Characterization of protein secondary structure

Article *in* IEEE Signal Processing Magazine · June 2004

DOI: 10.1109/MSP.2004.1296545 · Source: IEEE Xplore

CITATIONS
49

READS
349

4 authors, including:

Madhavi Ganapathiraju
University of Pittsburgh
120 PUBLICATIONS **2,017** CITATIONS
SEE PROFILE

Judith Klein-Seetharaman
The University of Warwick
254 PUBLICATIONS **10,168** CITATIONS
SEE PROFILE

Narayanaswamy Balakrishnan
Indian Institute of Science
209 PUBLICATIONS **2,351** CITATIONS
SEE PROFILE

Characterization of Protein Secondary Structure

Application of latent semantic analysis using different vocabularies

Madhavi K. Ganapathiraju, Judith Klein-Seetharaman, N. Balakrishnan, and Raj Reddy

A small organic molecule, 11-cis retinal, a vitamin A derivative, readjusts its shape by changing a single bond on seeing light to form all-trans retinal. While this small change may not seem very significant, it makes a big difference for perception of light in human vision: this is the way by which the brain knows that a photon has landed on the eye. How? The retinal is embedded inside another molecule, called rhodopsin, that belongs to a family of molecules called proteins. Rhodopsin provides multiple molecular interactions to the retinal (Figure 1), and many of these interactions are perturbed by the small change in the retinal induced by light. These perturbations in the immediate neighborhood of the retinal induce other perturbations in more distant parts of rhodopsin, and these changes are recognized by other proteins that interact with rhodopsin, inducing a complex cascade of molecular changes. These molecular changes are ultimately converted into an electrical signal that is recognized by the neurons in the brain. Thus the initial information of light isomerization is the signal that is processed by the proteins so that the human body can understand and react to it. "Signal transduction," as the transport of information is called, is only one of the many functions performed by proteins. Proteins are undoubtedly the most important functional units in living organisms, and there are tens of thousands of different proteins in the human body. Understanding how these proteins work is crucial to the understanding of the complex biological functions and malfunctions that occur in diseases.

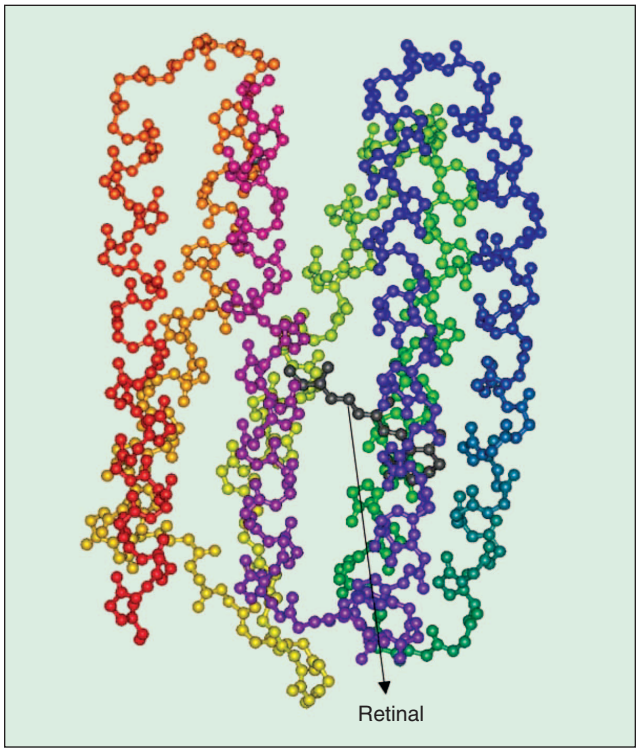
What do proteins look like? Proteins are composed of fundamental building blocks of chemical molecules called amino acids. When a protein is synthesized by the cells, initially it is just a string of amino acids. This string arranges itself in a process called protein folding into a complex three-dimensional structure capable of exerting the function of the specific protein. We will briefly review the fundamental building blocks of proteins, their primary and secondary structure (for references, see [1]).

Amino Acids—Building Blocks of Proteins

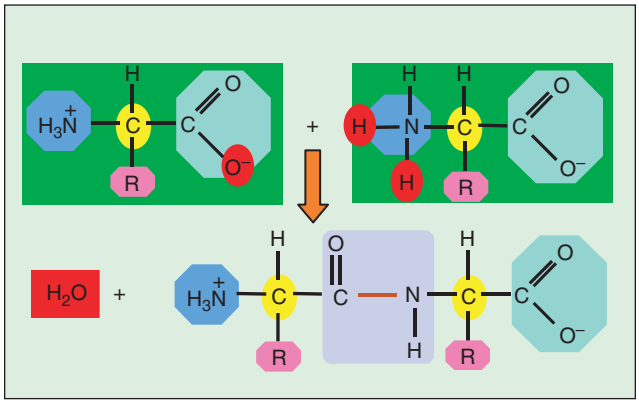
There are 20 different amino acids. The basic chemical composition common to all 20 amino acids is shown in Figure 2 (dark-green box). The central carbon atom, called Cα, forms four covalent bonds, one each with NH_3^+ (amino group), COO^- (carboxyl group), H (hydrogen), and R (side chain). The first three are common to all amino acids; the side-chain R is a chemical group that differs for each of the 20 amino acids. The side chains of the 20 amino acids are shown in Figure 3, along with their three-letter codes and one-letter codes commonly used to represent the amino acids.

