Estimating probabilistic context-free grammars for proteins using contact map constraints

Witold Dyrka Corresp., [1] , **François Coste** [2] , **Hugo Talibart** [2]

[1] Wydział Podstawowych Problemów Techniki, Katedra Inżynierii Biomedycznej, Politechnika Wrocławska, Wrocław, Poland

[2] Univ Rennes, Inria, CNRS, IRISA, Rennes, France

Corresponding Author: Witold Dyrka
Email address: witold.dyrka@pwr.edu.pl

Learning language of protein sequences, which captures non-local interactions between amino acids close in the spatial structure, is a long-standing bioinformatics challenge, which requires at least context-free grammars. However, the complex nature of protein interactions impedes unsupervised learning of context-free grammars. Using structural information to constrain the syntactic trees proved effective in learning probabilistic natural and RNA languages. In this work, we establish a framework for learning probabilistic context-free grammars for protein sequences from syntactic trees partially constrained using amino acid contacts obtained from wet experiments and computational predictions, whose reliability has substantially increased recently. Within the framework, we implement the maximum-likelihood and contrastive estimators of parameters for simple yet practical grammars. Tested on samples of protein motifs, grammars developed within the framework showed improved precision in recognizing sequences and generated parse trees with high fidelity to protein structures. The framework is applicable to other biomolecular languages and beyond wherever knowledge of non-local dependencies is available.

Estimating probabilistic context-free grammars for proteins using contact map constraints

Witold Dyrka*

Politechnika Wrocławska, Wydział Podstawowych Problemów Techniki, Katedra Inżynierii Biomedycznej, Poland

François Coste and Hugo Talibart

Univ Rennes, Inria, CNRS, IRISA, France

Abstract

Learning language of protein sequences, which captures non-local interactions between amino acids close in the spatial structure, is a long-standing bioinformatics challenge, which requires at least context-free grammars. However, the complex nature of protein interactions impedes unsupervised learning of context-free grammars. Using structural information to constrain the syntactic trees proved effective in learning probabilistic natural and RNA languages. In this work, we establish a framework for learning probabilistic context-free grammars for protein sequences from syntactic trees partially constrained using amino acid contacts obtained from wet experiments and computational predictions, whose reliability has substantially increased recently. Within the framework, we implement the maximum-likelihood and contrastive estimators of parameters for simple yet practical grammars. Tested on samples of protein motifs, grammars developed within the framework showed improved precision in recognizing sequences and generated parse trees with high fidelity to protein structures. The framework is applicable to other biomolecular languages and beyond wherever knowledge of non-local dependencies is available.

Keywords: *probabilistic context-free grammar, syntactic tree, structural constraints, protein sequence, protein contact map, maximum-likelihood estimator, contrastive estimation*

1 Introduction

1.1 Grammatical modeling of proteins

The essential biopolymers of life, nucleic acids and proteins, share the basic characteristic of the languages: infinite number of sequences can be expressed with a finite number of monomers. In

*Corresponding author: witold.dyrka@pwr.edu.pl

the case of proteins, merely 20 amino acid species (letters) build millions of sequences (words or sentences) folded in thousands of different spatial structures playing various functions in living organisms (semantics). Physically, the protein sequence is a chain of amino acids linked by peptide bonds. The physicochemical properties of amino acids and their interactions across different parts of the sequence define its spatial structure, which in turn determines biological function to great extent. Similarly to words of the natural language, protein sequences may be ambiguous (the same amino acid sequence folds into different structures depending on the environment), and often include non-local dependencies and recursive structures [Searls, 2013].

Not surprisingly the concept of *protein language* dates back to at least the 1960s [Pawlak, 1965], and since early applied works in the 1980s [Brendel and Busse, 1984, Jimenez-Montano, 1984] formal grammatical models have gradually gained importance in bioinformatics [Searls, 2002, 2013, Coste, 2016]. Most notably, Hidden Markov Models (HMM), which are weakly equivalent to probabilistic regular grammars, became the main tool of protein sequence analysis. Profile HMMs are commonly used for defining protein families [Sonnhammer et al., 1998, Finn et al., 2015] and for searching similar sequences [Eddy, 1998, 2011, Soeding, 2005, Remmert et al., 2012]; more expressive HMM are also developed [Coste and Kerbellec, 2006, Bretaudeau et al., 2012]. Yet, their explanatory power is limited since, as regular level models, they cannot capture non-local interactions, which occur between amino acids distant in sequence but close in the spatial structure of the protein. Many of these interactions have a character of nested, branched and crossing dependencies, which in terms of grammatical modeling require context-free (CF) and context-sensitive (CS) level of expressiveness [Searls, 2013]. However, grammatical models beyond regular levels have been rather scarcely applied to protein analysis (a comprehensive list of references can be found in [Dyrka et al., 2013]). This is in contrast to RNA modeling, where CF grammatical frameworks are well-developed and power some of the most successful tools [Sakakibara et al., 1993, Eddy and Durbin, 1994, Knudsen and Hein, 1999, Sükösd et al., 2012].

One difficulty with modeling proteins is that interactions between amino acids are often less specific and more *collective* in comparison to RNA. Moreover, the larger alphabet made of 20 amino acid species instead of just 4 bases in nucleic acids, combined with high computational complexity of CF and CS grammars, impedes inference, which may lead to solutions which do not outperform significantly HMMs [Dyrka and Nebel, 2009, Dyrka et al., 2013]. Yet, some studies hinted that CF level of expressiveness brought an added value in protein modeling when grammars fully benefiting from CF nesting and branching rules were compared in the same framework to grammars effectively limited to linear (regular) rules [Dyrka, 2007, Dyrka et al., 2013]. Good preliminary results were also obtained on learning sub-classes of CF grammars to model protein families, showing the interest of taking into account long-distance correlations in comparison to regular models [Coste et al., 2012, 2014].

An important advantage of CF and CS grammars is that parse trees they produce are human readable descriptors. In RNA modeling, the shape of parse trees can be used for secondary structure prediction [Dowell and Eddy, 2004]. In protein modeling, it was suggested that the shape of parse trees corresponded to protein spatial structure [Dyrka and Nebel, 2009], and that parse trees can convey biologically relevant information [Sciacca et al., 2011, Dyrka et al., 2013].

1.2 Grammar estimation with structural constraints

In this piece of research the focus is on learning probabilistic context-free grammars (PCFG) [Booth, 1969]. Learning PCFG consists in estimating the unfixed parameters of the grammar with the aim of shifting the probability mass from the entire space of possible sequences and their syntactic trees to the target population, typically represented by a sample. The problem is often confined to assigning probabilities to fixed production rules of a generic underlying non-probabilistic CFG [Lari and Young, 1990]. Typically the goal is to estimate the parameters to get a grammar maximizing the likelihood of the (positive) sample, while, depending on the target application, other approaches also exist. For example, the contrastive estimation aims at obtaining grammars discriminating target population from its neighborhood [Smith and Eisner, 2005].

The training sample can be made of a set of sequences or a set of syntactic trees. In the former case, all derivations for each sentence are considered valid. For a given underlying non-probabilistic CFG, probabilities of its rules can be estimated from sentences in the classical Expectation Maximization framework (e.g. the Inside-Outside algorithm [Baker, 1979, Lari and Young, 1990]). However, the approach is not guaranteed to find the globally optimal solution [Carroll and Charniak, 1992]. Heuristic methods applied for learning PCFG from positive sequences include also iterative biclustering of bigrams [Tu and Honavar, 2008], and genetic algorithms using a learnable set of rules [Kammeyer and Belew, 1996, Keller and Lutz, 1998, 2005] or a fixed covering set of rules [Tariman, 2004, Dyrka and Nebel, 2009].

Much more information about the language is conveyed in the syntactic trees. If available, a set of trees (a treebank) can be directly used to learn a PCFG [Charniak, 1996]. Usability of structural information is highlighted by the result showing that a large class of non-probabilistic CFG can be learned from unlabeled syntactic trees (called also *skeletons*) of the training sample [Sakakibara, 1992]. Algorithms for learning probabilistic CF languages, which exploit structural information in syntactic trees, have been proposed [Sakakibara et al., 1993, Eddy and Durbin, 1994, Carrasco et al., 2001, Cohen et al., 2014]. An interesting middle way between plain sequences and syntactic trees are partially bracketed sequences, which constrain the shape of the syntactic trees (skeletons) but not node labels. The approach was demonstrated to be highly effective in learning natural languages [Pereira and Schabes, 1992]. It was also applied to integrating uncertain information on pairing of nucleotides of RNA [Knudsen, 2005]. In this approach the modified bottom-up parser penalizes probability on derivations inconsistent with available information on nucleotide pairing in such way that the amount of the penalty is adjusted according to certainty of the structural information.

1.3 Protein contact constraints

To our knowledge constrained sets of syntactic trees have never been applied for estimating PCFG for proteins. In this research we propose to use spatial contacts between amino acids distant in sequence as a source of constraints. Indeed, an interaction forming dependency between amino acids usually requires a contact between them, defined as spatial proximity. Until recently, extensive contact maps were only available for proteins with experimentally solved structures, while individual interactions could be determined through mutation-based wet experiments.

