELMo (Peters et al. 2018; Heinzinger et al. 2019; Littmann et al. 2021; Elnaggar et al. 2021; Villegas-Morcillo et al. 2021) and BERT (Devlin et al. 2019; Elnaggar et al. 2021; Leem et al. 2021; Rao et al. 2019), where the token embeddings depend on context. All token embedding tasks involve predicting a token based on its context, where context can be

either a window of k-grams (as in word2vec), all previous tokens (as in LSTMs), or everything else in the sequence (as in transformer-based models). The tokens, which often are amino acids, are usually represented with one-hot encoding, and the context is defined as the rest of the protein sequence itself. In natural language, non-contextual token embeddings are unsatisfactory due to polysemy (the fact that words can have multiple meaning, depending on context), and the same is likely to hold for protein sequences, where biological function is typically encoded in several non-linear, long-distance dependencies (Akbar, Robert, Pavlović, et al. 2021).

As an alternative to using the rest of the protein sequence as the context that defines token meaning, context could be defined with the downstream task in mind. For example, in the case of antibody binding prediction, it could be reasonable to define the meaning of antibody tokens as the antigens they interact with by encoding the input as antibody-antigen pairs to begin with, before transferring the latent embeddings to novel antibody sequences. In existing protein LM studies, an example for an alternative definition of context is found in ProteinBERT (Brandes et al. 2022), which was pre-trained on protein sequences encoded together with their Gene ontology (GO) annotation, which is an annotation of the protein sequence function. The result is that the embeddings of single amino acids contained information from both the sequence and the GO annotation of the sequence. It remains to be determined how information about GO annotations contributed to better performance on the final tasks, and whether there is a bias toward better performance on only certain types of downstream tasks.

Another possible change is to pre-train the protein LM on protein-specific tasks, in addition to the self-supervised language modeling task, similarly to training natural language LMs to learn syntactic knowledge in addition to standard language modeling tasks (Dyer et al. 2016; Eriguchi, Tsuruoka, and Cho 2017). One popular self-supervised pre-training task is structural information prediction (Bepler and Berger 2021; Chen et al. 2022). However, as with ProteinBERT, there is a lack of deeper discussion of the information captured by LMs and of the downstream tasks the LMs would be biased to perform well on due to the pre-training task. For antibody sequences for example, different folding structures might bind different antigens (Guest et al. 2021), so pre-training on structural information might not be appropriate, as it would encode rigid structural information in the token embeddings. This does not negate the fact that structural information can be useful for antibody function prediction during the fine-tuning phase (Akbar, Robert, Pavlović, et al. 2021; Akbar, Robert, Weber, et al. 2021), provided that structural information is not rigidly determined during the pre-training phase. Furthermore, pre-training tasks that rely on information beyond just the sequence itself cannot be used if only unlabeled sequence data is available.

Token embeddings extracted from protein LMs are usually validated through clustering plots (Alley et al. 2019; Asgari and Mofrad 2015; Ostrovsky-Berman et al. 2021; Heinzinger et al. 2019) and through performance on downstream tasks (as seen for example in ProteinBERT (Brandes et al. 2022; Bepler and Berger 2021)). Besides the general limitations of clustering discussed above (i.e., the definition of similarity), another difficulty particular to protein tokens is that there is a lack of knowledge about the abstract functional meaning of protein tokens; what is known only is the physicochemical properties of the amino acids. Accordingly, in all studies that performed a clustering analysis, whether protein tokens were amino acids or 3-grams, similarity was defined in terms of physicochemical properties and tokens were indeed shown to cluster along those properties. None of the studies show whether amino acid and 3-gram tokens also encode more abstract functional meaning, likely because more abstract functional

meaning can only reasonably be expected to be encoded in larger tokens that are gained from domain-based tokenization.

For sequence-based prediction tasks, token embeddings are the sole source for deriving sequence embeddings, since protein sequences cannot rely on a larger context of ‘protein text’ the same way linguistic sentences do on linguistic texts. Currently the most popular method for calculating protein sequence embedding is through average pooling (i.e., they take the average of the token embeddings) (Alley et al. 2019; Elnaggar et al. 2021; Rao et al. 2019; Heinzinger et al. 2019; Detlefsen, Hauberg, and Boomsma 2022). In contrast, linguistics provides a principled, rule-driven method for deriving sentence meaning from structure called compositional semantics (Montague 1970), and there are multiple alternative, structure-sensitive sequence embedding techniques in NLP as well (e.g., structurally informed ones (McCoy, Frank, and Linzen 2020; Tai, Socher, and Manning 2015)). For protein LMs, it remains an understudied question whether these other strategies could improve performance and contribute to better overall interpretability (Detlefsen, Hauberg, and Boomsma 2022).

In summary, protein LMs lack an extensive investigation into the information that their learned token embeddings contain, as all evaluation remains superficial with plotting physicochemical properties and indirect with benchmarking on downstream tasks. We argue that together with domain-based tokenization and token embedding task definition that is directly relevant for the research question at hand, token embeddings could capture more abstract biological functions that go beyond physicochemical properties. Such token embeddings would also significantly improve the interpretability of the protein LMs built with them.