The 20 amino acids have distinct chemical properties. Many different classification schemes for grouping amino acids according to their properties have been proposed, and several hundred different scales relating the 20 amino acids to each other are available (see e.g., the online databases PDbase [2] and ProtScale [3]). As an example, Figure 3 shows the amino acids grouped based on their electronic properties, i.e., some are electron donors while others are electron acceptors or are

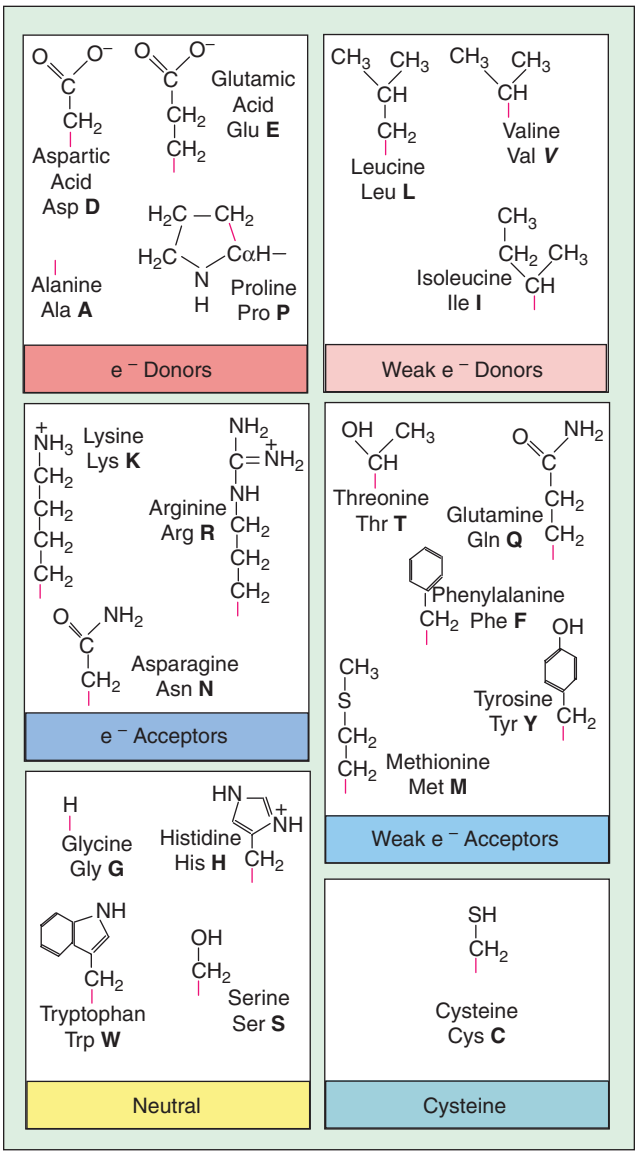
1053-5888/04/$20.00©2004IEEE



▲ *1. Rhodopsin—a member of the G-protein coupled receptor (GPCR) family. GPCRs form one of the most important families of proteins in the human body. They play a crucial role in signal transduction. The molecular architecture includes seven helices that transverse the membrane. The view shown is from the plane of the membrane. Progression from N-terminus (extracellular) to C-terminus (intracellular) is shown in rainbow colors.*

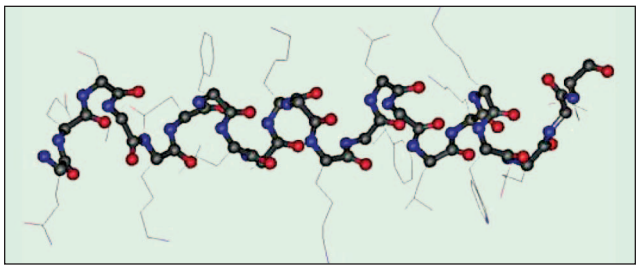


▲ *2. Amino acids and peptide bond formation. The basic amino acid structure is shown in the dark green color box. Each amino acid consists of the C-alpha carbon atom (yellow) that forms four covalent bonds, one each with: i) NH_3^+ amino group (blue), ii) COO^- carboxyl group (light green), iii) a hydrogen atom, and iv) a* side-chain R *(pink). In the polymerization of amino acids, the carboxyl group of one amino acid (shown in light green) reacts with the amino group of the other amino acid (shown in blue) under cleavage of water, H_2O (shown in red). The link that is formed as a result of this reaction is the* peptide bond. *Atoms participating in the peptide bond are shown with a violet background.*

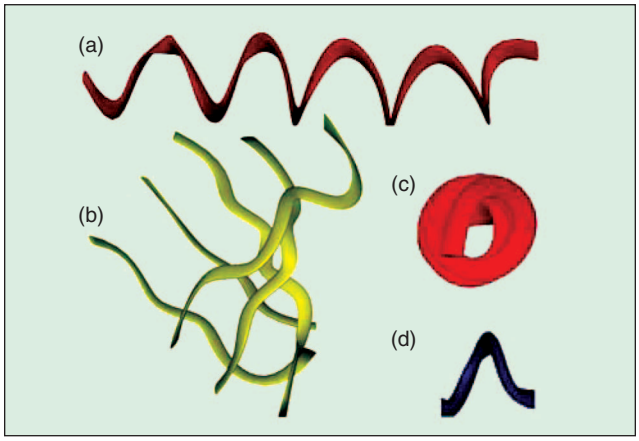


▲ *3. Side chains of the 20 amino acids: Side chains of each of the amino acids are shown, along with their three-letter and one-letter codes. Amino acids are grouped as B: Strong e^--donors, J: weak e^--donors, O: neutral, U: weak e^--acceptors, Z: strong e^-acceptors, and C: cysteine by itself in a group.*

neutral. The major difficulty in classifying amino acids by a single property is the overlap in chemical properties due to the different chemical groups that the amino acid side-chains are composed of. However, three amino acids are difficult to classify because of their properties, i.e., cysteine, proline, and glycine. Cysteine contains a sulphur (S) atom and can form a covalent bond with the sulphur atom of another cysteine. The disulphide bond gives rise to tight binding between these two residues and plays an important role for the structure and stability of proteins. Similarly, proline has a special role because the backbone is part of its side-chain structure. This restricts the conformations of amino acids and can result in kinks in otherwise regular protein structures. Glycine has a side chain that consists of only one hydrogen atom (H). Since H



▲ *4. Example of a peptide. The main chain atoms are shown in bold ball and stick representation: C-alpha and carbonyl carbon (black), nitrogen (blue), and oxygen (red). The side chain atoms are shown as thin lines. Hydrogen atoms are not shown.*



▲ *5. Some basic secondary structure types. (a) Side view of a helix. Every seventh residue (corresponding to two turns of a helix) is aligned. Therefore, every third to fourth residue is located on the same side of the helix. (b) View of the regular helix shown in (a) from the top. (c) β-sheet is a result of long-range interactions. The strands participating in the β-ladder interact with each other by way of hydrogen bonds. The interactions are long range because two strands in a β-ladder may be separated by a large number of other residues and possibly other structures. (d) View of a turn, causing a U-bend in the protein.*

is very small, glycine imposes much less restrictions on the polypeptide chain than any other amino acid.