Currently, reasonably reliable contact maps can also be obtained computationally from large collective alignments of evolutionary related sequences. The rationale for contact prediction is that

if amino acids at a pair of positions in the alignment interact then a mutation at one position of the pair often requires a compensatory mutation at the other position in order to maintain the interaction intact. Since only proteins maintaining interactions vital for function successfully endured the natural selection, an observable correlation in amino acid variability at a pair of positions is expected to indicate interaction. However, standard correlations are transitive and therefore cannot be immediately used as interaction predictors. The break-through was achieved recently by Direct Coupling Analysis (DCA)[Weigt et al., 2009], which disentangles direct from indirect correlations by inferring a model on the alignment which can give information on the interaction strength of the pairs. There are different DCA methods based on how the model, which is usually a type of the Markov Random Field, is obtained [Morcos et al., 2011, Jones et al., 2012, Ekeberg et al., 2013, Kamisetty et al., 2013, Seemayer et al., 2014, Baldassi et al., 2014]. The state-of-the-art DCA-based meta-algorithms achieve mean precision in the range 42-74% for top L predicted contacts and 69-98% for top $L/10$ predicted contacts, where L is the protein length [Wang et al., 2017]. Precision is usually lower for shorter sequences and especially for smaller alignments, however a few top hits may still provide relevant information [Daskalov et al., 2015].

2 Methods

2.1 General model

2.1.1 Basic notations

Let Σ be a non-empty finite set of atomic symbols (representing for instance amino acid species). The set of all finite strings over this alphabet is denoted by Σ^*. Let $|x|$ denote the length of a string x. The set of all strings of length n is denoted by $\Sigma^n = \{x \in \Sigma^* \colon |x| = n\}$. Let $x = x_1 \dots x_n$ be a sequence in Σ^n.

Unlabeled syntactic tree An unlabeled syntactic tree (UST) u for x is an ordered rooted tree such that the leaf nodes are labeled by x, which is denoted as $yield(u) = x$, and the non-leaf nodes are unlabeled. Let $\mathscr{U}_*$ denotes the set of all USTs that yield a sequence in Σ^*, let $\mathscr{U}_n = \{u \in \mathscr{U}_* : yield(u) \in \Sigma^n\}$, where n is a positive integer, and let $\mathscr{U}_x = \{u \in \mathscr{U}_* : yield(u) = x \in \Sigma^*\}$. Note that $\forall (x, w \in \Sigma^*,\ x \neq w)\ \mathscr{U}_x \cap \mathscr{U}_w = \emptyset$ and $\mathscr{U}_* = \cup_{x \in \Sigma^*} \mathscr{U}_x$. Moreover, let U denotes an arbitrary subset of $\mathscr{U}_*$.

Context-free grammar A context-free grammar (CFG) is a quadruple $G = \langle \Sigma, V, v_0, R \rangle$, where Σ is defined as above, V is a finite set of non-terminal symbols (also called variables) disjoint from Σ, $v_0 \in V$ is a special start symbol, and R is a finite set of rules rewriting from variables into strings of variables and/or terminals $R = \{r_i \colon V \to (\Sigma \cup V)^*\}$. Let $\alpha = \alpha_1 \dots \alpha_k$ be a sequence of symbols in $(\Sigma \cup V)^k$ for some natural k. A (left-most) derivation for G is a string of rules $r = r_1 \dots r_l \in R^l$, which defines an ordered parse tree y starting from the root node labeled by v_0. In each step, by applying a rule $r_i : v_j \to \alpha_1 \dots \alpha_k$, tree y is extended by adding edges from the already existing left-most node labeled v_j to newly added nodes labeled α_1 to α_k. Therefore, there is a one-to-one correspondence between derivation r and parse tree y. Derivation r is complete if all leaf nodes of the corresponding (complete) parse tree y are labeled by symbols in Σ. Sets $\mathscr{Y}_*$, $\mathscr{Y}_n$ and $\mathscr{Y}_x$

are defined as for the USTs. For a given parse tree y, $u(y)$ denotes the unlabeled syntactic tree obtained by removing the non-leaf labels on y. Given a UST u, let $\mathscr{Y}_G(u)$ be the set of all parse trees for grammar G such that $u(y) = u$. For a set of USTs U, $\mathscr{Y}_G(U) = \cup_{u \in U} \mathscr{Y}_G(u)$. Note that $\forall (u, v \in U,\ u \neq v)\ \mathscr{Y}_G(u) \cap \mathscr{Y}_G(v) = \emptyset$.

Probabilistic context-free grammar A probabilistic context-free grammar (PCFG) is a quintuple $\mathscr{G} = \langle \Sigma, V, v_0, R, \theta \rangle$, where θ is a finite set of probabilities of rules: $\theta = \{\theta_i = \theta(r_i) : R \to [0,1]\}$, setting for each rule $v_k \to \alpha$ its probability to be chosen to rewrite v_k with respect to other rules rewriting v_k (such that $\forall (v_k \in V)\ \sum_{v_k \to \alpha} \theta(v_k \to \alpha) = 1$). Let PCFG $\mathscr{G}$ that enhances the underlying non-probabilistic CFG $G = \langle \Sigma, V, v_0, R \rangle$ is denoted by $\mathscr{G} = \langle G, \theta \rangle$. The probability of parse tree y using the probability measure induced by $\mathscr{G}$ is given by the probability of the corresponding derivation $r = r_1 \dots r_l$:

$$prob(y \mid \mathscr{G}) = prob(r \mid \mathscr{G}) = \prod_{i=1}^{l} \theta(r_i).$$

$\mathscr{G}$ is said to be *consistent* when it defines probability distribution over $\mathscr{Y}_*$:

$$prob(\mathscr{Y}_* \mid \mathscr{G}) = \sum_{y \in \mathscr{Y}_*} prob(y \mid \mathscr{G}) = 1.$$

The probability of sequence $x \in \Sigma^*$ given $\mathscr{G}$ is:

$$prob(x \mid \mathscr{G}) = prob(\mathscr{Y}_x \mid \mathscr{G}) = \sum_{y \in \mathscr{Y}_x} prob(y \mid \mathscr{G}),$$

and the probability of UST $u \in \mathscr{U}_x$ given $\mathscr{G}$ is:

$$prob(u \mid \mathscr{G}) = prob(\mathscr{Y}_G(u) \mid \mathscr{G}) = \sum_{y \in \mathscr{Y}_G(u)} prob(y \mid \mathscr{G}).$$

Since $\mathscr{Y}_x$ and $\mathscr{Y}_G(u)$ define each a partition of $\mathscr{Y}_*$ for $x \in \Sigma^*$ and for $u \in \mathscr{U}_*$, a consistent grammar $\mathscr{G}$ defines also a probability distribution over Σ^* and $\mathscr{U}_*$.

2.1.2 Contact constraints

Most protein sequences fold into complex spatial structures. Two amino acids at positions i and j in the sequence x are said to be in contact if distance between their coordinates in spatial structure $d(i,j)$ is below a given threshold τ. A full contact map for a protein of length n is a binary symmetric matrix $\mathsf{m}^{\text{full}} = (m_{i,j})_{n \times n}$ such that $m_{i,j} = [d(i,j) < \tau]$, where $[x]$ is the Iverson bracket. Usually only a subset of the contacts is considered (cf section 1.3). A (partial) contact map for a protein of length n is a binary symmetric matrix $\mathsf{m} = (m_{i,j})_{n \times n}$ such that $m_{i,j} = 1 \implies d(i,j) < \tau$. Let $d_u(i,j)$ is the length of the path from i-th to j-th leaf in UST u for x. Given a threshold δ, UST u is said to be consistent with a contact map m of length n if $m_{i,j} = 1 \implies d_u(i,j) < \delta$.

For a contact map m of length n, let $\mathscr{U}_n^{\mathsf{m}}$ denotes the subset of $\mathscr{U}_n$ consistent with m, and $\mathscr{U}_x^{\mathsf{m}}$ denotes the subset of $\mathscr{U}_x$ consistent with m. Note that $\mathscr{U}_x^{\mathsf{m}} = \mathscr{U}_n^{\mathsf{m}} \cap \mathscr{U}_x$. Analogous notations apply to parse trees.

2.1.3 Estimation

Learning grammar $\mathscr{G} = \langle \Sigma, V, v_0, R, \theta \rangle$ can be seen as estimating the unfixed parameters of $\mathscr{G}$ with the aim of shifting the probability mass from the entire space of unlabeled syntactic trees $\mathscr{U}_*$ to the set of unlabeled syntactic trees for the target population $\mathfrak{U}_{\text{target}}$. In practice, only a sample of the target population can be used for learning, hence estimation is performed on $\mathfrak{U}_{\text{sample}} \subseteq \mathfrak{U}_{\text{target}}$. Note that even in the most general case the set of terminal symbols Σ is implicitly determined by the sample; moreover the start symbol v_0 is typically also fixed. A common special case confines learning grammar $\mathscr{G}$ to estimating θ for a fixed quadruple of non-probabilistic parameters $\langle \Sigma, V, v_0, R \rangle$ (which fully determine the non-probabilistic grammar G underlying $\mathscr{G}$). Given inferred grammar $\mathscr{G}_*$ and a query set of unlabeled syntactic trees $\mathscr{U}_{\text{query}}$, probability $prob(\mathscr{U}_{\text{query}} \mid \mathscr{G}_*)$ is an estimator of the likelihood that $\mathscr{U}_{\text{query}}$ belongs to population $\mathfrak{U}_{\text{target}}$.