5 Interpretability methods applied to protein language models can aid biological rule discovery

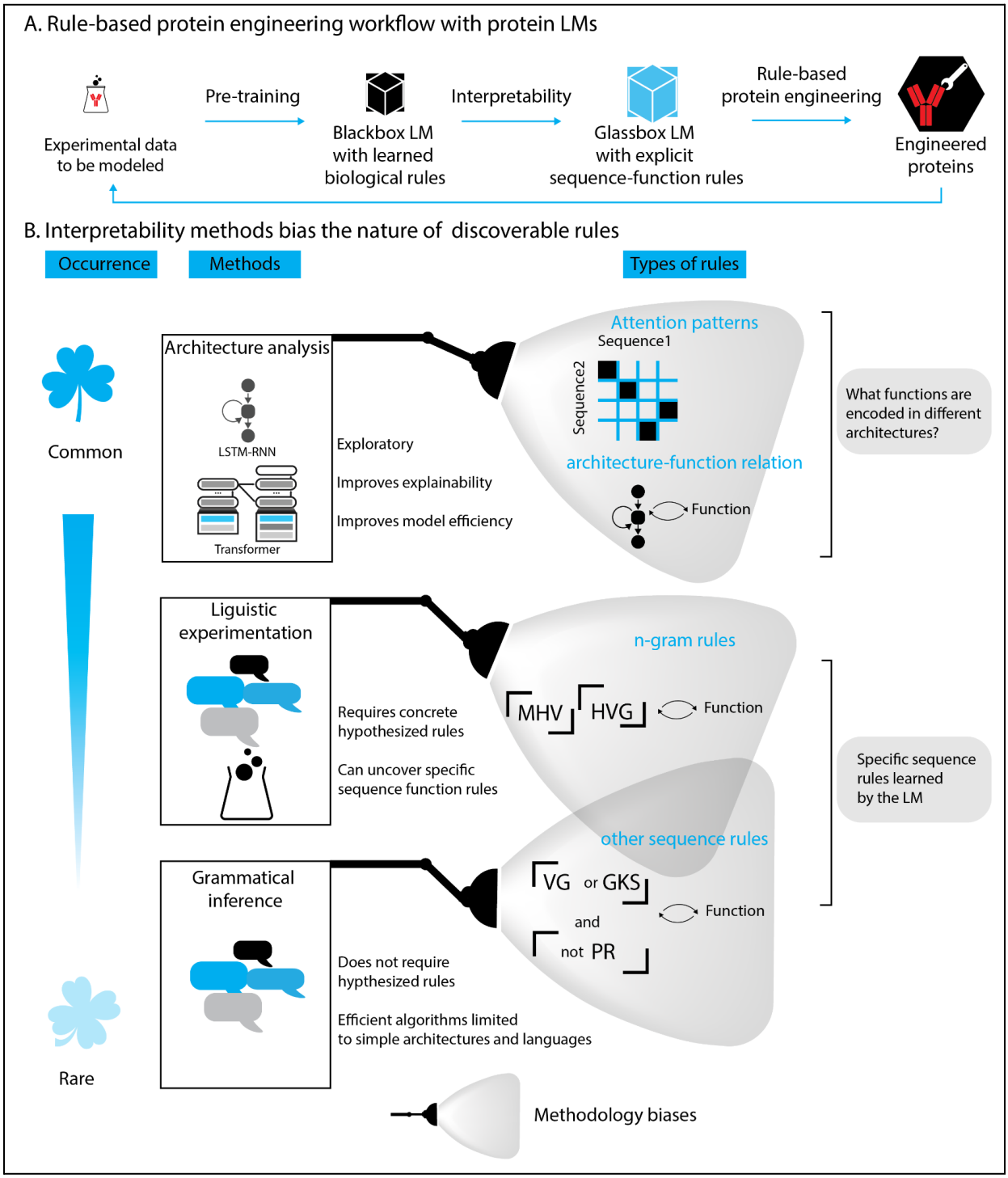

Figure 4 | **Interpretability methods for protein LMs. (A) Rule-based protein engineering workflow with protein LMs.** Blackbox protein LMs pre-trained to model the appropriate experimental data are expected to learn biological sequence-function rules present in the pre-training data. Rule- and pattern-extraction methods can increase the explainability and interpretability of the LM, resulting in a glass box LM with interpretable sequence-function rules. Sequence-function rules then can be leveraged for protein engineering. Engineered proteins that have been experimentally validated can potentially be added to the training data for improving protein LMs. **(B) Interpretability methods bias the nature of discoverable rules.** Different interpretability

methods highlight different types of information about the architecture and the sequences. Architecture analysis (1) is the most commonly used method with current protein LMs for explaining blackbox LMs (Rogers, Kovaleva, and Rumshisky 2021; Tenney, Das, and Pavlick 2019; Vig et al. 2021; Ruffolo, Sulam, and Gray 2022; Leem et al. 2021), but it can mostly only yield a better understanding of the architecture itself, and not necessary specific sequence-function rules. A better understanding of the architecture is nevertheless useful for improving the explainability and efficiency of the model. Linguistics-inspired experimentation (2) (Linzen, Dupoux, and Goldberg 2016; Goldberg 2019; Linzen 2018; Ettinger 2020; Warstadt et al. 2020; Hu et al. 2020), aims to find correlation between already known rules and the LM, by testing it with hand-crafted sequences that either follow or violate a hypothesized rule. Linguistics-inspired experimentation requires a concrete hypothesized rule (often based on pre-existing analysis), which might be unfeasible for protein sequences due to the vast space of all possible rules. Studies aimed at examining the type of rules various deep neural network architectures are capable of learning (Bhattamishra, Ahuja, and Goyal 2020; Clark, Tafjord, and Richardson 2020) can provide a way to limit the search space for rules, though the limits might still not be sufficiently restrictive for exhaustive rule extraction. Grammar inference methods for deep neural networks (3), which can extract rules of a predefined power (Weiss, Goldberg, and Yahav 2020; Eyraud and Ayache 2021; Q. Wang et al. 2018) do not require an a priori hypothesis for a concrete rule, but efficient algorithms as of now are limited to simpler architectures (e.g., RNNs) and are not yet practically suitable for more large-scale rule extraction.

The research goal of theoretical linguistics is to describe natural language with interpretable and explanatory rules, i.e., a grammar. To this end, linguistics applies iterative and analytical methodologies that are sustained by the relative ease of principled and fast data collection from speakers. In contrast, natural language LMs, which were developed in the field of NLP for primarily language engineering applications, prioritize accurate modeling of language data over discovering facts about language. Originally, symbolic language models in NLP were built based on the rules obtained from basic linguistics research, but with the rise of neural network models, high-performing, unsupervised, statistical LMs now vastly outnumber symbolic language models (Church 2011; Church and Liberman 2021).

More recently, the need for transforming black-box models into transparent, interpretable glass-box models has resulted in an increasing effort for obtaining better interpretability from deep natural language LMs (Rogers, Kovaleva, and Rumshisky 2021). For example, BERTology, a research program dedicated to interpreting the BERT language model, has developed out of this need. Improving model interpretability is crucial to better understand what exactly these models learn, and on a more practical level, to be able to pinpoint the causes for their failures (accountability). Among others, linguistics-inspired methods to probe LMs have been popular in NLP research (Linzen 2018; Ettinger 2020; Hu et al. 2020). Similar efforts, however, have not been widely adopted for protein LMs (Vig et al. 2021), and we argue that incorporation of interpretability and explainability concerns should be an essential part of protein LM design from the start.

