Detect protein secondary structure based on unsupervised word segmentation

Wang Liang¹, Zhao KaiYong²

Author Affiliations
¹ Sogou Tech, Beijing, 100080, P.R. China. ² Department of Computer Science, Hong Kong Baptist University, HK, 999077, P.R. China. Correspondence to Wang Liang: wangliang.f@gmail.com

ABSTRACT

Unsupervised word segmentation methods were applied to predict protein secondary structures. Protein sequences, such as ‘MTMDKSELVQKA……’, were used as input to these methods. Segmented ‘protein word’ sequences, such as ‘MTM DKSE LVQKA’, were then obtained. The protein sequence can also be ‘divided’ into segments, such as ‘MTMD KSE LVQKA’, according to its secondary structure. The boundaries of the ‘protein words’ produced by unsupervised segmentation show the accordance with the boundary of the secondary structure. This work may spark some more new ideas to develop novel protein structure prediction methods. Our experiment also suggests there may be more protein sequences in current noncoding regions.

INTRODUCTION

In 1951, Pauling and Corey predicted the existence of two periodic motifs in protein structures: a-helix and the b-sheet which turn out to be major feature of protein structure. Protein secondary structure mainly refers to the local conformation of the polypeptide backbone of proteins that is often discretely classified into a few states.

Accurate protein structure and function prediction relies, in part, on the accuracy of secondary structure prediction. Current algorithms for predicting the secondary structure provides accuracy rates of about 80% for a 3 state prediction: a-helix, b-strand and coils using neural networks and evolutionary information. The maximum achievable prediction could reach about 85% (1).

Most secondary structure prediction methods need learn the rules from the protein having the secondary structure (training set), and then predict the secondary structure of new protein (test set). These methods are normally called supervised method in machine learning area. The ab initio protein structure prediction is not dependent on exist structure information or template, only from the amino acid sequence information. It could be regarded as the unsupervised machine learning methods. Ab initio predictions are mainly for ternary structures.

Here we present a novel ab initio prediction method for protein secondary structure based on
unsupervised word segmentation method. It only uses the amino acid sequences without any structure information to detect the secondary structures. If this method work, it could also provide another explain about how protein primary structure governs its secondary structure.

Word segmentation mainly refers to the process dividing a string of written language into its component words. For some East-Asian languages, such as Chinese, no spaces or punctuations are placed between letters. The ‘letter’ sequences must be segmented into word sequences to process the text of these languages. For example, ‘iloveapple’ is segmented into ‘I love apple’.

The word segmentation also contains the supervised and unsupervised methods. The supervised method use the segmented text like ‘I love apple’ as input. But unsupervised method use text without any space or punctuations like ‘iloveapple’. Supervised methods are better than unsupervised methods; however, unsupervised approaches are highly adaptive to relatively unfamiliar ‘languages’ for which we do not have enough linguistic knowledge (2).

The word segmentation is very similar to the protein secondary structure assignment process. For protein sequences, such as “MTMDKSELVQKA”, the corresponding consecutive amino acids of the same secondary structure can also be regarded as a ‘structure word’. This process could be called secondary structure segmentation (Fig. 1):

![Secondary structure segmentation](image)

Fig.1 Secondary structure segmentation. The protein sequence is segmented by its secondary structure. This sequence contains three secondary structure words, ‘MTM’, ‘DKSEL’, and ‘VQKA’.

Our study aims to develop unsupervised or ab initio protein secondary prediction based on unsupervised segmentation methods. This method could discovery the secondary structure by only amino acid sequences without any structure information. There have been some research the design the methods similar to supervised segmentation to analyze protein secondary structure (3). Some work also use the mutual information to detect the structure of membrane protein without any known structure (4). But we still didn’t find the comprehensive analysis of the application of unsupervised word segmentation methods in structure prediction.

The main idea of most unsupervised word segmentation methods depends on certain predefined criterion, e.g., mutual information, to recognize a substring in the text as a word (5, 6). Candidate substrings have some basic types of goodness measurements (7, 8):

1. Frequency is the occurrence of candidate substrings in the text. Only the substrings that occur more than once are considered as qualified word candidates. Frequency mainly describes the certainty in a substring (9).

2. Boundary entropy is the measure of the uncertainty before or after the current substring.
(3) Description length gain performs segmentation to maximize the compression effect, which is a global effect throughout the text (11).