Maximum-likelihood grammar Let X be a sample set of sequences in Σ^*, and let M be a set of corresponding contact matrices. The sample set $\mathscr{S} = [X\mathsf{M}]$ consists of a set of tuples (x, m), where $x \in X$ and $\mathsf{m} \in \mathsf{M}$. Let $\mathfrak{U}_X^{\mathsf{M}}$ be the corresponding set of compatible USTs:

$$\mathfrak{U}_X^{\mathsf{M}} = \{\mathscr{U}_x^{\mathsf{m}} : (x, \mathsf{m}) \in \mathscr{S}\}.$$

Grammar $\mathscr{G}$ that concentrates probability mass on $\mathfrak{U}_X^{\mathsf{M}}$ can be estimated using the classical Bayesian approach:

$$\mathscr{G}_* = \arg\max_{\mathscr{G}} prob(\mathscr{G} \mid \mathfrak{U}_X^{\mathsf{M}}) = \arg\max_{\mathscr{G}} \frac{prob(\mathscr{G}) \cdot prob(\mathfrak{U}_X^{\mathsf{M}} \mid \mathscr{G})}{prob(\mathfrak{U}_X^{\mathsf{M}})}.$$

Noting that $prob(\mathfrak{U}_X^{\mathsf{M}})$ does not influence the result and, in the lack of prior knowledge, assuming $prob(\mathscr{G})$ uniformly distributed among all $\mathscr{G}$, the solution is then given by the maximum likelihood formula:

$$\mathscr{G}_* = \arg\max_{\mathscr{G}} prob(\mathscr{G} \mid \mathfrak{U}_X^{\mathsf{M}}) \simeq \mathscr{G}_{\text{ML}} = \arg\max_{\mathscr{G}} prob(\mathfrak{U}_X^{\mathsf{M}} \mid \mathscr{G}).$$

Assuming independence of $\mathscr{U}_x^{\mathsf{m}}$s:

$$\mathscr{G}_{\text{ML}} = \arg\max_{\mathscr{G}} \prod_{\mathscr{U}_x^{\mathsf{m}} \in \mathfrak{U}_X^{\mathsf{M}}} prob(\mathscr{U}_x^{\mathsf{m}} \mid \mathscr{G}) = \arg\max_{\mathscr{G}} \prod_{(x,\mathsf{m}) \in \mathscr{S}} \sum_{y \in \mathscr{Y}_x^{\mathsf{m}}} prob(y \mid \mathscr{G}).$$

In the absence of contact constraints, the maximization problem becomes equivalent to the standard problem of estimating grammar $\mathscr{G}$ given the sample X:

$$\mathscr{G}_{\text{ML}}^{\mathsf{m}=0} = \arg\max_{\mathscr{G}} \prod_{\mathscr{U}_x \in \mathfrak{U}_X} prob(\mathscr{U}_x \mid \mathscr{G}) = \arg\max_{\mathscr{G}} \prod_{x \in X} \sum_{y \in \mathscr{Y}_x} prob(y \mid \mathscr{G}),$$

where $\mathsf{m} = 0$ denotes a square null matrix of size equal to the length of the corresponding sequence, and $\mathfrak{U}_X = \{\mathscr{U}_x^{\mathsf{m}=0} : x \in X\}$.

Contrastive estimation Often it is reasonable to expect that $\mathscr{U}_{\text{query}}$ comes from a neighborhood of the target population $\mathscr{N}(\mathfrak{U}_{\text{target}}) \subset \mathscr{U}_*$. In such cases it is practical to perform *contrastive estimation* [Smith and Eisner, 2005], which aims at shifting the probability mass distributed by the

grammar from the neighborhood of the of sample $\mathscr{N}(\mathfrak{U}_{\text{sample}})$ to the sample itself $\mathfrak{U}_{\text{sample}}$, such that:

$$\mathscr{G}_{\text{CE}} = \arg\max_{\mathscr{G}} \prod_{\mathscr{U}_x \in \mathfrak{U}_{\text{sample}}} \frac{prob(\mathscr{U}_x \mid \mathscr{G})}{prob(\mathscr{N}(\mathscr{U}_x) \mid \mathscr{G})}.$$

Consider two interesting neighborhoods. First, assume that contact map m is known and shared in the entire target population and hence in the sample: $\mathfrak{U}_X^{\mathsf{m}} = \{\mathscr{U}_x^{\mathsf{m}} : x \in X\}$. This implies the same length n of all sequences. Then $\mathscr{U}_n^{\mathsf{m}}$ is a reasonable neighborhood of the target population, so

$$\mathscr{G}_{\text{CE}(\mathsf{m})} = \arg\max_{\mathscr{G}} \prod_{\mathscr{U}_x^{\mathsf{m}} \in \mathfrak{U}_X^{\mathsf{m}}} \frac{prob(\mathscr{U}_x^{\mathsf{m}} \mid \mathscr{G})}{prob(\mathscr{U}_n^{\mathsf{m}} \mid \mathscr{G})} = \arg\max_{\mathscr{G}} \frac{\prod_{x \in X} \sum_{y \in \mathscr{Y}_x^{\mathsf{m}}} prob(y \mid \mathscr{G})}{\left[\sum_{y \in \mathscr{Y}_n^{\mathsf{m}}} prob(y \mid \mathscr{G})\right]^{|X|}}.$$

Second, assume that sequence x is known to be yielded by the target population. Now, the goal is to maximize likelihood that the shapes of parse trees generated for sequences in the target population are consistent with contact maps. Then $\mathfrak{U}_X$ is a reasonable neighborhood of the sample $\mathfrak{U}_X^{\mathsf{M}}$, so

$$\mathscr{G}_{\text{CE}(X)} = \arg\max_{\mathscr{G}} \prod_{(x,\mathsf{m}) \in \mathscr{S}} \frac{prob(\mathscr{U}_x^{\mathsf{m}} \mid \mathscr{G})}{prob(\mathscr{U}_x \mid \mathscr{G})} = \arg\max_{\mathscr{G}} \prod_{(x,\mathsf{m}) \in \mathscr{S}} \frac{\sum_{y \in \mathscr{Y}_x^{\mathsf{m}}} prob(y \mid \mathscr{G})}{\sum_{y \in \mathscr{Y}_x} prob(y \mid \mathscr{G})}.$$

2.2 Simple(r) instance

2.2.1 Definitions

Let $\ddot{\mathscr{G}} = \langle \Sigma, V, v_0, R, \theta \rangle$ be a probabilistic context-free grammar such that $V = V_T \uplus V_N$, $R = R_a \uplus R_b \uplus R_c$, and

$$R_a = \{r_i : V_T \to \Sigma\},$$
$$R_b = \{r_j : V_N \to (V_N \cup V_T)\ (V_N \cup V_T)\},$$
$$R_c = \{r_k : V_N \to V_T\ V_N\ V_T\}.$$

Subsets R_a, R_b and R_c are referred to as *lexical*, *branching*, and *contact* rules, respectively. Joint subset $R_b \cup R_c$ is referred to as *structural* rules.

Let m be a contact matrix compatible with the context-free grammar, i.e. no pair of positions in contact overlaps nor crosses boundaries of other pairs in contact (though pairs can be nested one in another):

$$\forall (i,j)\ m_{i,j} = 1 \wedge (i \le k \le j \oplus i \le l \le j) \Rightarrow m_{k,l} = 0,$$

where $\oplus$ denotes the exclusive disjunction, and positions in contact are separated from each other by at least 2:

$$\forall (i,j)\ i < j + 2.$$

Let distance threshold in tree $\delta = 4$. Then a complete parse tree y generated by $\ddot{\mathscr{G}}$ is consistent with m only if for all $m_{i,j} = 1$ derivation

$$\alpha_{1,i-1}\ v_k\ \alpha_{j+1,n} \overset{*}{\Rightarrow} \alpha_{1,i-1}\ x_i\ v_l\ x_j\ \alpha_{j+1,n}$$

is performed with a string of production rules

$$[v_k \to v_t v_l v_u][v_t \to x_i][v_t \to x_j],$$

where $\alpha_{i,j} \in (\Sigma \cup V)^{j-i+1}$, $v_k, v_l \in V_N$ and $v_t, v_u \in V_T$.

2.2.2 Parsing

Given an input sequence x of length n and a grammar $\ddot{\mathscr{G}}$, $prob(x \mid \ddot{\mathscr{G}}) \equiv prob(\mathscr{Y}_x \mid \ddot{\mathscr{G}}) = \sum_{y \in \mathscr{Y}_x} prob(y \mid \ddot{\mathscr{G}})$ can be calculated in $O(n^3)$ by a slightly modified probabilistic Cocke-Kasami-Younger bottom-up chart parser [Cocke, 1969, Kasami, 1965, Younger, 1967]. Indeed, productions in $R_a \uplus R_b$ conforms to the Chomsky Normal Form [Chomsky, 1959], while it is easy to see that productions in R_c requires only $O(n^2)$. The algorithm computes $prob(x \mid \ddot{\mathscr{G}}) = prob(\mathscr{Y}_x \mid \ddot{\mathscr{G}})$ in chart table P of dimensions $n \times n \times |V|$, which effectively sums up probabilities of all possible parse trees $\mathscr{Y}_x$. In the first step, probabilities of assigning lexical non-terminals V_T for each terminal in the sequence x are stored in the bottom matrix $\mathsf{P}_1 = \mathsf{P}[1,:,:]$. Then, the table P is iteratively filled upwards with probabilities $\mathsf{P}[j,i,v] = prob(v \stackrel{*}{\Rightarrow} x_i \ldots x_{i+j-1} \mid v \in V, \ddot{\mathscr{G}})$. Finally, $prob(\mathscr{Y}_x^{\mathsf{m}} \mid \ddot{\mathscr{G}}) = \mathsf{P}[n,1,v_0]$.

New extended version of the algorithm (Fig. 1) computes $prob(\mathscr{Y}_x^{\mathsf{m}} \mid \ddot{\mathscr{G}})$, i.e. it considers only parse trees $\mathscr{Y}_x^{\mathsf{m}}$ which are consistent with m. To this goal it uses an additional table C of dimensions $\sum(\mathsf{m})/2 \times n \times |V_T|$. After completing P_1 (lines 10-12), probabilities of assigning lexical non-terminals V_T at positions involved in contacts are moved from P_1 to C (lines 13-21) such that each matrix $\mathsf{C}_p = \mathsf{C}[p,:,:]$ corresponds to p-th contact in m. In the subsequent steps C can only be used to complete productions in R_c; moreover both lexical non-terminals have to come either from P_1 or C, they can never be mixed (lines 35-40). The computational complexity of the extended algorithm is still $O(n^3)$ as processing of productions in R_c has to be multiplied by iterating over the number of contact pairs in m, which is $O(n)$ since the cross-serial dependencies are not allowed.