Inferring rules from protein LMs with interpretability methods is an integral step in the rational protein engineering pipeline (Akbar et al. 2022) (Figure 4A). A protein LM that has incorporated the design considerations discussed so far should have ideally learned relevant sequence-function rules. However, so far, rules remain hidden within the black-box model. Various interpretability and rule-extraction methodologies are then needed to find possible sequence-function rules and use them to inform rational protein design and novel protein synthesis (Akbar et al. 2022). It is crucial that the inferred rules are used as guidance rather than true answers about biology, and that they are experimentally validated. The novel proteins designed based on the inferred rules can in turn serve as additional data for further LM training after experimental validation. Rigorous interpretability-focused examination can also help evaluate whether well-performing models have learned truly meaningful representations and sequence-function mappings (McCoy, Pavlick, and Linzen 2019; McCoy et al. 2021; Niven and Kao 2019).

We distinguish three types of interpretability methods: architecture analysis, linguistics-inspired experimentation, and grammatical inference (Figure 4B). The choice of method biases the type of information that can be learned about the LM and the modeled sequences so it is crucial to be aware of the limitations inherent in the chosen method. Broadly, there are two types of information that can be gained: the localization of specific types of knowledge in the architecture, and the specific sequence-function rules that the model has successfully learned (Figure 4B, rightmost column).

Architecture analysis (e.g., studying specific layers in the architecture, probing the pre-trained embeddings, interpreting attention pattern heatmaps and saliency maps) can yield information about where and how the model architecture stores various types of knowledge about the sequence, and it is by far the most popular method of analyzing LM knowledge. Understanding the localization and method of knowledge storage in the architecture can improve the explainability and efficiency of the model. For example, in natural language BERT, it was found that lower layers possess information about linear word order, while the middle layers encode more hierarchical, syntactic information (Rogers, Kovaleva, and Rumshisky 2021; Tenney, Das, and Pavlick 2019). For protein LMs, attention patterns were shown to correlate with amino acid contact in the protein structure (Vig et al. 2021; Ruffolo, Sulam, and Gray 2022; Leem et al. 2021). At the same time, the usefulness of this knowledge remains limited for rational protein design, as it does not contribute to basic biological knowledge about the protein sequences. Furthermore, these types of explainability methods are often reliant on subjective human interpretation (Rudin 2019; Adebayo et al. 2021) and are demonstrated on a very small number of examples (Ruffolo, Gray, and Sulam 2021; Leem et al. 2021). As a result, the accuracy of these methods might be compromised. It is thus crucial to employ them with caution, and simulated ground truth data can help with a controlled benchmarking.

The two other methods, linguistics-inspired experimentation and grammatical inference can shed light on generalizable, well-defined sequence-function rules that the model has learned. Gaining such generalizable sequence-function rules is the most useful information for rational protein design, as they have the potential for leading to a more complete ‘grammar’ of protein languages. In linguistics-inspired experimentation, researchers probe the knowledge of the model of a hypothesized sequence rule by feeding it constructed sequences that either follow or violate the rule (Linzen, Dupoux, and Goldberg 2016; Goldberg 2019; Linzen 2018; Ettinger 2020; Warstadt et al. 2020; Hu et al. 2020). For this type of experiment, it is typical to construct materials that are similar to those used in psycholinguistics experiments that probe human language processing capabilities. For example, Goldberg (2019) has found that BERT can distinguish near perfectly between sentences such as the grammatical “*The game that the guards hate is bad*” and the ungrammatical “*The game that the guards hate are bad*”, indicating that BERT has learned structure-based subject-verb agreement rules in English, even when there are intervening distractor nouns (Goldberg 2019). This type of linguistics-inspired experimentation has the same challenges as psycholinguistics experimentations in that the design of the experiment must rigorously rule out possible confounding factors. An additional challenge of applying this method to protein LMs is that it requires a priori knowledge of a concrete hypothetical rule to test. Since the goal for protein LMs is to learn new rules, applying this methodology requires a substantial amount of guess work. Even if the hypothesis space of possible rules is limited based on domain knowledge in biology and theoretical analysis of the computational capacity of LM rule knowledge (Bhattamishra, Ahuja, and Goyal

2020; Clark, Tafjord, and Richardson 2020), it might still remain too vast to systematically test with this method.

Finally, there is a long history of developing grammatical inference algorithms with the purpose of extracting grammar (i.e., a set of rules) from a set of strings (Gold 1967; Angluin 1987). For example, Angluin's L* algorithm can learn a finite-state automata that describes a set of strings, if membership and equivalence queries are allowed (Angluin 1987). More recent research has attempted to apply these algorithms to extract grammar from neural networks. Weiss et al. have shown, for example, that the L* algorithm can be used to obtain a finite-state automata from an RNN, where the RNN itself serves as the oracle (Weiss, Goldberg, and Yahav 2020). The advantage of grammatical inference compared to linguistically inspired methods is that it does not require concrete hypothesized rules to use; however, it still requires a priori restriction on the *class* of possible rules, as there is no algorithm that can infer patterns generated by a Turing-complete algorithm. Another disadvantage of grammatical inference algorithms is that as of now, efficient algorithms only exist as proof of concept, for relatively simple models (mainly RNNs) and types of rules, and also cannot handle noisy input data well. Nevertheless, these algorithms could be potentially useful for applying to well-performing protein LMs as the grammatical inference field develops better algorithms (Eyraud and Ayache 2021; Q. Wang et al. 2018).

In summary, well-designed and well-performing protein LMs can only reach their full potential to be useful for rational protein design as transparent glass-box models with thoroughly understood, interpretable rules. We have identified three types of methodology (architecture analysis, linguistic experimentation, and grammatical inference), and each of them biases the type of information that can be gained about the LM and all of them requires a better a priori understanding of the types of biological rules that exist in protein sequences. Therefore, in order to access a full, transparent understanding of a protein LM, there needs to be ongoing probing of the LM that uses a diversity of methodology. An employment of ground truth (simulated) data can further help with examining these interpretability methods (Robert et al. 2021).

6 Conclusions

Similarities between protein and natural language sequences have inspired the use of LMs for protein sequences, which are originally tools for modeling linguistic sequences. Self-supervised protein LMs have a potential for identifying relevant sequence rules that can be further experimentally tested, and thereby contributing to fundamental questions in biological research and accelerating the rational protein therapeutics design. However, current practice in designing and building protein LMs have fallen short of appropriately adapting these models to protein sequences due to a lack of a deeper understanding on how they were originally built for modeling linguistic sequences. In this Perspective, we have highlighted various parts of the LM pipeline (pre-training data, tokenization, token embedding, sequence embedding, and rule extraction), and we have shown how understanding the original linguistic intent underlying each of these steps can inform the building of more appropriate protein LMs that answer specific downstream questions of interest. Protein LMs that have thoughtfully incorporated the considerations discussed at each point are then more likely to have learned the relevant biological rules for the sequences they model, and thus may be better suited for successful rule-extraction that can be used for rational protein design.