The goodness score of a candidate correlates to the probability that it is a true word. After constructing a word list with goodness, word segmentation methods can be designed to segment the letter sequence into a ‘word’ sequence. A Viterbi-style algorithm is usually applied to search for the segmentation with the highest goodness.

There are two representative unsupervised segmentation methods. The first is the Soft-counting method, which use the standard EM algorithm (12) to estimate parameters. The second is HDP (Hierarchical Dirichlet Process model), which is based on Bayes Inference theory (13, 14, 15) and mainly uses Gibbs sampling to estimate parameters. The later one is much more complex than first one, but could get better results in most conditions. This paper applies all these two methods to analyze the protein secondary structure. But we mainly introduce the soft-counting in method section, mainly because it’s easy to understand for interdisciplinary researchers.

The unsupervised methods can’t assign the structure types because they don’t use any structure information. So its segmentation performance is evaluated using the boundary F-score measure, $F = 2RP/(R + P)$. The recall R and precision P are the proportions of the correctly recognized word boundaries to all boundaries in the gold-standard and an output for word segmentation of a segmenter, respectively (Fig.2).

![Fig.2](image.png) Evaluation of word segmentation. There are 2 segmentation points in structure segmentation and 3 segmentation points in unsupervised word segmentation, 1 point match. So the precision is $1/33\approx3\%$ and the recall are $1/2=50\%$. The F-score is $2*0.33*0.5/(0.33+0.5)=39.8\%$

**RESULTS**

Unsupervised segmenting methods require only raw letter sequences. Here is the amino acid letter sequences. Data are selected from the PDB structure database. The PDB dataset contains about 100,000 protein sequences with secondary structures. These protein sequences are used as input to the unsupervised segmentation method. Here we select soft-counting and HDP methods. The corresponding secondary structure segmentations are treated as the gold-standard segmentation.

There are many kinds of secondary structure assignment methods such as DSSP, STRIDE, etc.
These different methods provide different assignments, especially at the edges of secondary structure segments. For example, some analysis showed that the percentage of agreement is only 63% between DSSP, P-CURVE and DEFINE (16). So our segmentation results are compared with assignments given by DSSP, DSSP-ACC (solvent accessibility), STRIDE, PSEA, KAKSI and XTLSSTR.

Here is an example:

<table>
<thead>
<tr>
<th>DSSP structure segmentation:</th>
<th>DSSP-ACC structure segmentation:</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVLS EG EWQLVHL HWAKV EAD VAGHGQDLIRLFKS H PETLEK F DRVKH L</td>
<td>MVLS EG EWQLVHL HWAKVE AD VAGHGQDLIRLFKS H PETLEK FDRVKHL</td>
</tr>
<tr>
<td>- - - - HHHHHHHHHHHHHH GGG</td>
<td>CCCC HHHHHHHHHHHHHHH TT HHHHHHHHHHHHHHH C GGGGGG TTTTTT</td>
</tr>
<tr>
<td>MVL SEG E W QL V L H VW AK V EAD VAG HG Q D IL I R LF KS H PE T L EK F DR V KH L</td>
<td>MVLS EG EWQLVHL HWAKVE AD VAGHGQDLIRLFKS H PETLEK FDRVKHL</td>
</tr>
<tr>
<td>E B EEE I B EE B I E BB EE B EEE I I I BB I E BBB I BB EE I EE I EE I EE B EE I</td>
<td>STRIDE structure segmentation:</td>
</tr>
<tr>
<td>MVLS EG EWQLVHL HWAKVE AD VAGHGQDLIRLFKS H PETLEK FDRVKHL</td>
<td></td>
</tr>
<tr>
<td>CCCC HHHHHHHHHHHHHHH TT HHHHHHHHHHHHHHH C GGGGGG TTTTTT</td>
<td></td>
</tr>
<tr>
<td>HDP word segmentation:</td>
<td>Soft-counting word segmentation:</td>
</tr>
<tr>
<td>M V LSE G EW QL VLH VW AK V EA DV AG H G QD IL I R LF K SHP ETL EK FDR VK HL</td>
<td>MV LSEGE WQLV LHVW A KVEAD VAGH GQDL IRLFK SHPET LEKFD R VKHL</td>
</tr>
</tbody>
</table>

Fig. 3 An example of structure segmentation and unsupervised word segmentation. The amino acid sequence is part of Protein 101m in PDB database.

The evaluation results of unsupervised word segmentation are shown in Table 1.