2.2.3 Calculating $prob(\mathscr{U}_n^{\mathsf{m}} \mid \ddot{\mathscr{G}})$

This section shows effective computing $prob(\mathscr{U}_n^{\mathsf{m}} \mid \ddot{\mathscr{G}})$, which is the denominator for the contrastive estimation of $\mathscr{G}_{\mathrm{CE}(\mathsf{m})}$ (cf. section 2.1.3). Given a sequence x of length n, a corresponding matrix m of size $n \times n$ and a grammar $\ddot{\mathscr{G}}$, the probability of the set of trees over any sequence of length n consistent with m is

$$prob(\mathscr{U}_n^{\mathsf{m}} \mid \ddot{\mathscr{G}}) \equiv \sum_{x \in \Sigma^n} prob(\mathscr{U}_x^{\mathsf{m}} \mid \ddot{\mathscr{G}}) = \sum_{x \in \Sigma^n} \sum_{y \in \mathscr{Y}_x^{\mathsf{m}}} prob(y \mid \ddot{\mathscr{G}}).$$

Given grammar $\ddot{\mathscr{G}}$, any complete derivation r is a composition $r = \dot{r} \circ \tilde{r}$, where $\dot{r} \in (R_a)^*$ and $\tilde{r} \in (R_b \cup R_c)^*$. Let y be the parse tree corresponding to derivation r, and let $\tilde{y}$ be an incomplete parse tree corresponding to derivation $\tilde{r}$. Note that for any y corresponding to $r = \dot{r} \circ \tilde{r}$ there exists one and only one $\tilde{y}$ corresponding to $\tilde{r}$. Let $\tilde{\mathscr{Y}}_x^{\mathsf{m}}$ denote the set of such incomplete trees $\tilde{y}$. Note that labels of the leaf nodes of $\tilde{y}$ are lexical non-terminals $\forall(i)\ \alpha_{i,i} \in V_T$, and that $\dot{r}$ represents the unique left-most derivation $yield(\tilde{y}) \stackrel{*}{\Rightarrow} x$. Thus,

$$\sum_{x \in \Sigma^n} \sum_{y \in \mathscr{Y}_x^{\mathsf{m}}} prob(y \mid \ddot{\mathscr{G}}) = \sum_{x \in \Sigma^n} \sum_{\tilde{y} \in \tilde{\mathscr{Y}}_x^{\mathsf{m}}} prob(\tilde{y} \mid \ddot{\mathscr{G}}) \cdot prob(yield(\tilde{y}) \stackrel{*}{\Rightarrow} x \mid \ddot{\mathscr{G}}).$$

Note that value of the expression will not change if the second summation is over $\tilde{y} \in \tilde{\mathscr{Y}}_n^{\mathsf{m}}$ since $\forall(\tilde{y} \notin \tilde{\mathscr{Y}}_x^{\mathsf{m}})\ prob(yield(\tilde{y}) \stackrel{*}{\Rightarrow} x \mid \ddot{\mathscr{G}}) = 0$. Combining with observation that $prob(\tilde{y} \mid \ddot{\mathscr{G}})$ does not depend on x, the expression can be therefore rewritten as:

$$\sum_{x \in \Sigma^n} \sum_{y \in \mathscr{Y}_x^{\mathsf{m}}} prob(y \mid \ddot{\mathscr{G}}) = \sum_{\tilde{y} \in \tilde{\mathscr{Y}}_n^{\mathsf{m}}} prob(\tilde{y} \mid \ddot{\mathscr{G}}) \cdot \sum_{x \in \Sigma^n} prob(yield(\tilde{y}) \stackrel{*}{\Rightarrow} x \mid \ddot{\mathscr{G}}).$$

```
01: function parse_cky_cm(x, m, Ra, Rb, Rc, Vt, Vn, v0)
02: # input:
03: # x - sequence, m - contact map
04: # Ra - lexical, Rb - branching, Rc - contact rules
05: # Vt - set of lexical, Vn - set of non-lexical non-terminals
06: # v0 - start symbol

07:     n = length(x)
08:     P[n, n, |Vn|+|Vt|] = 0.0
09:     C[sum(m)/2, n, |Vt|] = 0.0

10:     for i=1 to n
11:         for r in Ra
12:             if x[i]==r.rhs[1] P[1,i,r.lhs] = r.prob
13:     num_p=0
14:     for i=1 to n-2
15:         for j=i+2 to n
16:             if m[i,j]==1
17:                 for r in Ra
18:                     P[1,i,r.lhs] = P[1,j,r.lhs] = 0.0
19:                     if x[i]==r.rhs[1] C[p,i,r.lhs] = r.prob
20:                     if x[j]==r.rhs[1] C[p,j,r.lhs] = r.prob
21:                 num_p=num_p+1
22:     for j=2 to n
23:         for i=1 to n-j+1
24:             for k=1 to j-1
25:                 for r in Rb
26:                     P[j,i,r.lhs] += r.prob
27:                                 * P[  k,i,  r.rhs[1]]
28:                                 * P[j-k,i+k,r.rhs[2]]
29:             if (j>=3)
30:                 for r in Rc
31:                     P[j,i,r.lhs] += r.prob
32:                                 * P[1,  i,  r.rhs[1]]
33:                                 * P[j-2,i+1,r.rhs[2]]
34:                                 * P[1,  i+j,r.rhs[3]]
35:                 for c=0 to num_p-1
36:                     for r in Rc
37:                         P[j,i,r.lhs] += r.prob
38:                                     * C[p,  i,  r.rhs[1]]
39:                                     * P[j-2,i+1,r.rhs[2]]
40:                                     * C[p,  i+j,r.rhs[3]]
41:     return P[n, 1, v0]
```

Figure 1: Pseudocode of the modified CKY parser

However, if $\ddot{\mathscr{G}}$ is *proper*, then $\forall(\tilde{y} \in \tilde{\mathscr{Y}}_n^{\mathsf{m}})\ \sum_{x\in\Sigma^n} prob(yield(\tilde{y}) \stackrel{*}{\Rightarrow} x \mid \ddot{\mathscr{G}}) = 1$, as:

$$\sum_{x\in\Sigma^n} prob(yield(\tilde{y}) \stackrel{*}{\Rightarrow} x \mid \ddot{\mathscr{G}}) = \sum_{x\in\Sigma^n} \prod_{i=1}^{n} \theta(\alpha_{i,i} \to x_i) =$$

$$\sum_{x\in\Sigma^n} \theta(\alpha_{1,1} \to x_1) \cdot \ldots \cdot \theta(\alpha_{n,n} \to x_n) =$$

$$\begin{aligned}
&\theta(\alpha_{1,1} \to a_1) \cdot \theta(\alpha_{2,2} \to a_1) \cdot \ldots \cdot \theta(\alpha_{n-1,n-1} \to a_1) \cdot \theta(\alpha_{n,n} \to a_1) + \\
&\theta(\alpha_{1,1} \to a_1) \cdot \theta(\alpha_{2,2} \to a_1) \cdot \ldots \cdot \theta(\alpha_{n-1,n-1} \to a_1) \cdot \theta(\alpha_{n,n} \to a_2) + \\
&\vdots \\
&\theta(\alpha_{1,1} \to a_{|\Sigma|}) \cdot \theta(\alpha_{2,2} \to a_{|\Sigma|}) \cdot \ldots \cdot \theta(\alpha_{n-1,n-1} \to a_{|\Sigma|}) \cdot \theta(\alpha_{n,n} \to a_{|\Sigma|}) =
\end{aligned}$$

$$\begin{pmatrix}
\theta(\alpha_{1,1} \to a_1) \cdot \theta(\alpha_{2,2} \to a_1) \cdot \ldots \cdot \theta(\alpha_{n-1,n-1} \to a_1) + \\
\theta(\alpha_{1,1} \to a_1) \cdot \theta(\alpha_{2,2} \to a_1) \cdot \ldots \cdot \theta(\alpha_{n-1,n-1} \to a_2) + \\
\vdots \\
\theta(\alpha_{1,1} \to a_{|\Sigma|}) \cdot \theta(\alpha_{2,2} \to a_{|\Sigma|}) \cdot \ldots \cdot \theta(\alpha_{n-1,n-1} \to a_{|\Sigma|})
\end{pmatrix} \cdot \sum_{s=1}^{|\Sigma|} \theta(\alpha_{n,n} \to a_s),$$

where $a_s \in \Sigma$. Since $\ddot{\mathscr{G}}$ is *proper* then $\forall(v \in V_T)\ \sum_{s=1}^{|\Sigma|} \theta(v \to a_s) = 1$ and therefore the entire formula evaluates to 1, which can be easily shown by iterative regrouping. This leads to the final formula:

$$prob(\mathscr{U}_n^{\mathsf{m}} \mid \ddot{\mathscr{G}}) = \sum_{\tilde{y}\in\tilde{\mathscr{Y}}_n^{\mathsf{m}}} prob(\tilde{y} \mid \ddot{\mathscr{G}}).$$

Technically, $\sum_{\tilde{y}\in\tilde{\mathscr{Y}}_n^{\mathsf{m}}} prob(\tilde{y} \mid \ddot{\mathscr{G}})$ can be readily calculated by the bottom-up chart parser by setting $\forall(r_k \in R_a)\ \theta(r_k) = 1$.