Funding

We acknowledge generous support by The Leona M. and Harry B. Helmsley Charitable Trust (#2019PG-T1D011, to VG), UiO World-Leading Research Community (to VG), UiO:LifeScience Convergence Environment Immunolingo (to VG, GKS, and DTTH), EU Horizon 2020 iReceptorplus (#825821) (to VG), a Research Council of Norway FRIPRO project (#300740, to VG), a Research Council of Norway IKTPLUSS project (#311341, to VG and GKS), a Norwegian Cancer Society Grant (#215817, to VG), and Stiftelsen Kristian Gerhard Jebsen (K.G. Jebsen Coeliac Disease Research Centre) (to GKS).

Acknowledgements

We thank Kyunghyun Cho (New York University, New York, NY, USA) and Emily M. Bender (University of Washington, Seattle, WA, USA) for their very helpful comments on the manuscript.

Declaration of interests

V.G. declares advisory board positions in aiNET GmbH, Enpicom B.V, Specifica Inc, Adaptyv Biosystems, EVQLV, and Omniscope. V.G. is a consultant for Roche/Genentech, immunai, and Proteinea.

References

Adebayo, Julius, Michael Muelly, Harold Abelson, and Been Kim. 2021. "Post Hoc Explanations May Be Ineffective for Detecting Unknown Spurious Correlation." In . https://openreview.net/forum?id=xNOVfCCvDpM.

Agerri, Rodrigo, Iñaki San Vicente, Jon Ander Campos, Ander Barrena, Xabier Saralegi, Aitor Soroa, and Eneko Agirre. 2020. "Give Your Text Representation Models Some Love: The Case for Basque." *ArXiv:2004.00033 [Cs]*, April. http://arxiv.org/abs/2004.00033.

Akbar, Rahmad, Habib Bashour, Puneet Rawat, Philippe A. Robert, Eva Smorodina, Tudor-Stefan Cotet, Karine Flem-Karlsen, et al. 2022. "Progress and Challenges for the Machine Learning-Based Design of Fit-for-Purpose Monoclonal Antibodies." *MAbs* 14 (1): 2008790. https://doi.org/10.1080/19420862.2021.2008790.

Akbar, Rahmad, Philippe A. Robert, Milena Pavlović, Jeliazko R. Jeliazkov, Igor Snapkov, Andrei Slabodkin, Cédric R. Weber, et al. 2021. "A Compact Vocabulary of Paratope-Epitope Interactions Enables Predictability of Antibody-Antigen Binding." *Cell Reports* 34 (11): 108856. https://doi.org/10.1016/j.celrep.2021.108856.

Akbar, Rahmad, Philippe A. Robert, Cédric R. Weber, Michael Widrich, Robert Frank, Milena Pavlović, Lonneke Scheffer, et al. 2021. "In Silico Proof of Principle of Machine Learning-Based Antibody Design at Unconstrained Scale." biorxiv. https://doi.org/10.1101/2021.07.08.451480.

Alley, Ethan C., Grigory Khimulya, Surojit Biswas, Mohammed AlQuraishi, and George M. Church. 2019. "Unified Rational Protein Engineering with Sequence-Based Deep Representation Learning." *Nature Methods* 16 (12): 1315–22. https://doi.org/10.1038/s41592-019-0598-1.

Angluin, Dana. 1987. "Learning Regular Sets from Queries and Counterexamples." *Information and Computation* 75 (2): 87–106. https://doi.org/10.1016/0890-5401(87)90052-6.

Asgari, Ehsaneddin, Alice C. McHardy, and Mohammad R. K. Mofrad. 2019. "Probabilistic Variable-Length Segmentation of Protein Sequences for Discriminative Motif Discovery (DiMotif) and Sequence Embedding (ProtVecX)." *Scientific Reports* 9 (1): 3577. https://doi.org/10.1038/s41598-019-38746-w.

Asgari, Ehsaneddin, and Mohammad R. K. Mofrad. 2015. "Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics." *PLOS ONE* 10 (11): e0141287.

https://doi.org/10.1371/journal.pone.0141287.
Beltagy, Iz, Kyle Lo, and Arman Cohan. 2019. “SciBERT: A Pretrained Language Model for Scientific Text.” *ArXiv:1903.10676 [Cs]*, September. http://arxiv.org/abs/1903.10676.
Bender, Emily M., Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. “On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 🦜” In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 610–23. FAccT ’21. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3442188.3445922.
Bender, Emily M., and Alexander Koller. 2020. “Climbing towards NLU: On Meaning, Form, and Understanding in the Age of Data.” In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 5185–98. Online: Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.463.
Bepler, Tristan, and Bonnie Berger. 2021. “Learning the Protein Language: Evolution, Structure, and Function.” *Cell Systems* 12 (6): 654-669.e3. https://doi.org/10.1016/j.cels.2021.05.017.
Bhattamishra, Satwik, Kabir Ahuja, and Navin Goyal. 2020. “On the Ability and Limitations of Transformers to Recognize Formal Languages.” In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 7096–7116. Online: Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.576.
Brandes, Nadav, Dan Ofer, Yam Peleg, Nadav Rappoport, and Michal Linial. 2022. “ProteinBERT: A Universal Deep-Learning Model of Protein Sequence and Function.” *Bioinformatics*, January, btac020. https://doi.org/10.1093/bioinformatics/btac020.
Brown, Alex J., Igor Snapkov, Rahmad Akbar, Milena Pavlović, Enkelejda Miho, Geir K. Sandve, and Victor Greiff. 2019. “Augmenting Adaptive Immunity: Progress and Challenges in the Quantitative Engineering and Analysis of Adaptive Immune Receptor Repertoires.” *Molecular Systems Design & Engineering* 4 (4): 701–36. https://doi.org/10.1039/C9ME00071B.
Brown, Peter F, Vincent J Della Pietra, Robert L Mercer, Stephen A Della Pietra, and Jennifer C Lai. 1992. “An Estimate of an Upper Bound for the Entropy of English.” *Computational Linguistics* 18 (1): 10.
Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, et al. 2020. “Language Models Are Few-Shot Learners.” *ArXiv:2005.14165 [Cs]*, July. http://arxiv.org/abs/2005.14165.
Chen, Can, Jingbo Zhou, Fan Wang, Xue Liu, and Dejing Dou. 2022. “Structure-Aware Protein Self-Supervised Learning.” *ArXiv:2204.04213 [Cs, q-Bio]*, April. http://arxiv.org/abs/2204.04213.
Church, Kenneth. 2011. “A Pendulum Swung Too Far.” *Linguistic Issues in Language Technology* 6 (October). https://doi.org/10.33011/lilt.v6i.1245.
Church, Kenneth, and Mark Liberman. 2021. “The Future of Computational Linguistics: On Beyond Alchemy.” *Frontiers in Artificial Intelligence* 4. https://www.frontiersin.org/article/10.3389/frai.2021.625341.
Clark, Peter, Oyvind Tafjord, and Kyle Richardson. 2020. “Transformers as Soft Reasoners over Language.” In , 4:3882–90. https://doi.org/10.24963/ijcai.2020/537.
Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. “Unsupervised Cross-Lingual Representation Learning at Scale.” *ArXiv:1911.02116 [Cs]*, April. http://arxiv.org/abs/1911.02116.
Detlefsen, Nicki Skafte, Søren Hauberg, and Wouter Boomsma. 2022. “Learning Meaningful Representations of Protein Sequences.” *Nature Communications* 13 (1): 1914. https://doi.org/10.1038/s41467-022-29443-w.
Devi, Ganesan, Ashish V. Tendulkar, and Sutanu Chakraborti. 2017. “Protein Word Detection Using Text Segmentation Techniques.” In *BioNLP 2017*, 238–46. Vancouver, Canada,: Association for Computational Linguistics. https://doi.org/10.18653/v1/W17-2330.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. “BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding.” In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–86. Minneapolis, Minnesota: Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423.