<table>
<thead>
<tr>
<th></th>
<th>HDP method</th>
<th></th>
<th>Soft-counting method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Precision</td>
<td>Recall</td>
<td>F-score</td>
</tr>
<tr>
<td>DSSP</td>
<td>38.7%</td>
<td>56.1%</td>
<td>45.8%</td>
</tr>
<tr>
<td>DSSP-ACC</td>
<td>60.7%</td>
<td>54.1%</td>
<td>57.2%</td>
</tr>
<tr>
<td>STRIDE</td>
<td>29.4%</td>
<td>57.3%</td>
<td>38.8%</td>
</tr>
<tr>
<td>PSEA</td>
<td>34.3%</td>
<td>47.6%</td>
<td>39.9%</td>
</tr>
<tr>
<td>KAKSI</td>
<td>18.1%</td>
<td>62.8%</td>
<td>38.9%</td>
</tr>
<tr>
<td>XTLSSTR</td>
<td>24.3%</td>
<td>40.7%</td>
<td>30.5%</td>
</tr>
</tbody>
</table>
Table 1 shows the consistency between the unsupervised word segmentation and the structure segmentation of DSSP and DSSP-ACC. But it seems that other kinds of structure assignments don’t match the unsupervised word segmentation clearly.

**DISCUSSION**

Most unsupervised segmentation need select the maximal word length. The segmentation effect also greatly replies on this length. If the word length is too long, there will be too many low frequency words. It's difficult to evaluate the parameters of low frequency words. It's the main disadvantage of unsupervised methods.

Here we set 9 as the maximal word length according to Zipf’s law. To align the maximal word length of word segmentation and structure segmentation, we could regard the long secondary structure words are ‘compound words’, which are combinations of short words. These long compound words can be divided into short words. The segmented word sequences constructed through structure segmentation enable the use of the supervised method to segment the long secondary structure words into short sub-words. The frequency of structure segmentation words is used as goodness to re-segment the long secondary structure segments.

The relation of maximal word length and F-score is shown in Fig.4:

![Fig.4. Relation of maximal word length and segmentation F-score (HDP segmentation method, gold-standard is DSSP structure segmentation)](image)

Maximal word length is pre-requisite for soft-counting method. For HDP, it’s not a obligatory requirement. If we don’t set any word length limitations, the F-score is about 37.6% for HDP methods.

As a comparison, the boundary F-scores of unsupervised segmentation methods for Chinese texts
reach about 75%. The F-scores for English text deleting spaces between words are normally more than 70%. For protein sequences, the best F-score we obtained is only about 57%. And most such value is no more than 40%. This is could be explained by low frequency words problem.

For the word only appear several times, their statistical features is not reliable. So such words are normally neglected in unsupervised segmentation. It’s the main shortcoming of unsupervised methods.

In English data set, the top 10% frequency word occurrences normally account for about 90% of all letters in data set. The low frequency word, for example, 1 frequency words account for about 50% of all words in vocabulary, but their occurrences only account for about 1% of all letters in data set. So although unsupervised method can't deal with the low frequency word, it could still get the good segmentation performance.

Then for protein data set, we regard the segments in DSSP structure segmentation as word. One frequency words account for about 50% of all words in vocabulary, but their occurrences account for about 58% of all letters in data set. If we regarded the 20 as the threshold of low frequency word, the occurrences of low frequency words account for about 67% of all letters in protein data set. Such percentage is only about 17% for English corpus. If we set the maximal word length limitation 9 for structure segments, this percentage is still more than 46%.

So in theory, the best F-score is about 33% for unsupervised protein word segmentation. If setting the word length limitation, this value is about 54%. Our experiment results in Table 1 roughly conform to this theory inference.

More low frequency words, lower segmentation F-score. This result could also explain following results:

(1) Reducing maximal word length could improve the segmentation performance. For example, length of 10% of the words in secondary structure segmentation are more than 15. The 15 letters can represent 20^15, about 3.3e+19 distinct words. For 2 letters, there only be 20^2=420 possible words, whose occurrences probability is much more than 15 length words in a same long sequence. So setting a short maximal word length could reduce number of low frequency words and improve the segmentation performance.

(2) HDP method is better than soft-counting. We normally pre-set a maximal word length for soft-counting method. But HDP could automatically select this value. We find the HDP methods select much shorter word length than soft-counting method, which is shown in Fig.5. So its result is better than soft-counting.
Fig.5 Word length distribution in segmentation of HDP, soft-counting, DSSP and DSSP-ACC

(3) High score of DSSP-ACC also lies on its short word length, which is shown in Fig.5.