2.3 Evaluation

2.3.1 Learning

The present PCFG-CM approach was evaluated in practice for grammatical models $\ddot{G}$ and $\bar{G} = \ddot{G} \setminus R_c$ (the same grammar but without the contact rules) using an on-site framework for learning the probabilities of rules [Dyrka and Nebel, 2009, Dyrka et al., 2013]. For a given underlying CFG $\ddot{G}$, the framework estimates probabilities θ of the corresponding PCFG $\ddot{\mathscr{G}} = \langle \ddot{G}, \theta \rangle$ from the positive sample using a genetic algorithm in the Pittsburgh flavor, where each individual represents a whole grammar. Unlike previous applications of the framework in which probabilities of the lexical rules were fixed according to representative physicochemical properties of amino acids, in this research probabilities of all rules were subject to evolution. The objective functions were implemented for estimators $\ddot{\mathscr{G}}_{\mathrm{ML}}$, $\ddot{\mathscr{G}}_{\mathrm{CE}(X)}$, and $\ddot{\mathscr{G}}_{\mathrm{CE}(\mathsf{m})}$. Besides, the setup of the genetic algorithm closely followed that of [Dyrka and Nebel, 2009].

The input non-probabilistic grammar $\ddot{G}$ consisted of an alphabet of twenty terminal symbols representing amino acid species

$$\Sigma = \{A, C, D, E, F, G, H, I, K, L, M, N, Q, P, R, S, T, V, W, Y\},$$

a set of non-terminals symbols $V = V_T \uplus V_N$, where $V_T = \{l_1, l_2, l_3\}$ and $V_N = \{v_0, v_1, v_2, v_3\}$, and a set of rules $R = R_a \uplus R_b \uplus R_c$, which consisted of all possible allowed combinations of symbols, hence $|R_a| = 60, |R_b| = 196, |R_c| = 144$. The set of non-contact rules was identical to the standard grammar in [Dyrka and Nebel, 2009]. The number of non-terminal symbols was limited to a few in order to keep the number of parameters to be optimized by the genetic algorithm reasonably small. Combinations of symbols in rules were not constrained beyond general definition of the model $\ddot{G}$ in order to avoid interference with the contact-map constraints, for the sake of transparent evaluation of the PCFG-CM.

2.3.2 Performance measures

Performance of grammars was evaluated using a variant of the k-fold Cross-Validation scheme in which $k-2$ parts are used for training, 1 part is used for validation and parameter selection, and 1 part is used for final testing and reporting results. The negative set was not used in the training phase.

In order to avoid composition bias, protein sequences in the test sample were scored against the null model (a unigram), which assumed global average frequencies of amino acids, no contact information, and the sequence length of the query protein. The amino acid frequencies were obtained using the online ProtScale tool for the UniProtKB/Swiss-Prot database [Gasteiger et al., 2005]).

Discriminative performance Grammars were assessed on the basis of the average precision (AP) in the recall-precision curve (RPC). The advantage of RPC over the more common Receiver Operating Characteristic (ROC) is robustness to unbalanced samples where negative data is much more numerous than positive data [Davis and Goadrich, 2006]. AP approximates the area under RPC.

Descriptive performance Intuitively, a decent explanatory grammar generates parse trees consistent with the spatial structure of the analyzed protein. The most straightforward approach to assess descriptive performance is to use the UST of the most likely parse tree as a predictor of spatial contacts between positions in the protein sequence, parameterized by the cutoff δ on path length between the leaves. The natural threshold for grammar $\ddot{G}$, which is $\delta = 4$ (the shortest distance between terminals generated by R_b rules), was used for calculating the precision of contact prediction. In addition, AP of the RPC, which sums up over all possible cutoffs, was computed to allow comparison with grammars without pairing rules. Eventually, the recall of contact prediction for the threshold $\delta = 4$, measured with regard to the partial contact map used in the training, was used to assess the learning process.

Implementation The PCFG-CM parser and the Protein Grammar Evolution framework were implemented in C++ using GAlib [Wall, 2005] and Eigen [Guennebaud et al., 2010]. Performance measures were implemented in Python 2 [van Rossum and de Boer, 1991] using Biopython [Cock et al., 2009], igraph [Csardi and Nepusz, 2006], NumPy [van der Walt et al., 2011], pyparsing [McGuire, 2008], scikit-learn [Pedregosa et al., 2011] and SciPy [Jones et al., 2001].

3 Results

3.1 Materials

Probabilistic grammars were estimated for three samples of protein fragments related to functionally relevant gapless motifs [Sigrist et al., 2002, Bailey and Elkan, 1994]. Within each sample, all sequences shared the same length, which avoided sequence length effects on grammar scores (this could be resolved by an appropriate null model). For each sample, one experimentally solved spatial structure in the Protein Data Bank (PDB) [Berman et al., 2000] was selected as a representative. Three samples included amino acid sequence of two small ligand binding sites (already analyzed in [Dyrka and Nebel, 2009]) and a functional amyloid (Table 1):

- *CaMn*: a Calcium and Manganese binding site found in the legume lectins [Sharon and Lis, 1990]. Sequences were collected according to the PROSITE PS00307 pattern [Sigrist et al., 2013] true positive and false negative hits. Original boundaries of the pattern were extended to cover the entire binding site, similarly to [Dyrka and Nebel, 2009]. The motif folds into a stem-like structure with multiple contacts, many of them forming nested dependencies, which stabilize anti-parallel beta-sheet made of two ends of the motif (Fig. 2a based on pdb:2zbj [de Oliveira et al., 2008]);

- *NAP*: the Nicotinamide Adenine dinucleotide Phosphate binding site fragment found in an aldo/keto reductase family [Bohren et al., 1989]. Sequences were collected according to the PS00063 pattern true positive and false negative hits (four least consistent sequences were excluded). The motif is only a part of the binding site of the relatively large ligand. Intra-motif contacts seem to be insufficient for defining the fold, which depends also on interactions with amino acids outside the motif (Fig. 2b based on pdb:1mrq [Couture et al., 2003]);

- *HET-s*: the HET-s-related motifs r1 and r2 involved in the prion-like signal transduction in fungi identified in a recent study [Daskalov et al., 2015]. The largest subset of motif sequences with length of 21 amino acids was used to avoid length effects on grammar scores. When interacting with a related motif r0 from a cooperating protein, motifs r1 and r2 adopt the beta-hairpin-like folds which stack together. While stacking of multiple motifs from several proteins is essential for stability of the structure, interactions between hydrophobic amino acids within a single hairpin are also important. In addition, correlation analysis revealed strong dependency between positions 17 and 21 [Daskalov et al., 2015] (Fig. 2c based on [van Melckebeke et al., 2010]).

Diversity of sequences ranged from the most homogeneous CaMn to the most diverse HET-s, which consisted of 5 subfamilies [Daskalov et al., 2015].

Negative samples were designed to roughly approximate the entire space of protein sequences. They were based on the negative set from [Dyrka and Nebel, 2009], which consisted of 829 single chain sequences of 300-500 residues retrieved from the Protein Data Bank [Berman et al., 2000] at identity of 30% (accessed on 12th December 2006). For each positive sample, the corresponding negative sample was obtained by cutting the basic negative set into subsequences of the length of positive sequences.

Table 1: Datasets. Notations: *sim* - maximum sequence similarity, *npos/nneg* - number of positive/negative sequences, *len* - sequence length in amino acids, *ncon* - total number of non-local contacts (sequence separation 3+), *msiz* - number of contacts selected for training

id	type	sim	npos	nneg	len	pdb	ncon	msiz
CaMn	binding-site	71%	24	28560	27	2zbj	41	6
NAP	binding-site	70%	64	47736	16	1mrq	11	2
HET-s	amyloid	70%	160	33248	21	2kj3	10	3

All samples were made non-redundant at level of sequence similarity around 70%. Contact pairings were assigned manually and collectively to all sequences in the set based on a selected representative spatial structure in the PDB database (Fig. 2).

3.2 Performance

Probabilistic grammars with the contact rules $\ddot{\mathscr{G}}$ were learned through estimation of probabilities of rules θ for non-probabilistic CFG $\ddot{G}$ using training samples made of sequences coupled with a contact map $\mathfrak{U}_X^{\mathsf{m}}$, or using sequences alone $\mathfrak{U}_X$. Probabilistic grammars without the contact rules $\bar{\mathscr{G}}$ were learned using sequences alone $\mathfrak{U}_X$, since these grammars cannot generate parse trees consistent with contact maps for the distance threshold $\delta = 4$. Note that since there is the one-to-one correspondence between input sample set $S = [X\mathsf{M}]$ and sample of UST sets $\mathfrak{U}_X^{\mathsf{M}}$, notations developed for the sets of USTs are used to denote the input samples.

3.2.1 Discriminative power

For evaluation of the discriminative power of the PCFG-CM approach, the rule probabilities were estimated using the maximum-likelihood estimator (denoted ML) and the contrastive estimator with regard to a given contact map (denoted CE(m)). The discriminative performance of the resulting probabilistic grammars for test data made of sequences alone $\mathfrak{U}_X$ and sequences coupled with a contact map $\mathfrak{U}_X^{\mathsf{m}}$ is presented in Table 2 in terms of the average precision (AP).