Doddapaneni, Sumanth, Gowtham Ramesh, Mitesh M. Khapra, Anoop Kunchukuttan, and Pratyush Kumar. 2021. “A Primer on Pretrained Multilingual Language Models,” July. https://arxiv.org/abs/2107.00676v2.

Dyer, Chris, Adhiguna Kuncoro, Miguel Ballesteros, and Noah A. Smith. 2016. “Recurrent Neural Network Grammars.” *ArXiv:1602.07776 [Cs]*, October. http://arxiv.org/abs/1602.07776.

Elhanati, Yuval, Zachary Sethna, Quentin Marcou, Curtis G. Callan, Thierry Mora, and Aleksandra M. Walczak. 2015. “Inferring Processes Underlying B-Cell Repertoire Diversity.” *Philosophical Transactions of the Royal Society B: Biological Sciences* 370 (1676): 20140243. https://doi.org/10.1098/rstb.2014.0243.

Elnaggar, Ahmed, Michael Heinzinger, Christian Dallago, Ghalia Rihawi, Yu Wang, Llion Jones, Tom Gibbs, et al. 2021. “ProtTrans: Towards Cracking the Language of Life’s Code Through Self-Supervised Deep Learning and High Performance Computing.” *ArXiv:2007.06225 [Cs, Stat]*, May. http://arxiv.org/abs/2007.06225.

Eriguchi, Akiko, Yoshimasa Tsuruoka, and Kyunghyun Cho. 2017. “Learning to Parse and Translate Improves Neural Machine Translation.” arXiv:1702.03525. arXiv. https://doi.org/10.48550/arXiv.1702.03525.

Ettinger, Allyson. 2020. “What BERT Is Not: Lessons from a New Suite of Psycholinguistic Diagnostics for Language Models.” *Transactions of the Association for Computational Linguistics* 8 (December): 34–48. https://doi.org/10.1162/tacl_a_00298.

Eyraud, Rémi, and Stéphane Ayache. 2021. “Distillation of Weighted Automata from Recurrent Neural Networks Using a Spectral Approach.” *Machine Learning*, April. https://doi.org/10.1007/s10994-021-05948-1.

Firth, John R. 1957. “A Synopsis of Linguistic Theory 1930–55.” *Selected Papers of JR. Firth 1952–1959*.

Gage, Philip. 1994. “A New Algorithm for Data Compression.” *The C Users Journal* 12 (2): 23–38.

Gimona, Mario. 2006. “Protein Linguistics — a Grammar for Modular Protein Assembly?” *Nature Reviews Molecular Cell Biology* 7 (1): 68–73. https://doi.org/10.1038/nrm1785.

Gold, E Mark. 1967. “Language Identification in the Limit.” *Information and Control* 10 (5): 447–74. https://doi.org/10.1016/S0019-9958(67)91165-5.

Goldberg, Yoav. 2019. “Assessing BERT’s Syntactic Abilities.” *ArXiv:1901.05287 [Cs]*, January. http://arxiv.org/abs/1901.05287.

Greiff, Victor, Ulrike Menzel, Enkelejda Miho, Cédric Weber, René Riedel, Skylar Cook, Atijeh Valai, et al. 2017. “Systems Analysis Reveals High Genetic and Antigen-Driven Predetermination of Antibody Repertoires throughout B Cell Development.” *Cell Reports* 19 (7): 1467–78. https://doi.org/10.1016/j.celrep.2017.04.054.

Guest, Johnathan D., Thom Vreven, Jing Zhou, Iain Moal, Jeliazko R. Jeliazkov, Jeffrey J. Gray, Zhiping Weng, and Brian G. Pierce. 2021. “An Expanded Benchmark for Antibody-Antigen Docking and Affinity Prediction Reveals Insights into Antibody Recognition Determinants.” *Structure* 29 (6): 606-621.e5. https://doi.org/10.1016/j.str.2021.01.005.

Heinzinger, Michael, Ahmed Elnaggar, Yu Wang, Christian Dallago, Dmitrii Nechaev, Florian Matthes, and Burkhard Rost. 2019. “Modeling Aspects of the Language of Life through Transfer-Learning Protein Sequences.” *BMC Bioinformatics* 20 (1): 723. https://doi.org/10.1186/s12859-019-3220-8.

Hie, Brian L., Kevin K. Yang, and Peter S. Kim. 2022. “Evolutionary Velocity with Protein Language Models Predicts Evolutionary Dynamics of Diverse Proteins.” *Cell Systems*, February, S2405471222000382. https://doi.org/10.1016/j.cels.2022.01.003.

Hie, Brian, Ellen D. Zhong, Bonnie Berger, and Bryan Bryson. 2021. “Learning the Language of Viral Evolution and Escape.” *Science* 371 (6526): 284–88. https://doi.org/10.1126/science.abd7331.

Hofmann, Valentin, Janet Pierrehumbert, and Hinrich Schütze. 2021. “Superbizarre Is Not Superb: Derivational Morphology Improves BERT’s Interpretation of Complex Words.” In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 3594–3608. Online: Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.279.