Proteins are formed by concatenation of these amino acids in a linear fashion (like beads in a chain). Amino acids are linked to each other through the so-called peptide bond. This bond forms as a result of the reaction between the carboxyl and amino groups of neighboring residues (a residue is any amino acid in the protein), shown schematically in Figure 2. The oxygen (O) from the carboxyl group on the left amino acid and two hydrogens (H) from the amino group on the right amino acid get separated out as a water molecule (H_2O), leading to the formation of a covalent bond between the carbonyl carbon (C) and nitrogen (N) atom of the carboxyl and amino groups, respectively. This covalent bond, which is fundamental to all proteins, is the peptide bond. The carboxyl group of the right amino acid is free to react in a similar fashion with the amino group of another amino acid. The N, C, O, and H atoms that participate in the peptide bond, along with CαH, form the main-chain or the backbone of the protein sequence. The side-chains are connected to the Cα. The progression of peptide bonds between amino acids gives rise to a protein chain. A short chain of amino acids joined together through such bonds is called a peptide, a sample of which is shown in Figure 4. Backbone atoms are shown in bold, side chains on Cα atoms are shown as line-diagrams. Inside the cell, the synthesis of proteins happens in principle in the same fashion as outlined above by joining amino acids one after the other from left to right, except that in the cell proteins control each step.

Conventionally, a protein chain is written left to right, beginning with the NH_3^+ (amino) group on the left, and ending with the COO^- (carboxyl) group on the right. Hence, the left end of a protein is called N-terminus and the right end is called a C-terminus.

Secondary Structure

Inspection of three-dimensional structures of proteins such as the one shown in Figure 1 has revealed the presence of repeating elements of regular structure, termed "secondary structure." These regular structures are stabilized by molecular interactions between atoms within the protein, the most important being the so-called hydrogen (H) bond. H-bonds are noncovalent bonds formed between two electronegative atoms that share one H. There is a convention on the nomenclature designating the common patterns of H-bonds that give rise to specific secondary structure elements, the Dictionary of Secondary Structures of Proteins (DSSP) [4]. DSSP annotations mark each residue (amino acid) to be belonging to one of seven types of secondary structure: H (alpha-helix), G (3-helix or 3_{10} helix), I (5-helix or π-helix), B (residue in isolated beta-bridge), E (extended strand participates in β-ladder), T (Hydrogen bond turn), S (bend), and "_" (when none of the above structures are applicable).

The first three types are helical, designating secondary structure formed due to H-bonds between the carbonyl group of residue i and the NH group on the $i+n$th residue, where the value of n defines whether it is a $3_{10}-(n=3)$, $\alpha-(n=4)$ or $\pi-(n=5)$ helix. Therefore, the interactions between amino acids that lead to the formation of a helix are local (within six amino acids) to the residues within the helix. Figure 5(a) shows an example of a protein segment that has a general helix structure. The top view of the helical structure is shown in Figure 5(b) that shows a perfect circle arising due the well-aligned molecules in the α-helix. Sheets, on the other hand, form due to long-range interaction between amino acids, that is, residues $i, i+1 \ldots i+n$ form hydrogen bonds with residues $i+k, i+k+1 \ldots i+k+n$ (parallel beta sheet), or with residues $i+k, i+k-1 \ldots i+k-n$ (anti-parallel beta sheet). Figure 5(c) shows protein segments that conform to a sheet. A turn is defined as a short segment, that causes the protein to bend [Figure 5(d)].

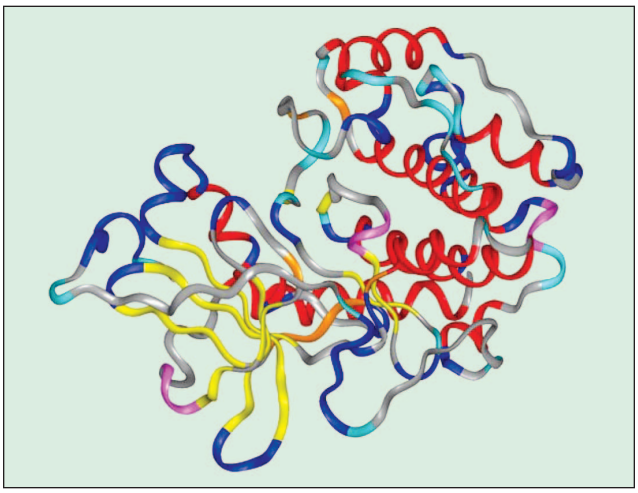
Typically, the seven secondary structure types are reduced into three groups, helix (includes types "H," alpha helix and "G," 3_{10} helix), strand (includes "E," beta-ladder and "B," beta-bridge), and coil (all other types). Figure 6 shows the secondary structure types helix, strand, turn and coil in the context of a single protein (the catalytic subunit of cAMP-dependent protein kinase).

Secondary Structure Prediction

Advances in genome sequencing have made available amino acid compositions of thousands of proteins. However, determination of the function of a protein from its amino acid sequence directly is not always possible. Knowing the three-dimensional shape of the protein, that is, knowing the relative positions of each of the atoms in space, would give information on potential interaction sites in the protein, which would make it possible to analyze or infer the function of the protein. Thus the study of determining or predicting protein structure from the amino acid sequences has secured an important place both in experimental and computational areas of research. The experimental methods X-ray crystallography and nuclear magnetic resonance (NMR) spectroscopy can accurately determine protein structure; but these experimental methods are labor intensive and time consuming and for some proteins are not applicable at all. For example, X-ray crystallography requires precipitation of proteins into regular crystals, an empirical process which is very difficult to achieve—so much so, that it is being experimented with growing protein crystals on the International Space Station.

Protein structure is dependent primarily on its amino acid sequence. The structure prediction problem is, in principle, a coding theory problem; but the number of possible conformations is too large to exhaustively enumerate the possible structures for proteins. For example, a small protein might consist of 100 amino acids, and the backbone is defined by two degrees of freedom for the angles around the bonds to the left and right of the Cα, this is $2^{100} \approx 10^{29}$ possible combinations. Even with advanced statistical and information theories and exponential increases in computing power, this is not yet feasible. Furthermore, multuple structures may be sterically compatible with a given protein sequence. Often, two different amino acid sequences possess the same structure, while sometimes, although very rarely, the same amino acid sequence gives rise to different structures. These complexities are intractable by current computational methods.