The baseline is the average precision of grammars estimated without contact constraints, $\bar{\mathscr{G}}_{\mathrm{ML}}^{\mathsf{m}=0}$ and $\ddot{\mathscr{G}}_{\mathrm{ML}}^{\mathsf{m}=0}$, tested on sequences alone $\mathfrak{U}_X$, which ranged from 0.43-0.46 for HET-s to 0.94-0.96 for CaMn. The scores show negative correlation with diversity of the samples and limited effect of adding contact rules (though the latter may result from more difficult learning of increased number of parameters with added rules). Grammars with the contact rules estimated without a contact map $\ddot{\mathscr{G}}_{\mathrm{ML}}^{\mathsf{m}=0}$ performed much worse when tested on the samples coupled with a contact map $\mathfrak{U}_X^{\mathsf{m}}$. This indicated that, in general, parses consistent with the constraints were not preferred by default when grammars were trained on sequences alone.

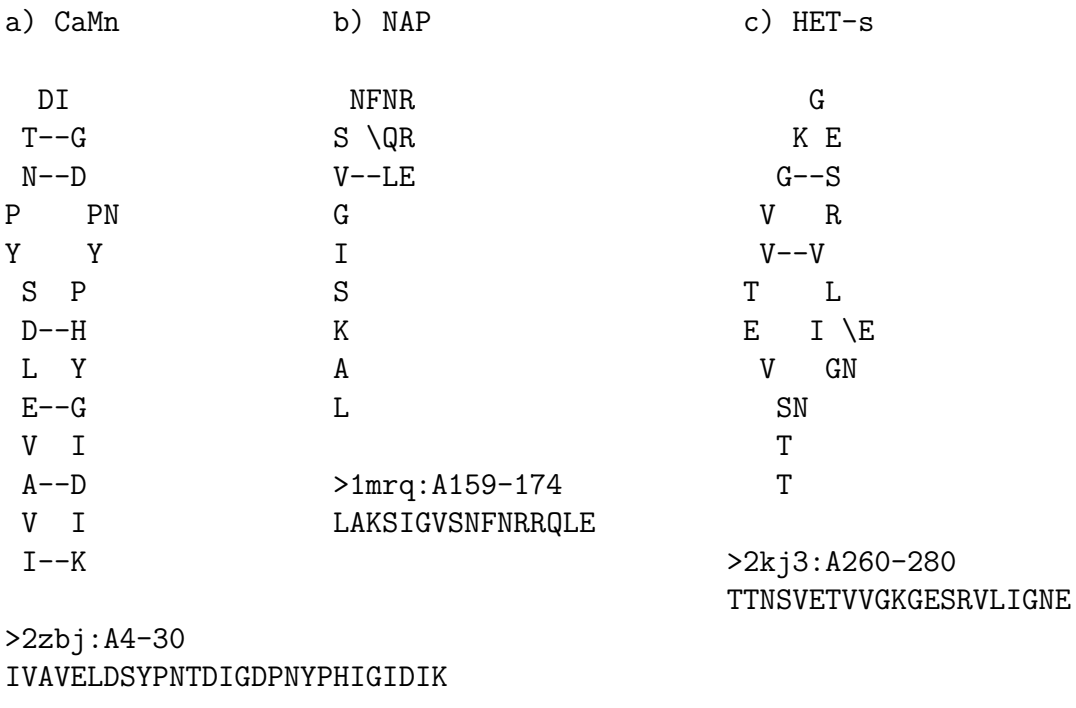

Figure 2: Schematic representation of structures of the sample motifs. Context-free-compatible contact pairings selected in this study are marked with dashes and slashes. Order of amino acids in sequence and their coordinates in the structure are given below each diagram. Notes: 1) in CaMn, only 4 out of 7 real hydrogen bond-related contacts in the stem-like part were included in the contact map for the sake of simplicity; 2) in HET-s, for example, the pair of V5 and I18 conforms to definition of contact, however it crosses another contact between L17 and E21.

Table 2: Discriminative performance of grammars in terms of AP

Grammar	$\bar{\mathscr{G}}_{\mathrm{ML}}^{\mathsf{m}=0}$	$\ddot{\mathscr{G}}_{\mathrm{ML}}^{\mathsf{m}=0}$		$\ddot{\mathscr{G}}_{\mathrm{ML}}$		$\ddot{\mathscr{G}}_{\mathrm{CE}(\mathsf{m})}$	
Test sample	$\mathfrak{U}_X$	$\mathfrak{U}_X$	$\mathfrak{U}_X^{\mathsf{m}}$	$\mathfrak{U}_X$	$\mathfrak{U}_X^{\mathsf{m}}$	$\mathfrak{U}_X$	$\mathfrak{U}_X^{\mathsf{m}}$
CaMn	0.94	0.96	0.67	0.95	0.95	0.79	0.98
NAP	0.78	0.86	0.28	0.75	0.79	0.24	0.91
HET-s	0.46	0.43	0.24	0.60	0.81	0.23	0.94

For all three samples, the highest AP (0.91-0.98) achieved grammars obtained using the contrastive estimation with regard to a contact map $\ddot{\mathscr{G}}_{\mathrm{CE}(\mathsf{m})}$ tested on the samples with the same map $\mathfrak{U}_X^{\mathsf{m}}$. The improvement relative to the baseline was most pronounced for HET-s, yet still statistically significant ($p < 0.05$) for NAP. As expected, the contrastively estimated grammars performed poorly on sequences alone $\mathfrak{U}_X$ except for the CaMn sample.

The maximum-likelihood grammars estimated with a contact map $\ddot{\mathscr{G}}_{\mathrm{ML}}$ tested on sequences coupled with the same map $\mathfrak{U}_X^{\mathsf{m}}$ performed worse than the contrastively estimated grammars but comparably or significantly better (HET-s) than the baseline. The average precision of these grammars was consistently lower when tested on sequences alone $\mathfrak{U}_X$, yet still considerable (from 0.60 for HET-s to 0.95 for CaMn). It is notable that in the HET-s case, the maximum-likelihood grammars estimated with a contact map $\ddot{\mathscr{G}}_{\mathrm{ML}}$ achieved better AP on sequences alone $\mathfrak{U}_X$ than the maximum-likelihood grammars estimated without a contact map $\ddot{\mathscr{G}}_{\mathrm{ML}}^{\mathsf{m}=0}$.

Universally high AP for CaMn can be contributed to the relatively strong pairing signal from the long stem-like part of the motif particularly suitable for modeling with the contact rules.

3.2.2 Descriptive power

For evaluation of the descriptive power of the PCFG-CM approach, the rule probabilities were estimated using the maximum-likelihood estimator (denoted ML) and the contrastive estimator with regard to the sequence set (denoted CE(X)). Descriptive value of the most probable parse trees generated using the resulting probabilistic grammars for test sequences without contact information $\mathfrak{U}_X$ is presented in Table 3. Efficiency of the learning was measured on the basis of the recall at the distance threshold $\delta = 4$ with regard to the context-free compatible contact map m used in the training. Consistency of the most likely parse tree with the protein structure was measured on the basis of the precision of contact prediction at the distance threshold $\delta = 4$ with regard to all contacts in the reference spatial structure with separation in sequence of at least 3. Both measures are not suitable for assessing grammars without contact rules $\bar{\mathscr{G}}$. Therefore, average precision over all thresholds δ was used as a complementary measure of consistency of the most likely trees with the protein structure. Note that the AP scores achievable for a context-free parse tree are reduced by overlapping of pairings.

The baseline is the result for grammars with the contact rules estimated without contact constraints $\ddot{\mathscr{G}}_{\mathrm{ML}}^{\mathsf{m}=0}$. The most likely parse trees generated using these grammars conveyed practically no information about contacts for NAP and HET-s (recall w.r.t. contact map m close to zero) and limited information about contacts for CaMn (moderate recall of 0.45). Increase of the recall to 0.79-0.98 obtained for the most likely parse trees generated using grammars $\ddot{\mathscr{G}}_{\mathrm{ML}}$ and $\ddot{\mathscr{G}}_{\mathrm{CE}(X)}$ testifies efficiency of the learning process with the contact constraints.

Importantly, consistency of the most likely parse trees with the protein structure measured by the precision followed a similar pattern and increased from 0.13 for HET-s, 0.14 for NAP, and 0.69 for CaMn when grammars with the contact rules were estimated without a contact map ($\ddot{\mathscr{G}}_{\mathrm{ML}}^{\mathsf{m}=0}$), to 0.52-0.57, 0.64, and 0.84-0.87, respectively, when grammars were estimated with a contact map ($\ddot{\mathscr{G}}_{\mathrm{ML}}$ and $\ddot{\mathscr{G}}_{\mathrm{CE}(X)}$). Accordingly, evaluation in terms of the average precision over distance thresholds indicated that distances in the most likely parse trees better reflected the protein structure if grammars were trained with the contact constraints.

Table 3: Descriptive quality of the most likely parse trees derived from sequences alone, in terms of recall at the distance threshold $\delta = 4$ w.r.t. the training contact map m, and precision at $\delta = 4$ (and AP over thresholds δ) w.r.t. the full contact map of the reference *pdb* structure for sequence separation 3+. Note that the shortest length of any path between leaves in the most likely parse trees of grammars without the contact rules $\bar{G}$ equals 5, which makes measures using $\delta = 4$ unutile.