Hofmann, Valentin, Hinrich Schütze, and Janet Pierrehumbert. 2022. “An Embarrassingly Simple Method to Mitigate Undesirable Properties of Pretrained Language Model Tokenizers.” In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 385–93. Dublin, Ireland: Association for Computational Linguistics. https://aclanthology.org/2022.acl-short.43.

Hu, Jennifer, Jon Gauthier, Peng Qian, Ethan Wilcox, and Roger Levy. 2020. “A Systematic Assessment of Syntactic Generalization in Neural Language Models.” In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 1725–44. Online: Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.158.

Jumper, John, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, et al. 2021. “Highly Accurate Protein Structure Prediction with AlphaFold.” *Nature* 596 (7873): 583–89. https://doi.org/10.1038/s41586-021-03819-2.

Kao, Wei-Tsung, and Hung-yi Lee. 2021. “Is BERT a Cross-Disciplinary Knowledge Learner? A Surprising Finding of Pre-Trained Models’ Transferability.” In *Findings of the Association for Computational Linguistics: EMNLP 2021*, 2195–2208. Punta Cana, Dominican Republic: Association for Computational Linguistics. https://aclanthology.org/2021.findings-emnlp.189.

Kaplan, Jared, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. “Scaling Laws for Neural Language Models.” arXiv:2001.08361. arXiv. https://doi.org/10.48550/arXiv.2001.08361.

Krishna, Kundan, Jeffrey Bigham, and Zachary C. Lipton. 2021. “Does Pretraining for Summarization Require Knowledge Transfer?” *ArXiv:2109.04953 [Cs]*, September. http://arxiv.org/abs/2109.04953.

Kudo, Taku, and John Richardson. 2018. “SentencePiece: A Simple and Language Independent Subword Tokenizer and Detokenizer for Neural Text Processing.” In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 66–71. Brussels, Belgium: Association for Computational Linguistics. https://doi.org/10.18653/v1/D18-2012.

Kutuzov, Andrey, and Elizaveta Kuzmenko. 2019. “To Lemmatize or Not to Lemmatize: How Word Normalisation Affects ELMo Performance in Word Sense Disambiguation.” *ArXiv:1909.03135 [Cs]*, September. http://arxiv.org/abs/1909.03135.

Lauscher, Anne, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. “From Zero to Hero: On the Limitations of Zero-Shot Cross-Lingual Transfer with Multilingual Transformers.” *ArXiv:2005.00633 [Cs]*, May. http://arxiv.org/abs/2005.00633.

Lee, Jinhyuk, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. “BioBERT: A Pre-Trained Biomedical Language Representation Model for Biomedical Text Mining.” *Bioinformatics* 36 (4): 1234–40. https://doi.org/10.1093/bioinformatics/btz682.

Leem, Jinwoo, Laura Sophie Mitchell, James Henry Royston Farmery, Justin Barton, and Jacob Daniel Galson. 2021. “Deciphering the Language of Antibodies Using Self-Supervised Learning.” https://doi.org/10.1101/2021.11.10.468064.

Lin, Tianyang, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. 2021. “A Survey of Transformers.” *ArXiv:2106.04554 [Cs]*, June. http://arxiv.org/abs/2106.04554.

Linzen, Tal. 2018. “What Can Linguistics and Deep Learning Contribute to Each Other?,” 1–12.

https://doi.org/arXiv:1809.04179v2.
Linzen, Tal, Emmanuel Dupoux, and Yoav Goldberg. 2016. “Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies.” *Transactions of the Association for Computational Linguistics* 4 (December): 521–35. https://doi.org/10.1162/tacl_a_00115.
Littmann, Maria, Michael Heinzinger, Christian Dallago, Tobias Olenyi, and Burkhard Rost. 2021. “Embeddings from Deep Learning Transfer GO Annotations beyond Homology.” *Scientific Reports* 11 (1): 1160. https://doi.org/10.1038/s41598-020-80786-0.
Liu, Chi-Liang, Tsung-Yuan Hsu, Yung-Sung Chuang, and Hung-Yi Lee. 2020. “A Study of Cross-Lingual Ability and Language-Specific Information in Multilingual BERT.” *ArXiv:2004.09205 [Cs]*, April. http://arxiv.org/abs/2004.09205.
Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. “RoBERTa: A Robustly Optimized BERT Pretraining Approach.” *ArXiv:1907.11692 [Cs]*, July. http://arxiv.org/abs/1907.11692.
Madani, Ali, Ben Krause, Eric R. Greene, Subu Subramanian, Benjamin P. Mohr, James M. Holton, Jose Luis Olmos, et al. 2021. “Deep Neural Language Modeling Enables Functional Protein Generation across Families.” *BioRxiv*, July, 2021.07.18.452833. https://doi.org/10.1101/2021.07.18.452833.
Marcou, Quentin, Thierry Mora, and Aleksandra M. Walczak. 2018. “High-Throughput Immune Repertoire Analysis with IGoR.” *Nature Communications* 9 (1): 561. https://doi.org/10.1038/s41467-018-02832-w.
McCoy, R. Thomas, Robert Frank, and Tal Linzen. 2020. “Does Syntax Need to Grow on Trees? Sources of Hierarchical Inductive Bias in Sequence-to-Sequence Networks.” *Transactions of the Association for Computational Linguistics* 8 (December): 125–40. https://doi.org/10.1162/tacl_a_00304.
McCoy, R. Thomas, Ellie Pavlick, and Tal Linzen. 2019. “Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference.” In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3428–48. Florence, Italy: Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-1334.
McCoy, R. Thomas, Paul Smolensky, Tal Linzen, Jianfeng Gao, and Asli Celikyilmaz. 2021. “How Much Do Language Models Copy from Their Training Data? Evaluating Linguistic Novelty in Text Generation Using RAVEN.” *ArXiv:2111.09509 [Cs]*, November. http://arxiv.org/abs/2111.09509.
Meier, Joshua, Roshan Rao, Robert Verkuil, Jason Liu, Tom Sercu, and Alexander Rives. 2021. “Language Models Enable Zero-Shot Prediction of the Effects of Mutations on Protein Function.” *BioRxiv*, July, 2021.07.09.450648. https://doi.org/10.1101/2021.07.09.450648.
Mielke, Sabrina J., Zaid Alyafeai, Elizabeth Salesky, Colin Raffel, Manan Dey, Matthias Gallé, Arun Raja, et al. 2021. “Between Words and Characters: A Brief History of Open-Vocabulary Modeling and Tokenization in NLP.” *ArXiv:2112.10508 [Cs]*, December. http://arxiv.org/abs/2112.10508.
Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. “Distributed Representations of Words and Phrases and Their Compositionality.” *ArXiv:1310.4546 [Cs, Stat]*, October. http://arxiv.org/abs/1310.4546.
Mikolov, Tomas, Wen-tau Yih, and Geoffrey Zweig. 2013. “Linguistic Regularities in Continuous Space Word Representations.” In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 746–51. Atlanta, Georgia: Association for Computational Linguistics. https://aclanthology.org/N13-1090.
Montague, Richard. 1970. “Universal Grammar.” *Theoria* 36 (3): 373–93.
Morris, Tim P., Ian R. White, and Michael J. Crowther. 2019. “Using Simulation Studies to Evaluate Statistical Methods.” *Statistics in Medicine* 38 (11): 2074–2102. https://doi.org/10.1002/sim.8086.
Naseem, Usman, Imran Razzak, Shah Khalid Khan, and Mukesh Prasad. 2021. “A Comprehensive Survey on Word Representation Models: From Classical to State-of-the-Art Word Representation Language Models.” *ACM Transactions on Asian and Low-Resource Language Information*