As an intermediate step towards solving the grander problem of determining three-dimensional protein structures, the prediction of secondary structural elements is more tractable but is in itself a not yet fully solved problem. Protein secondary structure prediction from amino acid sequence dates back to the early 1970s, when Chou and Fasman, and others, developed statistical methods to predict secondary structure from primary sequence [5], [6]. These first methods were based on the patterns of occurrence of specific amino acids in the three secondary structure types—helix, strand, and coil. These methods achieved a three-class accuracy (Q_3) of less than 60%. In the next generation methods, coupling effects of neighboring residues were considered, and moving-window computations have been introduced. These methods employed pattern matching and statistical tools including information theory, Bayesian inference and decision rules [4], [7]–[9]. These methods also used representations of protein sequences in terms of the chemical properties of amino acids with particular emphasis on polar-nonpolar patterns and interactions [10], [11], amino acid patterns in different types of helices [12], electronic properties of amino acids and their preferences in different structures [4], structural features in side chain interactions [13], [14]. The Q_3 accuracy was still limited to about 65%, the reason for this being that only local properties of amino acids have been used. Long-range interactions that are particularly important for strand predictions were not taken into account. In the 1990s, secondary structure prediction methods began making use of evolutionary information from alignments of sequences in protein sequence databases that match the query sequence. These methods have taken Q_3 accuracy up to 78% [15]. However, for a substantial number of proteins such alignments are not found and techniques using sequence alone are still necessary.



▲ *6. A protein that is color coded based on annotation by the DSSP: The protein shows only the main chain, with the following color codes: H: α-helix (red), G: 3_{10}-helix (pink), E: extended strand that participates in β-ladder (yellow), B: residue in isolated β-bridge (orange), T: hydrogen bond turn (dark blue), and S: bend (light blue). Residues not conforming to any of the previous types are shown in grey. The protein is the catalytic subunit of cAMP-dependent protein kinase (pdb ID 1BKX, available at http://www.rcsb.org/pdb).*

Proteins and Language

Protein sequence analysis is in many respects similar to human language analysis, and just as processing of text needs to distinguish signals from noise, the challenge in

protein sequence analysis is to identify "words" that map to a specific "meaning" in terms of the structure of the protein—the greatest challenge being identification of "word"-equivalents in protein sequence "language." Understanding the protein structures encoded in the human genome sequence has therefore been dubbed "reading the book of life." Knowledge/rule based methods, machine learning methods (such as hidden Markov models and neural networks), and hybrid methods have been used to capture meanings from a sequence of words in natural languages and prediction of protein structure and function alike. Protein secondary structure prediction and natural language processing aim at studying higher order information from composition, while tackling problems like redundancy and multiple interpretations. In recent years, expressions such as text segmentation, data compressibility, Zipf's law, grammatical parsing, n-gram statistics, text categorization and classification, and linguistic complexity which were normally used in the context of natural language processing have become common words in biological sequence processing.

Latent semantic analysis (LSA) is a natural language text processing method that is used to extract hidden relations between words by way of capturing semantic relations using global information extracted from a large number of documents rather than comparing the simple occurrences of words in two documents. LSA can therefore identify words in a text that address the same topic or are synonymous, even when such information is not explicitly available, and thus finds similarities between documents, when they lack word-based document similarity. LSA is a proven method in the case of natural language processing and is used to generate summaries, compare documents, and generate thesauri and further for information retrieval [16], [17]. Here, we review our recent work on the characterization of protein secondary structure using LSA. In the same way that LSA captures conceptual relations in text, based on the distribution of words across text documents, we use it to capture secondary structure propensities in protein sequences using different vocabularies.

Table 1. Word document matrix for the sample protein shown in Figure 8. The rows correspond to the words (amino acids shown here), and columns correspond to the documents.

	Document Number								
Vocabulary	1	2	3	4	5	6	7	8	9
A	0	0	0	1	0	0	1	1	0
C	0	0	0	0	0	0	0	0	0
D	0	0	0	0	1	0	0	0	0
E	0	0	0	0	0	0	0	0	0
F	1	1	0	0	0	0	0	0	0
G	0	0	1	0	0	0	0	1	0
H	0	0	0	0	0	0	0	0	1
I	0	2	0	1	0	0	0	0	0
K	2	0	0	1	0	0	0	1	0
L	0	2	0	0	0	1	1	1	1
M	0	0	0	0	0	0	0	1	0
N	1	2	1	0	1	0	0	0	2
P	3	0	0	0	0	0	3	0	0
Q	0	1	0	0	0	0	0	0	0
R	0	2	0	0	0	0	0	0	0
S	0	0	0	0	0	1	0	0	0
T	0	1	0	0	0	0	1	0	0
V	1	1	0	1	0	1	0	0	0
W	0	0	0	1	0	0	0	0	0
Y	0	0	0	1	0	0	0	1	1

Latent Semantic Analysis

In LSA, text documents and the words comprising these documents are analyzed using singular value decomposition (SVD). Each word and each document is represented as a linear combination of hidden abstract concepts. LSA can identify the concept classes based on the co-occurrence of words among the documents. It can identify these classes even without prior knowledge of the number of classes or the definition of concepts, since the LSA measures the similarity between the documents by the overall pattern of words rather than by the specific constructs. This feature of LSA makes it amenable for use in applications like automatic thesaurus acquisition. LSA and its variants such as probabilistic latent semantic indexing are used in language modeling, information retrieval, and text summarization and other such applications. A tutorial introduction to LSA in the contexts of text documents is given in [16] and [17]. Basic construction of the model is described here, first in terms of text documents, followed by adaptation to analysis of biological sequences. In the paradigm considered here, the goal of LSA is as follows: for any new document unseen before, identify the documents present in the given document collection that are most similar thematically.