Gram.	$\bar{\mathscr{G}}_{\mathrm{ML}}^{\mathsf{m}=0}$	$\ddot{\mathscr{G}}_{\mathrm{ML}}^{\mathsf{m}=0}$		$\ddot{\mathscr{G}}_{\mathrm{ML}}$		$\ddot{\mathscr{G}}_{\mathrm{CE}(X)}$	
Ref.	pdb	m	pdb	m	pdb	m	pdb
CaMn	(0.24)	0.45	0.69 (0.53)	0.92	0.87 (0.66)	0.98	0.84 (0.66)
NAP	(0.16)	0.00	0.14 (0.12)	0.96	0.64 (0.29)	0.96	0.64 (0.29)
HET-s	(0.08)	0.02	0.13 (0.14)	0.79	0.52 (0.24)	0.97	0.57 (0.27)

4 Discussion

The most effective way of training descriptors for a given sample was the contrastive estimation with reference to the contact map. This approach is only possible when a single contact map that fits all sequences in the target population can be used with the trained grammar. The maximum-likelihood estimators were effective when contacts were relevant to structure of the sequence (HET-s, CaMn). This is expected, as use of the contact rules is likely to be optimal for deriving a pair of amino acids in contact if they are actually correlated. Interestingly, in the case of HET-s, the maximum-likelihood grammar trained with the contact constraints $\ddot{\mathscr{G}}_{\mathrm{ML}}$ compared favorably with the maximum-likelihood grammar trained without the constraints $\ddot{\mathscr{G}}_{\mathrm{ML}}^{\mathsf{m}=0}$ even when tested on sequences alone (AP 0.60 versus 0.43). This indicates that if contacts are relevant for the structure of sequence, the PCFG-CM approach can improve robustness of learning to local optima.

The most likely parse trees, derived for inputs defined only by sequences, reproduced a vast majority of contacts (recall of at least 0.79 at $\delta = 4$) enforced by the contact-constrained training input. Moreover, precision of contact prediction at $\delta = 4$ and sequence separation 3+ was above 0.50, up to 0.87. This translated to the overall overlap with the full contact maps in the range of 0.27-0.39 (not shown). Note that only a fraction of contacts can be represented in the parse tree of context-free grammar, and not even all of them were enforced in training. The benefit of the contrastive estimation with reference to the sequence set was limited in comparison to the maximum-likelihood grammars. However, it should be noted that the shape of the most likely parse tree, which was used in the evaluation, does not necessarily reflect the most likely shape of parse tree. Unfortunately, the latter cannot be efficiently computed [Dowell and Eddy, 2004].

5 Conclusions

Complex character of non-local interactions between amino acids makes learning languages of protein sequences challenging. In this work we proposed a solution consisting on using structural information to constrain the syntactic trees, a technique which proved effective in learning probabilistic natural and RNA languages. We established a framework for learning probabilistic context-free grammars for protein sequences from syntactic trees partially constrained using contacts between amino acids. Within the framework, we implemented the maximum-likelihood and contrastive estimators for the rule probabilities of relatively simple yet practical covering grammars. Computational validation showed that additional knowledge present in the partial contact maps can be effectively incorporated into the probabilistic grammatical framework through the concept of syntactic tree consistent with the contact map. Grammars estimated with the contact constraints maintained good precision when used as classifiers, and derived the most likely parse trees displaying improved fidelity to protein structures compared to the baseline grammars estimated without the constraints.

The computational experiments mainly served assessing intuitions which led to development of the PCFG-CM approach. The full-scale practical application to bioinformatic problems such as sequence search would certainly require several enhancements. For example, accurate accounting for various sequence lengths would likely require a more elaborated null model. Moreover, to increase the number of non-terminal symbols, the learning framework has to be improved. This includes more efficient estimation of probabilities of numerous rules and/or added capability of inferring rules during learning [Unold, 2005, 2012, Coste et al., 2012, 2014]. A preliminary testing suggests that scoring inputs with the product of probabilities obtained using grammars with the lexical rule probabilities fixed according to representative physicochemical properties of amino acids [Dyrka and Nebel, 2009], and the appropriately adjusted null model, results in more discriminative power compared to the current approach (not shown). An extension of the PCFG-CM framework to account for uncertain contact information [Knudsen, 2005] should be feasible through introducing the concept of the fuzzy sets of syntactic trees. These application-related developments are left for future work.

Though tested in the learning setting consisting in optimizing only rule probabilities, the estimators defined in the present PCFG-CM framework can be used in more general learning schemes inferring also the grammar structure. Indeed, such schemes may even more benefit from constraining the larger search space. It is also interesting to consider extending the framework beyond context-free grammars as contacts in proteins are often overlapping and thus context-sensitive. In this case, however, the one-to-one correspondence between the parse tree and the derivation breaks, therefore it may be advisable to redefine the grammatical counterpart of the spatial distance in terms of derivation steps in order to take advantage from higher expressiveness.

Acknowledgements WD acknowledges Olgierd Unold and Mateusz Pyzik for interesting discussions in the course of the project.

References

T. L. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pages 28–36. AAAI Press, Menlo Park, California, 1994.

J. Baker. Trainable grammars for speech recognition. In D.Klatt and J. Wolf, editors, *Speech Communication Papers for the 97th Meeting of the Acoustical Society of America*, pages 547–550, 1979.

C. Baldassi, M. Zamparo, C. Feinauer, A. Procaccini, R. Zecchina, M. Weigt, and A. Pagnani. Fast and accurate multivariate gaussian modeling of protein families: predicting residue contacts and protein-interaction partners. *PloS one*, 9(3):e92721, 2014.

H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. T. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The Protein Data Bank. *Nucleic Acid Research*, 28:235–242, 2000.

K. M. Bohren, B. Bullock, B. Wermuth, and K. H. Gabbay. The aldo-keto reductase superfamily. cdnas and deduced amino acid sequences of human aldehyde and aldose reductases. *Journal of Biological Chemistry*, 264(16):9547–51, 1989.

T. L. Booth. Probabilistic representation of formal languages. In *10th Annual Symposium on Switching and Automata Theory (swat 1969)*, pages 74–81, Oct 1969.

V. Brendel and H. Busse. Genome structure described by formal languages. *Nucleic Acid Research*, 12:2561–2568, 1984.

A. Bretaudeau, F. Coste, F. Humily, L. Garczarek, G. Le Corguillé, C. Six, M. Ratin, O. Collin, W. M. Schluchter, and F. Partensky. CyanoLyase: a database of phycobilin lyase sequences, motifs and functions. *Nucleic Acids Research*, page 6, Nov. 2012.

R. C. Carrasco, J. Oncina, and J. Calera-Rubio. Stochastic inference of regular tree languages. *Machine Learning*, 44(1):185–197, Jul 2001. ISSN 1573-0565.

G. Carroll and E. Charniak. Two experiments on learning probabilistic dependency grammars from corpora. In *The Workshop on Statistically-Based Natural Language Programming Techniques*, pages 1–13. AAAI, 1992.

E. Charniak. Tree-bank grammars. Technical Report CS–96–02, Brown University, Department of Computer Science, 1996.

N. Chomsky. On certain formal properties of grammars. *Information and Control*, 2(2):137 – 167, 1959. ISSN 0019-9958.

P. J. A. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski, and M. J. L. de Hoon. Biopython: freely available python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25(11):1422–1423, 2009.

J. Cocke. *Programming languages and their compilers: Preliminary notes*. Courant Institute of Mathematical Sciences, New York University, 1969. ISBN B0007F4UOA.

S. B. Cohen, K. Stratos, M. Collins, D. P. Foster, and L. Ungar. Spectral learning of latent-variable pcfgs: Algorithms and sample complexity. *Journal of Machine Learning Research*, 15:2399–2449, 2014.

F. Coste. *Learning the Language of Biological Sequences*, pages 215–247. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016. ISBN 978-3-662-48395-4.

F. Coste and G. Kerbellec. Learning Automata on Protein Sequences. In A. Denise, P. Durrens, S. Robin, E. Rocha, A. de Daruvar, and A. Groppi, editors, *JOBIM*, pages 199–210, Bordeaux, France, July 2006.

F. Coste, G. Garet, and J. Nicolas. Local Substitutability for Sequence Generalization. In J. Heinz, C. de la Higuera, and T. Oates, editors, *ICGI 2012*, volume 21 of *JMLR Workshop and Conference Proceedings*, pages 97–111, Washington, United States, Sept. 2012. University of Maryland, MIT Press.

F. Coste, G. Garet, and J. Nicolas. A bottom-up efficient algorithm learning substitutable languages from positive examples. In A. Clark, M. Kanazawa, and R. Yoshinaka, editors, *ICGI (International Conference on Grammatical Inference)*, volume 34 of *Proceedings of Machine Learning Research*, pages 49–63, Kyoto, Japan, Sept. 2014.

J.-F. Couture, P. Legrand, L. Cantin, V. Luu-The, F. Labrie, and R. Breton. Human 20hydroxysteroid dehydrogenase: Crystallographic and site-directed mutagenesis studies lead to the identification of an alternative binding site for c21-steroids. *Journal of Molecular Biology*, 331(3):593 – 604, 2003. ISSN 0022-2836.

G. Csardi and T. Nepusz. The igraph software package for complex network research. *InterJournal*, Complex Systems:1695, 2006.

A. Daskalov, W. Dyrka, and S. J. Saupe. Theme and variations: evolutionary diversification of the het-s functional amyloid motif. *Scientific Reports*, 5:12494, 01 2015.

J. Davis and M. Goadrich. The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd International Conference on Machine Learning*, 2006.

T. de Oliveira, P. Delatorre, B. da Rocha, E. de Souza, K. Nascimento, G. Bezerra, T. R. Moura, R. Benevides, E. Bezerra, F. Moreno, V. Freire, W. de Azevedo, and B. Cavada. Crystal structure of dioclea rostrata lectin: Insights into understanding the ph-dependent dimer-tetramer equilibrium and the structural basis for carbohydrate recognition in diocleinae lectins. *Journal of Structural Biology*, 164(2):177 – 182, 2008. ISSN 1047-8477.

R. D. Dowell and S. R. Eddy. Evaluation of several lightweight stochastic context-free grammars for rna secondary structure prediction. *BMC Bioinformatics*, 5(1):71, Jun 2004. ISSN 1471-2105.