The low maximal word length could improve the F-score, but the segmentations containing too many short words or letters has no use for most text information processing technologies. So if we want to apply some new text processing technologies in protein sequence analyzing, Soft-counting method is a better choice.

According the experiment above, we could draw these conclusions:

(1) Unsupervised Protein word segmentation is not equal to structure segmentation, but they have close relations.

(2) The long structure word may be regarded as compound word.

(3) Unsupervised segmentation methods could be regarded as an ‘encoding’ process that enables the sequences expresses the most information. This operation is very similar to the Ab initio 3D structure prediction methods, which seek the lowest energy state of sequence. So the protein unsupervised segmentation may give a new kind of explanation about how protein first structure decides its high level structure. For example, Infants have no preliminary knowledge of languages, but can still identify word boundaries in continuous speech. This phenomenon is the tips for some unsupervised segmentation researches. Similar learning operation may also happen in genes system. Some researches analyze the evolution of human languages (17, 18). A language begins with few simple ‘words’. New terms with complex semantics are eventually invented to convey more information (19). This process is also similar to the evaluation of genes.

In the future, the F-score of protein word segmentation may be improved in these directions: First, if new structure alignment methods appears, it could set new gold-standard and may improve this value. Secondly, we can wait more protein sequences and structure data. The effect of unsupervised methods mainly depends on the size of data set. Thirdly, we could also add
‘biological grammar rule’ in unsupervised segmentation methods. The research for human English shows even set the “first letter rule” could clearly improve the unsupervised segmentation results (12).

We know the text information processing systems are normally designed in ‘words’ level, but not letter level. The major hurdle in the use of these technologies in protein sequence analyzing is the identification of functional equivalents of ‘protein words’. The unsupervised segmentation just could build such a dictionary. Although we are far from build an ideal protein dictionary, current result is enough to test these technologies in proteins. After segmenting the protein letter sequence into “word sequence”, we could use almost all text information processing technology, such as search engine, automatic proofreading, machine translation, to analyze protein sequence. As a demo, we have built a protein sequences search engine based on mature open source systems (20).

Furthermore, we discuss why these unsupervised segmentation methods work very well for most human languages text, but not for protein sequences.

The segmentation process could be regarded as an encoding process, replacing the letters of a word with related word symbol. A basic assumption of unsupervised segmentation is that the sequences always apply the most effective encoding method to express the information. And the unsupervised methods try to seek such an encoding method. This assumption is applicable for most human languages text. But it seems not very appropriate for protein sequences. There may be two opposite explanations:

First, current structure segmentation is not efficient. This means we can find a different structure assignment method, whose structure segments could express gene’s functions better.

The ‘Description Length’ is normally applied to evaluate segmentation encoding process (11). A codebook in which each word is represented by a unique string can be used to encode a corpus of words as efficiently as possible. The total number of letters required to encode the corpus (sum of the lengths of the codebook and encoded corpus) using a well-designed codebook/vocabulary would be less than the original corpus. Smaller units, such as morphemes or phonemes, which require fewer code words and thus a shorter codebook, can be encoded further. However, efficiently encoding the corpus becomes more difficult using fewer code words. Meanwhile, some words may never be used when too many words are in the codebook. Thus the length of the codebook and the length of the encoded corpus must be balanced. The Description length principle states that a codebook that leads to the shortest total combined length must be chosen. So ‘Description Length’ can be used to describe segmentation efficiency.

The description length for DSSP segmentation and soft-counting segmentation is shown in Fig.6:
From Fig. 6, we could find the description length value of DSSP structure segmentation is more than the value of soft-counting segmentation methods. In DSSP segmentation, about 45% description length value is used for vocabulary, in soft-counting segmentation, this value is only 16%. Structure segmentation designs a large codebook, but only few of its abilities are used. Thus, secondary structure segmentation maybe not a good encoding method.

Second, the structure segmentation scheme is very efficient, but our data set is unbalanced. Data unbalance phenomena is very common in corpus construction process. For example, if we only select several paragraphs of a long book to build the data set, this problem will appear.

We construct the experimental data by filter the similar protein sequences in PDB. The PDB protein data could also be regarded as randomly selected from all available protein data. If our data set is unbalanced, it means all current protein data is unbalanced or current methods to detect the protein-coding regions are not very comprehensive. If so, there may be more protein sequences in current non-coding regions.