Let the number of given documents be N; let A be the vocabulary used in the document collection and let M be the total number of words in A. Each document d_i is represented as a vector of length M

$$d_i = [C_{1i}\ \ C_{2i}, \ldots, C_{Mi}]$$

where C_{ji} is the number of times word j appears in document i, and is zero if word j does not appear in document i. The entire document collection is represented as a matrix where the document vectors are the columns. The matrix, called a word-document matrix, would look as shown in Table 1 and would have the form

$$W = [C_{ji}],\ 1 < j < M,\ 1 < i < N.$$

The information in the document is thus represented in terms of its constituent words; documents may be

compared to each other by comparing the similarity between document vectors. To compensate for the differences in document lengths and overall counts of different words in the document collection, each word count is normalized by the length of the document in which it occurs, and the total count of the words in the corpus. This representation of words and documents is called vector space model (VSM). Documents represented this way can be seen as points in the multidimensional space spanned by the words in the vocabulary. However, this representation does not recognize synonymous or related words. Using thesaurus-like additions, this can be incorporated by merging the word counts of similar words into one. Another way to capture such word relations when explicit thesaurus-like information is not available is to use LSA.

In the specific application of vector space model to LSA, SVD is performed on the word-document matrix that decomposes it into three matrices related to W as

$$W = USV^T$$

where U is $M \times M$, S is $M \times M$, and V is $M \times N$. U and V are left and right singular matrices, respectively. SVD maps the document vectors onto a new multidimensional space in which the corresponding vectors are the columns of the matrix SV^T. Matrix S is a diagonal matrix whose elements appear in decreasing order of magnitude along the main diagonal and indicate the energy contained in the corresponding dimensions of the M-dimensional space. Normally only the top R dimensions for which the elements in S are greater than a threshold are considered for further processing. Thus, the matrices U, S, and V are reduced to $M \times R$, $R \times R$ and $R \times N$, respectively, leading to data compression and noise removal. The space spanned by the R vectors is called eigenspace.

The application that LSA is used for in this work is in direct analogy document classification. Given a corpus of in discrete analogy documents that belong to different "topics," the goal is to identify the topic to which a new document belongs. Document vectors given by the columns in SV^T can be compared to each other using similarity measure such as cosine similarity. Cosine similarity is the measure of the cos of the angle between two vectors. It is one when the two vectors are identical and zero when the vectors are completely orthogonal. The document that has maximum cosine similarity to the given document is the one that is most similar to it.

Given a new document that was not in the corpus originally (called a test document or unseen document), documents similar to it semantically may be retrieved, from the corpus by retrieving documents having high cosine similarity with the test document. For "document classification," the topic of the new document is assigned the same topic as that of the majority of the similar documents that have been retrieved. The number of similar documents retrieved from the corpus may be controlled by way of a threshold, or by choosing to retrieve only "k most similar documents." The assignment of topic of the unseen data based on that of the majority of the k most similar documents is called k-nearest neighbor (kNN) classification. For each such unseen document i, its representation t_i in the eigenspace needs to be computed. Since SVD depends on the document vectors from which it is built, the representation of t_i is not directly available. Hence to retrieve semantically similar documents, it is required to add the unseen document to the original corpus, and then vector space model and latent semantic analysis model be recomputed. The unseen document may then be compared in the eigenspace with the other documents, and the documents similar to it can be identified using cosine similarity.

In case of text document retrieval where the size of the word-document matrix W is large, say in the order of $2{,}000 \times 10{,}000$ on the lower side, SVD would be computationally expensive; also, it usually requires real time response. Hence performing SVD every time a new test document is received is not suitable. However, from the mathematical properties of the matrices U, S, and V, the test vector may be approximated as

$$t_i = \tilde{d}_j U$$

where $\tilde{d}_j$ is the segment vector normalized by word counts in W and by segment length [16]. This approximation is in the context of text processing. In case of protein secondary structure prediction, it is a one-time computation and does not impose hard response time requirements. Hence the segments from the modeling set and the unseen set may be used together to form the word-document matrix before SVD. For every new set of proteins, SVD can be computed.

Application of LSA to Proteins

In natural language, text documents are represented as bag-of-words in the LSA model. For the application of LSA to protein sequences, first a suitable analogy for words has to be identified. As described above, known functional building blocks of protein sequences are the 20 amino acids. These have been used in all previous secondary structure prediction methods. However, it is known that often a particular chemical subgroup within the side chain of an amino acid bears "meaning" for the structure of a protein, while in other instances amino acids with similar properties can be exchanged by each other. This introduces ambiguity in the choice of the vocabulary and we therefore experimented with three different vocabularies, the amino acids, chemical subgroups and amino acid types. The amino acid types are derived based on the overall similarity of the chemical properties of individual amino acids, based on those outlined in Figure 3. The chemical subgroups were derived from the amino acid structures shown in Figure 3 in the procedure shown in Figure 7. Each amino acid was

decomposed into the individual functional groups, such as carbonyl groups and amino groups. Thus, 18 different chemical groups were derived from the 20 different amino acids. In analogy to the treatment of text documents as bag-of-words, we then treated segments of protein sequences that belong to a particular secondary structural type as "documents" that are composed of bag-of-X, where $X =$ amino acids, amino acid types or chemical groups depending on the vocabulary used. To illustrate how we arrive at the equivalent of the word-document matrix described above, a segment of a protein from the dataset is shown in Figure 8 as an example. The top row shows the amino acids (residues) of the segment and the bottom row shows the corresponding DSSP label, where the three equivalence classes: $X = \{\mathrm{H}, \mathrm{G}\}$, $\Upsilon = \{\mathrm{B}, \mathrm{E}\}$, $Z = \{\mathrm{I}, \mathrm{S}, \mathrm{T}\}$ of the DSSP assignments were used as category equivalents. In the sample sequence, helix class (X) is shown in red, sheet class (Υ) is shown in yellow, and the random coil class (Z) is shown in blue, corresponding to the color coding in Figure 6. Subsequences of proteins in which consecutive residues form the same secondary structure are treated as documents. For example, the protein sequence shown in Figure 8 would give rise to nine documents namely, PKPPVKFN, RRIFLLNTQNVI, NG, YVKWAI, ND, VSL, ALPPTP, YLGAMK, and YNLLH.