Processing 20 (5): 1–35. https://doi.org/10.1145/3434237.
Nijkamp, Erik, Jeffrey Ruffolo, Eli N. Weinstein, Nikhil Naik, and Ali Madani. 2022. "ProGen2: Exploring the Boundaries of Protein Language Models." arXiv. https://doi.org/10.48550/arXiv.2206.13517.
Niven, Timothy, and Hung-Yu Kao. 2019. "Probing Neural Network Comprehension of Natural Language Arguments." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4658–64. Florence, Italy: Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-1459.
Ofer, Dan, Nadav Brandes, and Michal Linial. 2021. "The Language of Proteins: NLP, Machine Learning & Protein Sequences." *Computational and Structural Biotechnology Journal* 19 (January): 1750–58. https://doi.org/10.1016/j.csbj.2021.03.022.
Olsen, Tobias H., Fergus Boyles, and Charlotte M. Deane. 2022. "Observed Antibody Space: A Diverse Database of Cleaned, Annotated, and Translated Unpaired and Paired Antibody Sequences." *Protein Science* 31 (1): 141–46. https://doi.org/10.1002/pro.4205.
Olsen, Tobias Hegelund, Iain H. Moal, and Charlotte M. Deane. 2022. "AbLang: An Antibody Language Model for Completing Antibody Sequences." bioRxiv. https://doi.org/10.1101/2022.01.20.477061.
Ostrovsky-Berman, Miri, Boaz Frankel, Pazit Polak, and Gur Yaari. 2021. "Immune2vec: Embedding B/T Cell Receptor Sequences in ℝN Using Natural Language Processing." *Frontiers in Immunology* 0. https://doi.org/10.3389/fimmu.2021.680687.
Pan, Sinno Jialin, and Qiang Yang. 2010. "A Survey on Transfer Learning." *IEEE Transactions on Knowledge and Data Engineering* 22 (10): 1345–59. https://doi.org/10.1109/TKDE.2009.191.
Pan, Yirong, Xiao Li, Yating Yang, and Rui Dong. 2020. "Morphological Word Segmentation on Agglutinative Languages for Neural Machine Translation." arXiv:2001.01589. arXiv. https://doi.org/10.48550/arXiv.2001.01589.
Peters, Matthew E., Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. "Deep Contextualized Word Representations." *ArXiv:1802.05365 [Cs]*, March. http://arxiv.org/abs/1802.05365.
Pinter, Yuval. 2021. "Integrating Approaches to Word Representation." *ArXiv:2109.04876 [Cs]*, September. http://arxiv.org/abs/2109.04876.
Qiu, Xipeng, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. "Pre-Trained Models for Natural Language Processing: A Survey." *Science China Technological Sciences* 63 (10): 1872–97. https://doi.org/10.1007/s11431-020-1647-3.
Rae, Jack W., Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, et al. 2022. "Scaling Language Models: Methods, Analysis & Insights from Training Gopher." arXiv:2112.11446. arXiv. https://doi.org/10.48550/arXiv.2112.11446.
Rao, Roshan, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Xi Chen, John Canny, Pieter Abbeel, and Yun S. Song. 2019. "Evaluating Protein Transfer Learning with TAPE." *Advances in Neural Information Processing Systems* 32 (December): 9689–9701.
Rives, Alexander, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, et al. 2021. "Biological Structure and Function Emerge from Scaling Unsupervised Learning to 250 Million Protein Sequences." *Proceedings of the National Academy of Sciences* 118 (15). https://doi.org/10.1073/pnas.2016239118.
Robert, Philippe A., Rahmad Akbar, Robert Frank, Milena Pavlović, Michael Widrich, Igor Snapkov, Maria Chernigovskaya, et al. 2021. "One Billion Synthetic 3D-Antibody-Antigen Complexes Enable Unconstrained Machine-Learning Formalized Investigation of Antibody Specificity Prediction." *BioRxiv*, July, 2021.07.06.451258. https://doi.org/10.1101/2021.07.06.451258.
Rogers, Anna, Olga Kovaleva, and Anna Rumshisky. 2021. "A Primer in BERTology: What We Know About How BERT Works." *Transactions of the Association for Computational Linguistics* 8 (January): 842–66. https://doi.org/10.1162/tacl_a_00349.
Rudin, Cynthia. 2019. "Stop Explaining Black Box Machine Learning Models for High Stakes Decisions

and Use Interpretable Models Instead." *Nature Machine Intelligence* 1 (5): 206–15. https://doi.org/10.1038/s42256-019-0048-x.

Ruffolo, Jeffrey A., Jeffrey J. Gray, and Jeremias Sulam. 2021. "Deciphering Antibody Affinity Maturation with Language Models and Weakly Supervised Learning." *ArXiv:2112.07782 [Cs, q-Bio]*, December. http://arxiv.org/abs/2112.07782.

Ruffolo, Jeffrey A., Jeremias Sulam, and Jeffrey J. Gray. 2022. "Antibody Structure Prediction Using Interpretable Deep Learning." *Patterns* 3 (2): 100406. https://doi.org/10.1016/j.patter.2021.100406.

Schluter, Natalie. 2018. "The Word Analogy Testing Caveat." In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 242–46. New Orleans, Louisiana: Association for Computational Linguistics. https://doi.org/10.18653/v1/N18-2039.