We could also detect coding regions by segmentation methods. We back translate the DSSP secondary structure words into DNA forms and get a DNA vocabulary. Then we could use this vocabulary to segment the DNA sequences based on supervised segmentation methods. For a DNA sequence, if most of its area could be covered by this DNA vocabulary, we think this sequence is likely to be the coding region.

For example, a DNA sequence “ATGGTTTGTGTCTTGAGGGT” is segmented into:

```
ATGGTTTGTGTCTTGAGGGT DNA sequence
MVL S Protein words sequence
```
Fig. 7 Segment DNA sequence based on protein word vocabulary. The green parts could be translated into protein words. The red part can’t be translated.

If part of sequence couldn’t be covered by our protein vocabulary, it will be segmented into DNA letters. We can count the segments whose length larger than 1 to calculate percentage of protein word coding region. In Fig. 7, the DNA sequence length is 21bps. There are 3 protein words ‘’ATGGTTTTG(MVL)’’, ‘’TCT(S)’’ and ‘’GAGGGT(EG)’’. They cover 18 bps. So there are 18/21≈86% protein word coding regions in this sequence.

We run this test for human genome. This genome is divided into 500bps equal length sequences. The test results are shown in Fig. 8:

![Distribution of protein word encoding sequence in Homo sapiens chromosome 1](image)

Fig. 8 Distribution of protein word encoding sequence in Homo sapiens chromosome 1. For example, there are about 75% DNA sequences whose percentage of protein word encoding region lies in [0.6, 0.7].

Here is a typical example of DNA sequence segmentation:

```
TTGGTTG G TTGTCTATTGATGTT TTTGCT C TATTCTAAG A A TTGGAG A GAG A GAGGTT A A AAT C TCT G ACT ATG A TTGTGG A TTGTCT GCTGAT GTT TTTGCT C TATTGTCT AAGAAT TGG A GAG A GAG A GAGGTT A A A A TCT C C G ACT ATG A TTGTGG A TTGTCT ATTGCTTTTTT GGT C TATTGTCT AAGAAT
```

This sequence is part of “gi|157704448|ref|AC_000133.1| Homo sapiens chromosome 1, alternate assembly HuRef, whole genome shotgun sequence:307501-308000”. The green segments could be translated into protein words. Although its 87% region is protein word encoding region, such sequence is normally regarded as no-coding sequence.

For a DNA sequence, if its protein word encoding region is more than 70%, we regard this sequence as encoding sequence. Then in human genome, there will be 16% sequences are encoding sequences. This speculation is quite different with the recognized conclusion.

Although adding these protein word encoding sequences could improve the data unbalanced...
problem and make the secondary structure segmentation seems more reasonable, these sequences may have no clear biological meanings. So we will discuss the relations between protein word segmentation and gene finding problem in the future.

MATERIALS AND METHODS

Protein Structure Data

Unsupervised segmenting methods only need raw letter sequence. We mainly use the data of PDB (http://www.rcsb.org/pdb/) as our experiment data. This dataset contain about 100,000 pieces of protein sequences. We use CD-HIT algorithms to delete the similar protein sequence. Its codes could be found in: http://weizhong-lab.ucsd.edu/cd-hit/download.php. We also use some protein sequence data of website “uniprot.org” (http://www.uniprot.org/downloads), which is a central repository of amino acid sequence. We select the UniProtKB/Swiss-Prot and UniProtKB/TrEMBL.

Source code

All source codes and experiment instructions of this paper could be found in: https://github.com/maris205/secondary_structure_detection

Maximal word length evaluation

We could use “zipf’s laws” to evaluate the length of words. The “zipf’s laws” states, in a long enough document, about 50% words only occur once. These words are called ‘Hapax legomenon’.

Because we have no any structure information, we could construct words by intersecting segmenting the amino acid sequences and calculate the percentage of ‘Hapax legomenon’. For instance, the sequence “AAAQL”, assume the maximal word length is 2. All there intersecting segmentation is A, A, A, Q, L, AA, AA, RQ, RL. There are 6 different words, AA,AC,CG, Q, L,RQ and RL appears once, which are ‘Hapax legomenon’. So there are 4 ‘Hapax legomenon’. Its percentage is 4/6, about 66%. If for a maximal word length, the percentage of “Hapax legomenon” near 50%. This length could be regarded as word length of most word in data set. The maximal word length could be set according to this length. For our protein sequence, the relation of word length ‘n’ of intersecting segmentation and the percentage of ‘Hapax legomenon’ is show in Fig.9:
In Fig.9, we find 50% line of ‘Hapax legomenon’ near word length 6. So we set the 9 as the maximal word length for unsupervised methods. If we have more corpus, we could set a longer word length.