The corpus used here was derived from the JPred distribution material, a secondary structure prediction benchmark dataset. It consists of 513 protein sequences, with DSSP annotations for each amino acid [18]. Other information such as multiple sequence alignment was not used in this analysis. Computations are performed using the MATLAB software package. To accommodate the computation of SVD of the large word document matrix, only a randomly selected subset of 80 proteins from the 513 proteins in the JPred set were chosen for our corpus. Future validation of a larger set of proteins may be performed by special purpose SVD packages such as SVDPack [19]. Of the 80 proteins used here, 50 proteins were chosen to represent the "training" or "seen" data and the remaining 30 proteins the "test" data. Validation was done using leave-one-out testing as described before. The protein sequences are then separated based on their structural segments. These segments (documents) put together form the corpus. Using the chemical group vocabulary with size $M = 18$, and the number of documents being the number of segments obtained from the 50 protein sequences, 1,621, the word-document matrix W is of size $18 \times 1{,}621$. Similarly, in the case of the amino acid vocabulary, its size would be $20 \times 1{,}621$.

Assignment of Secondary Structure Category to Unseen Data

Let the documents $d_1, d_2, \ldots, d_{NI}$ be the nonoverlapping protein segments for which structural categories $C_1, C_2, \ldots, C_{NI}$ are known. Let $t_1, , t_2, \ldots, t_{N2}$ be the nonoverlapping test segments with known length for which secondary structure is to be predicted. A kNN classification is used to predict secondary structure of the test data. For each test segment (document) t_i, the cosine similarity of t_i to all the training segments $d_1, d_2 \ldots d_{NI}$ is computed, of which k segments that have maximum similarity to t_i are identified. These k segments are the kNN of t_i. Structural category S to which most of the kNN of t_i belong is the predicted category of t_i. This process is repeated for each of the test segments.

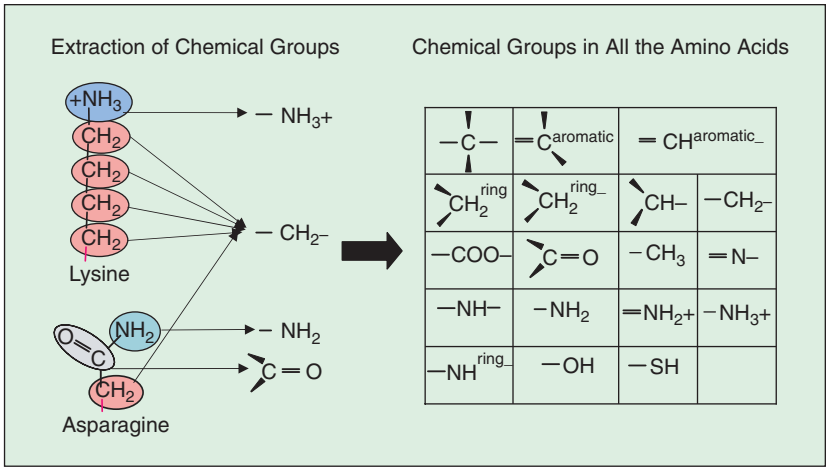
Results

Tables 2 and 3 show the results of the vector space model and the latent semantic analysis model, respectively, for the prediction of helix, strand, and coil. Performance measures employed here are same as those used in information retrieval, namely, precision and recall. Precision of

Table 2. Results of secondary structure prediction into three classes (helix, strand, coil) using vector space model and different choices of vocabulary.

			Results with Vector Space Model									
			Precision					Recall				
	Vocabulary		Helix	Strand	Coil	Micro Average	Macro Average	Helix	Strand	Coil	Micro Average	Macro Average
1	Amino Acids	Seen data	97.8	56.7	91.4	82.7	81.9	99.6	87.6	65.9	77.5	84.3
2		Unseen data	42.7	30.1	83.3	62.0	52.0	65.8	67.3	20.0	40.6	51.0
3	Chemical Groups	Seen data	96.7	58.9	92.9	83.6	82.6	99.6	88.3	68.4	79.0	85.4
4		Unseen data	64.6	53.9	78.4	69.5	65.5	55.3	48.7	85.7	69.7	63.0
5	Amino Acid Types	Seen data	77.1	57.0	81.7	72.0	NA	95.5	80.3	28.8	68.1	NA
6		Unseen data	72.5	48.4	77.4	66.1	NA	84.9	71.1	27	61.1	NA

any category refers to the ratio of correctly predicted segments in a category to the total number of segments predicted to belong to that category. Overall precision is the average precision across all the categories to which the segments belong. Average can be calculated in one of the two ways: microaverage is the average per segment, and macroaverage is the average per category. Recall of any category is the ratio of the number of segments correctly predicted from that category to the total number of segments actually belonging to that category. The models were tested by the leave-one-out method. Each segment in the training data is treated as a test segment, its category unknown, and is assigned a structural category by kNN algorithm with respect to the remaining training segments. The results of this are termed "seen-data" in the tables.



▲ *7. Derivation of the chemical group vocabulary: The basic chemical groups that form the building blocks of the amino acids are shown for two examples: lysine and asparagine. The chemical groups are identified by circles and correspond to one word in the vocabulary. This is carried out for all 20 amino acids. The final chemical group vocabulary is shown on the right. Its size is 18.*

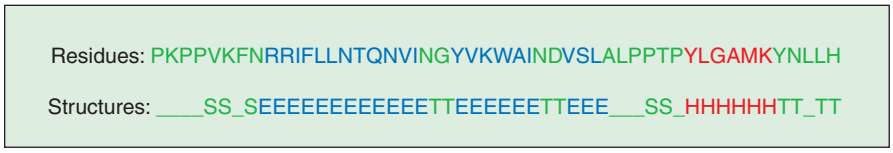
Precision and recall values for classification are presented for both seen data and unseen data in the first two rows in each table, for amino acids as vocabulary. The word-document matrix is first constructed using the segments for which secondary is known and is often called the training set. This in Tables 2 and 3 is represented as "Seen data." Since the overall word counts are based on the seen data, the segments representation is more accurate for seen data. The word count normalizations for unseen or test data are approximated by the statistics of previously constructed word-document matrix and its analysis. Apart from this distinction, there is no other advantage in seen data in comparison to unseen data. In each model (VSM and LSA), precision and recall accuracies for each of the vocabulary types are given for each individual class separately, followed by the micro- and macro-averages for the three classes.