W. Dyrka. Probabilistic context-free grammar for pattern detection in protein sequences. Master's thesis, Faculty of Computing, Information Systems and Mathematics, Kingston University, London, 2007.

W. Dyrka and J.-C. Nebel. A stochastic context free grammar based framework for analysis of protein sequences. *BMC Bioinformatics*, 10:323, 2009.

W. Dyrka, J. Nebel, and M. Kotulska. Probabilistic grammatical model for helixhelix contact site classification. *Algorithms for Molecular Biology*, 8(1):31, Dec 2013. ISSN 1748-7188.

S. R. Eddy. Profile hidden markov models. *Bioinformatics*, 14(9):755–763, 1998.

S. R. Eddy. Accelerated profile hmm searches. *PLoS Computational Biology*, 7(10):e1002195, 10 2011.

S. R. Eddy and R. Durbin. Rna sequence analysis using covariance models. *Nucleic Acids Research*, 22(11):2079–2088, 1994.

M. Ekeberg, C. Lövkvist, Y. Lan, M. Weigt, and E. Aurell. Improved contact prediction in proteins: using pseudolikelihoods to infer potts models. *Physical Review E*, 87(1):012707, 2013.

R. D. Finn, P. Coggill, R. Y. Eberhardt, S. R. Eddy, J. Mistry, A. L. Mitchell, S. C. Potter, M. Punta, M. Qureshi, A. Sangrador-Vegas, G. A. Salazar, J. Tate, and A. Bateman. The pfam protein families database: towards a more sustainable future. *Nucleic Acids Research*, 2015.

E. Gasteiger, C. Hoogland, A. Gattiker, S. Duvaud, M. Wilkins, R. Appel, and A. Bairoch. Protein identification and analysis tools on the expasy server. In J. M. Walker, editor, *The Proteomics Protocols Handbook*, pages 571–607. Humana Press, 2005.

G. Guennebaud, B. Jacob, et al. Eigen v3. http://eigen.tuxfamily.org, 2010.

M. Jimenez-Montano. On the syntactic structure of protein sequences and the concept of grammar complexity. *Bull Math Biol*, 46:641–659, 1984.

D. Jones, D. Buchan, D. Cozzetto, and M. Pontil. PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics*, 28:184–190, 2012.

E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python. www.scipy.org, 2001.

H. Kamisetty, S. Ovchinnikov, and D. Baker. Assessing the utility of coevolution-based residue–residue contact predictions in a sequence-and structure-rich era. *Proceedings of the National Academy of Sciences*, 110(39):15674–15679, 2013.

T. E. Kammeyer and R. K. Belew. Stochastic context-free grammar induction with a genetic algorithm using local search. In *In: Foundations of Genetic Algorithms IV (R.K. Belew*, pages 3–5. Morgan Kaufmann, 1996.

T. Kasami. An efficient recognition and syntax analysis algorithm for context-free languages. Technical Report AFCRL-65-758, Air Force Cambridge Research Laboratory, Bedford, MA, 1965.

B. Keller and R. Lutz. Learning scfgs from corpora by a genetic algorithm. In *Artificial Neural Nets and Genetic Algorithms*, pages 210–214, Vienna, 1998. Springer Vienna. ISBN 978-3-7091-6492-1.

B. Keller and R. Lutz. Evolutionary induction of stochastic context free grammars. *Pattern Recognition*, 38(9):1393 – 1406, 2005. ISSN 0031-3203. Grammatical Inference.

B. Knudsen and J. Hein. Rna secondary structure prediction using stochastic context-free grammars and evolutionary history. *Bioinformatics*, 15:446–54, 1999.

M. Knudsen. Stochastic context-free grammars and rna secondary structure prediction. Master's thesis, Aarhus University, Denmark, 2005.

K. Lari and S. Young. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech & Language*, 4(1):35 – 56, 1990. ISSN 0885-2308.

P. McGuire. Pyparsing. http://pyparsing.wikispaces.com, 2008.

F. Morcos, A. Pagnani, B. Lunt, A. Bertolino, D. S. Marks, C. Sander, R. Zecchina, J. N. Onuchic, T. Hwa, and M. Weigt. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences*, 108(49):E1293–E1301, 2011.

Z. Pawlak. *Gramatyka i matematyka*. PWN, Warsaw, Poland, 1965.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

F. Pereira and Y. Schabes. Inside-outside reestimation from partially bracketed corpora. In *Proceedings of the 30th Annual Meeting on Association for Computational Linguistics*, ACL '92, pages 128–135, Stroudsburg, PA, USA, 1992. Association for Computational Linguistics.

M. Remmert, A. Biegert, A. Hauser, and J. Soeding. Hhblits: lightning-fast iterative protein sequence searching by hmm-hmm alignment. *Nature Methods*, 9(2):173–175, 2012.

Y. Sakakibara. Efficient learning of context-free grammars from positive structural examples. *Information and Computation*, 97(1):23 – 60, 1992. ISSN 0890-5401.

Y. Sakakibara, M. Brown, R. C. Underwood, and I. S. Mian. Stochastic context-free grammars for modeling RNA. In *27th Hawaii Int Conf System Sciences*, pages 349–58, 1993.

E. Sciacca, S. Spinella, D. Ienco, and P. Giannini. Annotated stochastic context free grammars for analysis and synthesis of proteins. In C. Pizzuti, M. Ritchie, and M. Giacobini, editors, *Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*, volume 6623 of *Lecture Notes in Computer Science*, pages 77–88. Springer Berlin / Heidelberg, 2011. ISBN 978-3-642-20388-6.

D. B. Searls. The language of genes. *Nature*, 420(6912):211–217, November 2002. ISSN 0028-0836.

D. B. Searls. A primer in macromolecular linguistics. *Biopolymers*, 99(3):203–217, 2013.

S. Seemayer, M. Gruber, and J. Söding. CCMpred — fast and precise prediction of protein residue-residue contacts from correlated mutations. *Bioinformatics*, 2014.

N. Sharon and H. Lis. Legume lectins–a large family of homologous proteins. *The FASEB Journal*, 4(14):3198–3208, 1990. PMID: 2227211.

C. Sigrist, L. Cerutti, N. Hulo, A. Gattiker, L. Falquet, M. Pagni, A. Bairoch, and P. Bucher. PROSITE: a documented database using patterns and profiles as motif descriptors. *Briefings in Bioinformatics*, 3:265–274, 2002.

C. J. A. Sigrist, E. de Castro, L. Cerutti, B. A. Cuche, N. Hulo, A. Bridge, L. Bougueleret, and I. Xenarios. New and continuing developments at prosite. *Nucleic Acids Research*, 41(D1): D344–D347, 2013.

N. A. Smith and J. Eisner. Guiding unsupervised grammar induction using contrastive estimation. In *IJCAI Workshop on Grammatical Inference Applications*, pages 73–78, 2005.

J. Soeding. Protein homology detection by hmmhmm comparison. *Bioinformatics*, 21(7):951–960, 2005.

E. L. L. Sonnhammer, S. R. Eddy, E. Birney, A. Bateman, and R. Durbin. Pfam: Multiple sequence alignments and hmm-profiles of protein domains. *Nucleic Acids Research*, 26(1):320–322, 1998.

Z. Sükösd, B. Knudsen, J. Kjems, and C. N. Pedersen. Ppfold 3.0: fast rna secondary structure prediction using phylogeny and auxiliary data. *Bioinformatics*, 28(20):2691–2692, 2012.

K. Tariman. Genetic algorithms for stochastic context-free grammar parameter estimation. Master's thesis, The University of Georgia, United States, 2004.

K. Tu and V. Honavar. Unsupervised learning of probabilistic context-free grammar using iterative biclustering. In A. Clark, F. Coste, and L. Miclet, editors, *Grammatical Inference: Algorithms and Applications*, pages 224–237, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg. ISBN 978-3-540-88009-7.

O. Unold. Context-free grammar induction with grammar-based classifier system. *Archives of Control Sciences*, Vol. 15, no. 4:681–690, 2005.

O. Unold. Fuzzy grammar-based prediction of amyloidogenic regions. In J. Heinz, C. Higuera, and T. Oates, editors, *Proceedings of the Eleventh International Conference on Grammatical Inference*, volume 21 of *Proceedings of Machine Learning Research*, pages 210–219, University of Maryland, College Park, MD, USA, 05–08 Sep 2012. PMLR.

S. van der Walt, S. C. Colbert, and G. Varoquaux. The NumPy array: A structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011.

H. van Melckebeke, C. Wasmer, A. Lange, E. AB, A. Loquet, A. Böckmann, and B. H. Meier. Atomic-resolution three-dimensional structure of het-s(218289) amyloid fibrils by solid-state nmr spectroscopy. *Journal of the American Chemical Society*, 132(39):13765–13775, 2010.

G. van Rossum and J. de Boer. Interactively testing remote servers using the Python programming language. *CWI Quarterly*, 4:283–303, 1991.

M. Wall. Matthew's GAlib: A C++ genetic algorithm library. http://lancet.mit.edu/ga, 2005.

S. Wang, S. Sun, Z. Li, R. Zhang, and J. Xu. Accurate de novo prediction of protein contact map by ultra-deep learning model. *PLOS Computational Biology*, 13(1):1–34, 01 2017.

M. Weigt, R. White, H. Szurmant, J. Hoch, and T. Hwa. Identification of direct residue contacts in protein-protein interaction by message passing. *Proceedings of the National Academy of Sciences*, 106:67–72, 2009.

D. H. Younger. Recognition and parsing of context-free languages in time n3. *Information and Control*, 10(2):189 – 208, 1967. ISSN 0019-9958.