Schwartz, Lane, Francis Tyers, Lori Levin, Christo Kirov, Patrick Littell, Chi-kiu Lo, Emily Prud'hommeaux, et al. 2020. "Neural Polysynthetic Language Modelling." arXiv:2005.05477. arXiv. https://doi.org/10.48550/arXiv.2005.05477.

Shannon, C. E. 1951. "Prediction and Entropy of Printed English." *The Bell System Technical Journal* 30 (1): 50–64. https://doi.org/10.1002/j.1538-7305.1951.tb01366.x.

Shin, Seongjin, Sang-Woo Lee, Hwijeen Ahn, Sungdong Kim, HyoungSeok Kim, Boseop Kim, Kyunghyun Cho, et al. 2022. "On the Effect of Pretraining Corpora on In-Context Learning by a Large-Scale Language Model." arXiv:2204.13509. arXiv. https://doi.org/10.48550/arXiv.2204.13509.

Shuai, Richard W., Jeffrey A. Ruffolo, and Jeffrey J. Gray. 2021. "Generative Language Modeling for Antibody Design." https://doi.org/10.1101/2021.12.13.472419.

Stern, Mitchell, William Chan, Jamie Kiros, and Jakob Uszkoreit. 2019. "Insertion Transformer: Flexible Sequence Generation via Insertion Operations." In *Proceedings of the 36th International Conference on Machine Learning*, 5976–85. PMLR. https://proceedings.mlr.press/v97/stern19a.html.

Strait, B. J., and T. G. Dewey. 1996. "The Shannon Information Entropy of Protein Sequences." *Biophysical Journal* 71 (1): 148–55. https://doi.org/10.1016/S0006-3495(96)79210-X.

Szymborski, Joseph, and Amin Emad. 2022. "RAPPPID: Towards Generalisable Protein Interaction Prediction with AWD-LSTM Twin Networks." bioRxiv. https://doi.org/10.1101/2021.08.13.456309.

Tai, Kai Sheng, Richard Socher, and Christopher D. Manning. 2015. "Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks." In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 1556–66. Beijing, China: Association for Computational Linguistics. https://doi.org/10.3115/v1/P15-1150.

Tenney, Ian, Dipanjan Das, and Ellie Pavlick. 2019. "BERT Rediscovers the Classical NLP Pipeline." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4593–4601. Florence, Italy: Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-1452.

Unsal, Serbulent, Heval Atas, Muammer Albayrak, Kemal Turhan, Aybar C. Acar, and Tunca Doğan. 2022. "Learning Functional Properties of Proteins with Language Models." *Nature Machine Intelligence* 4 (3): 227–45. https://doi.org/10.1038/s42256-022-00457-9.

Vig, Jesse, Ali Madani, Lav R. Varshney, Caiming Xiong, Richard Socher, and Nazneen Fatema Rajani. 2021. "BERTology Meets Biology: Interpreting Attention in Protein Language Models." *ArXiv:2006.15222 [Cs, q-Bio]*, March. http://arxiv.org/abs/2006.15222.

Villegas-Morcillo, Amelia, Stavros Makrodimitris, Roeland C H J van Ham, Angel M Gomez, Victoria Sanchez, and Marcel J T Reinders. 2021. "Unsupervised Protein Embeddings Outperform Hand-Crafted Sequence and Structure Features at Predicting Molecular Function." *Bioinformatics* 37 (2): 162–70. https://doi.org/10.1093/bioinformatics/btaa701.

Vries, Wietse de, Martijn Wieling, and Malvina Nissim. 2022. “Make the Best of Cross-Lingual Transfer: Evidence from POS Tagging with over 100 Languages.” In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.

Wang, Qinglong, Kaixuan Zhang, Alexander G. Ororbia II, Xinyu Xing, Xue Liu, and C. Lee Giles. 2018. “An Empirical Evaluation of Rule Extraction from Recurrent Neural Networks.” *Neural Computation* 30 (9): 2568–91. https://doi.org/10.1162/neco_a_01111.

Wang, Yanbin, Zhu-Hong You, Shan Yang, Xiao Li, Tong-Hai Jiang, and Xi Zhou. 2019. “A High Efficient Biological Language Model for Predicting Protein–Protein Interactions.” *Cells* 8 (2): 122. https://doi.org/10.3390/cells8020122.

Warstadt, Alex, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel Bowman. 2020. “BLiMP: A Benchmark of Linguistic Minimal Pairs for English.” *Proceedings of the Society for Computation in Linguistics* 3 (1): 437–38. https://doi.org/10.7275/zejz-qs04.

Weber, Cédric R, Rahmad Akbar, Alexander Yermanos, Milena Pavlović, Igor Snapkov, Geir K Sandve, Sai T Reddy, and Victor Greiff. 2020. “ImmuneSIM: Tunable Multi-Feature Simulation of B- and T-Cell Receptor Repertoires for Immunoinformatics Benchmarking.” *Bioinformatics* 36 (11): 3594–96. https://doi.org/10.1093/bioinformatics/btaa158.

Weiss, Gail, Yoav Goldberg, and Eran Yahav. 2020. “Extracting Automata from Recurrent Neural Networks Using Queries and Counterexamples.” *ArXiv:1711.09576 [Cs]*, February. http://arxiv.org/abs/1711.09576.

Welleck, Sean, Kianté Brantley, Hal Daumé Iii, and Kyunghyun Cho. 2019. “Non-Monotonic Sequential Text Generation.” In *Proceedings of the 36th International Conference on Machine Learning*, 6716–26. PMLR. https://proceedings.mlr.press/v97/welleck19a.html.

Xu, Jingjing, Hao Zhou, Chun Gan, Zaixiang Zheng, and Lei Li. 2021. “Vocabulary Learning via Optimal Transport for Neural Machine Translation.” In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 7361–73. Online: Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.571.

Xu, Minghao, Zuobai Zhang, Jiarui Lu, Zhaocheng Zhu, Yangtian Zhang, Chang Ma, Runcheng Liu, and Jian Tang. 2022. “PEER: A Comprehensive and Multi-Task Benchmark for Protein Sequence Understanding.” arXiv. https://doi.org/10.48550/arXiv.2206.02096.

Yang, Kevin K, Zachary Wu, Claire N Bedbrook, and Frances H Arnold. 2018. “Learned Protein Embeddings for Machine Learning.” *Bioinformatics* 34 (15): 2642–48. https://doi.org/10.1093/bioinformatics/bty178.