Vocabulary: Amino Acids

The results for VSM and LSA using the amino acids as vocabulary are shown in Tables 2 and 3, respectively (first and second rows). The precision of both helix and sheet are higher with LSA than with VSM: 69.1 and 52.3%, in comparison to 42.7 and 30.1%, respectively. Only coil is predicted more accurately with VSM. The recall values drop when going from VSM to LSA but yield better confidence in secondary structure assignment. The average performance over the three classes (helix, strand, and coil), of both precision and recall, is significantly better with the combination of LSA with amino acids as vocabulary.

Table 3. Results of secondary structure prediction into three classes (helix, strand, coil) using latent semantic analysis and different choices of vocabulary.

			Results with Latent Semantic Analysis									
			Precision					Recall				
	Vocabulary		Helix	Strand	Coil	Micro Average	Macro Average	Helix	Strand	Coil	Micro Average	Macro Average
1	Amino Acids	Seen data	98.9	60.1	94.9	85.8	84.6	99.6	92.1	69.4	80.6	87.1
2		Unseen data	69.1	52.3	73.6	67.1	67.7	42.8	49.6	84.4	67.6	58.9
3	Chemical Groups	Seen data	99.6	66.2	82.7	82.6	80.9	99.6	89	54.2	81	80.9
4		Unseen data	80	50	50	55.7	59.7	40	40	80	64.4	55.1
5	Amino Acid Types	Seen data	82.7	53.3	75.6	70.6	70.6	96.2	81.4	23.5	67	67
6		Unseen data	90	70	30	60.5	60.5	70	50	70	63.5	63.5



▲ *8. A sample protein with DSSP annotations: First row, called residues, shows a protein (only a part of the protein is shown here, as an example). The letters indicate the amino acids in the protein sequence. The second row, called structures, shows DSSP label of the amino acid in the corresponding position in the first row. The color coding indicates the grouping of the structures into three classes: helices = {H, G} in red, sheet = {B, E} in blue, and coil = {T, S, I, '_'} in green.*

Vocabulary: Chemical Groups

Next, we studied the effect of increasing the detail in description of the amino acids by rewriting the sequence using chemical groups as vocabulary and explored the performance of the two models using this vocabulary. The chemical groups represent the amino acids in greater detail, namely in terms of their chemical composition. Thus, overlap in chemical properties because of the same chemical group being a component of the amino acid side chain is accounted for using this vocabulary. The basic chemical groups that form the building blocks in the 20 amino acids that were treated as "words" are shown in Figure 7.

Tables 2 and 3 show the secondary structure classification accuracy using chemical composition using VSM and LSA, respectively (third and fourth rows). For VSM, the choice of the chemical composition as vocabulary as opposed to the amino acids is clearly advantageous. The increases in precision for helix and strand are comparable to those seen when going from VSM to LSA in the case of amino acids. The precision of coil prediction is similar for both amino acid and chemical group vocabularies. For the prediction of helix, going from VSM to LSA gives even better results. However, the strand and coil predictions are comparable or lower in LSA than in VSM. Thus, for the chemical vocabulary, the combination of VSM with chemical groups gives the best Q_3 performance in precision.

One might argue that LSA is already capable of extracting synonymous words; and hence that it would be able to identify similarities between amino acids. However similarity of amino acids arises due to similarity in chemical composition whereas, LSA determines synonymy based on context; hence it might give additional advantage to give explicit indication of amino acid similarity.

Vocabulary: Amino Acid Types

Finally, we investigated the effect of decreasing the detail in the description of the amino acid sequence. While the chemical vocabulary, studied in the previous section, is more detailed than the amino acid vocabulary, the amino acid type vocabulary is less detailed than the amino acid vocabulary. Amino acid types are basically a reduced set of amino acids in which amino acids were mapped into different classes based on their electronic properties. Words would then be the "classes of amino acids." As described in the introduction, amino acids can be grouped by their chemical properties. Since there is significant overlap in chemical properties of the 20 different amino acid side chains, many different reduced vocabularies have been proposed. The most simple and widely used classification scheme is to define two groups, hydrophobic and polar [20], [21]. There are also various alphabets with letter size between 2 and 20 [21]–[23]. The grouping of amino acids that is used in this work is shown in Figure 3.

The results for VSM and LSA using the amino acid types as vocabulary are shown in Tables 2 and 3, respectively (fifth and sixth rows). Using amino acid types as the vocabulary slightly improved classification accuracy of helix in comparison to using chemical groups, but did not have significant effect on strand and coil when using the VSA model. However, when the LSA model was applied, the combination of the LSA model with this vocabulary yielded by far the best prediction accuracy for helix and strand types, also keeping the recall value high. Helix was predicted with 90% and strand with 70% precision in comparison to 80% and 53.9%, the best results with any of the other combinations of models and vocabularies. However, the prediction of coil using LSA and amino acid type was very poor. In this case, the VSM with using amino acids as vocabulary was best, most likely due to the highly predictive nature of proline for coil due to its disruptive nature for regular secondary structure (see introduction).

Conclusions and Future Work

While the average three-class precision (Q_3) was best using chemical groups as vocabulary and using VSM analysis, classification accuracy in individual classes was not the best with this model. Helices and sheets were best classified using LSA with amino acid types as vocabulary, with 90% and 70% precision, 70% and 50% recall. Coils are characterized with higher precision using amino acids as vocabulary and VSM for analysis. The results demonstrate that VSM and LSA capture sequence preferences in structural types. Protein sequences represented in terms of chemical groups and amino acid types provide more clues on structure than the classically used amino acids as functional building blocks. Significantly, comparing results within the same analysis model (VSM or LSA), the precision in classifying helix and strand increases when going from amino acids to chemical groups or amino acid types for unseen data. Furthermore, it does not show a corresponding drop in recall. This result suggests that different alphabets differ in the amount of information they carry for a specific prediction task within a given prediction method. Future work includes testing other types of amino acid alphabets.

The analysis presented here is based on sequences alone, without the use of any evolutionary information or global optimization which yields up to 78% Q_3 in third generation secondary structure prediction methods described above. While the average performance of LSA seems comparable to the best methods reported in the literature, the precision of classification yielded by LSA is shown to be higher for different secondary structure types depending on the underlying vocabulary used. Note, however that the results presented here are "per segment" and not per residue. Since the segment length information is not preserved in the LSA representation, it is not possible to directly compare these results with those in the literature, which report accuracies "per residue." Usually, the accuracy is highest in the center of the secondary structure element to be predicted with rapidly decreasing accuracy towards the edges. LSA is not dependent on this positional effect because of its nature in viewing the segments as a "bag." It is therefore likely that LSA will be highly complementary to existing secondary structure segmentation approaches. Furthermore, *n*-grams of words which are popularly used in both biological and natural language modeling, in combination with LSA and VSM, and vocabulary choices based on the prediction accuracy for individual secondary structure types may also be combined favorably. In essence, the method presented here provides a fertile ground for further experimentation with dictionaries that can be constructed using different properties of the amino acids and proteins.

Acknowledgments

This research was supported by National Science Foundation Information Technology Research Grant NSF 0225656.

Madhavi K. Ganapathiraju received her B.S. degree from Delhi University and her M.Eng. degree in electrical communications engineering from the Indian Institute of Science. She is currently a Ph.D. student at the Language Technologies Institute and a multimedia information specialist at the Institute for Software Research International at Carnegie Mellon University.

Judith Klein-Seetharaman is an assistant professor at the Department of Pharmacology at the University of Pittsburgh School of Medicine and holds secondary appointments at the Research Center Jülich (Germany) and at the School of Computer Science at Carnegie Mellon University.

N. Balakrishnan is the Satish Dhawan Chair Professor at the Indian Institute of Science. He is also chair of the Division of Information Sciences at Indian Institute of Science. He is an honorary professor at Jawaharlal Nehru Centre for Advanced Scientific Research and at National Institute of Advanced Studies.

Raj Reddy is the Herbert A. Simon University Professor of Computer Science and Robotics in the School of Computer Science at Carnegie Mellon University and the director of Carnegie Mellon West.

References

[1] J. Voet and J.G. Voet, *Biochemistry*, 2nd ed. New York: Wiley, 1995.

[2] PDbase, http://www.scsb.utmb.edu/comp_biol.html/venkat/prop.html

[3] ProtScale, http://us.expasy.org/cgi-bin/protscale.pl

[4] D.S. Dwyer, "Electronic properties of the amino acid side chains contribute to the structural preferences in protein folding," *J. Biomol. Struct. Dyn.*, vol. 18, no. 6, pp. 881–892, 2001.

[5] P.Y. Chou and G.D. Fasman, "Prediction of the secondary structure of proteins from their amino acid sequence," *Adv. Enzymol. Relat. Areas Mol. Biol.*, vol. 47, pp. 45–148, 1978.

[6] J. Garnier, D.J. Osguthorpe, and B. Robson, "Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins," *J. Mol. Biol.*, vol. 120, no. 1, pp. 97–120, 1978.

[7] J.F. Gibrat, J. Garnier, and B. Robson, "Further developments of protein secondary structure prediction using information theory. New parameters and consideration of residue pairs," *J. Mol. Biol.*, vol. 198, no. 3, pp. 425–443, 1987.

[8] N. Zhang, "Markov models for classification of protein helices," *Biochemistry*, vol. 218, 2001.

[9] S.C. Schmidler, J.S. Liu, and D.L. Brutlag, "Bayesian segmentation of protein secondary structure," *J. Comput. Biol.*, vol. 7, no. 1-2, pp. 233–248, 2000.

[10] C.D. Andrew, S. Penel, G.R. Jones, and A.J. Doig, "Stabilizing nonpolar/polar side-chain interactions in the alpha-helix," *Proteins*, vol. 45, no. 4, pp. 449–455, 2001.

[11] Y. Mandel-Gutfreund and L.M. Gregoret, "On the significance of alternating patterns of polar and non-polar residues in beta-strands," *J. Mol. Biol.*, vol. 323, no. 3, pp. 453–461, 2002.

[12] N. Zhang, "Markov models for classification of protein helices," *Biochemistry*, vol. 218, 2001.

[13] A. Thomas, R. Meurisse, and R. Brasseur, "Aromatic side-chain interactions in proteins. II. Near- and far-sequence Phe-X pairs," *Proteins*, vol. 48, no. 4, pp. 635–644, 2002.

[14] A. Thomas, R. Meurisse, B. Charloteaux, and R. Brasseur, "Aromatic side-chain interactions in proteins. I. Main structural features," *Proteins*, vol. 48, no. 4, pp. 628–634, 2002.

[15] B. Rost, "Review: Protein secondary structure Prediction continues to rise," *J. Struct. Biol.,* vol. 134, no. 2-3, pp. 204–218, 2001.

[16] J. Bellegarda, "Exploiting latent semantic information in statistical language modeling," *Proc. IEEE*, vol. 88, no. 8, pp. 1279–1296, 2000.

[17] T. Landauer, P. Foltx, and D. Laham, "Introduction to latent semantic analysis," *Discourse Processes*, vol. 25, pp. 259–284, 1998.

[18] J.A. Cuff, M.E. Clamp, A.S. Siddiqui, M. Finlay, and G.J. Barton, "JPred: A consensus secondary structure prediction server," *Bioinformatics*, vol. 14, no. 10, pp. 892–893, 1998.

[19] SVDPack, http://www.netlib.org/svdpack/

[20] H.S. Chan and K.A. Dill, "Compact polymers," *Macromolecules*, vol. 22, no. 12, pp. 4559–4573, 1989.

[21] K.F. Lau and K.A. Dill, "Lattice statistical mechanics model of the conformational and sequence spaces of proteins," *Macromolecules*, vol. 22, no. 10, pp. 3986–3997, 1989.

[22] J. Wang and W. Wang, "A computational approach to simplifying the protein folding alphabet," *Nat. Struct. Biol.*, vol. 6, no. 11, pp. 1033–1038, 1999.

[23] T. Li, K. Fan, J. Wang, and W. Wang, "Reduction of protein sequence complexity by residue grouping," *Protein Eng.*, vol. 16, no. 5, pp. 323–330, 2003.