**Unsupervised segmentation methods (soft-counting)**

Most unsupervised methods are designed based on Expectation Maximization (EM) method, an iterative method for finding maximum likelihood estimates of parameters in statistical models. For soft-counting method, we could evaluate the probabilities of all possible words.

N-multigram language model is a typical model for EM based methods, here the n is the maximal word length. This model assumes the likelihood of a sequence is the sum of likelihoods of its all forms of segmentations. For example, a sequence ‘AATD’, assume the maximal word length is 3. Then its likelihood:

\[
L(AATD) = \sum p(A) p(A) p(T) p(D) + p(A) p(A) p(TD) + p(A) p(ATD) p(D) + p(A) p AT p(D) + p(AA) p(T) p(D) + p(AA) p(TD) + p(AAT) p(D) + p(A) p(ATD)
\]

Then for EM method:

In initial step, it gives the random or uniform initial probability for all words firstly. Here we set
probability as 0.1 for all the words.

For E-step, we calculate the probability of all forms of segmentation. For example above, in the first round, the probability of every word is 0.1, so:

\[ p(A)p(A)p(T)p(D) = 0.1 \times 0.1 \times 0.1 \times 0.1 = 0.0001 \]

\[ p(AA)p(T)p(D) = 0.1 \times 0.1 \times 0.1 = 0.001 \]

......

The sum of all probabilities is 0.0331, which is the sequence probability. We should normalize this value to 1. So the probability of each form of segmentation becomes:

\[ p(A)p(A)p(T)p(D) = \frac{0.0001}{0.0331} = 0.003 \]

\[ p(AA)p(T)p(D) = \frac{0.001}{0.0331} = 0.03 \]

......

For M-step, we evaluate the words probability according to segmentation forms above.

For \( p(A)p(A)p(T)p(D) = 0.003 \), every word ‘A’, ‘A’, ‘T’, ‘D’ in this segmentation get the probability of 0.0001.

For \( p(AA)p(T)p(D) = 0.03 \), every word ‘AA’, ‘T’, ‘D’ gets probability of 0.001.

Every segmentation forms all assign its probability to its every words. Then for each word, we add its probability together. For word ‘A’, its probabilities sum is 0.3988, the ‘AA’ is 0.3323, etc.

We do the same thing for every sequence in test data. Then for each word, we add its probabilities together as the probability of this word. We should also normalize the probabilities of all words to ensure the sum of all words probabilities is 1.

We repeat the E-step and M-step until the probability of words don’t change anymore. Finally, we could get a vocabulary with word probability.

After obtaining a list of word candidates each associated with a goodness like probability:

\[ W = \{ (\omega_i, g(\omega_i)) \}_{i=1,K,a} \]

, here \( \omega_i \) is a word candidate and \( g(\omega_i) \) is a goodness function.

To find the optimal segmentation of a given letter sequence, a Viterbi-style one to search for the best segmentation \( S^* \) for a text \( T \):

\[ S^* = \arg \max_{\omega_1, \omega_2, \ldots, \omega_n \in T \{ \omega_i, g(\omega_i) \} \in W} \]
Supervised segmentation methods

Because we have obtained the segmented protein word sequence, we could simply count the occurrence of each word and use frequency as goodness to design the Viterbi-style segmentation method.

Description length

Description length is the empirical description length of a corpus in bits that can be estimated by the Shannon-Fano code or Huffman code as below, following classic information theory. It can be formulated in terms of token counts in the corpus as below for empirical calculation:

The description length of a corpus \( X = x_1 x_2 \ldots x_n \), a sequence of tokens (e.g., letters, words), its description length:

\[
DL(X) = -n \sum_{x \in V} p(x) \log p(x) = - \sum_{x \in V} c(x) \log \frac{c(x)}{|X|}
\]

Where \( V \) is the set of distinct tokens in \( X \) and \( c(x) \) is the count of \( x \) in \( X \).

For segmentation method, its description is:

\[
DL(X[r \rightarrow s] \oplus s), \text{ where } r \text{ is an index, } X[r \rightarrow s] \text{ represents the resultant corpus by the operation of replacing all occurrences of } s \text{ with } r \text{ through out } X \text{ and } \oplus \text{ represents the concatenation of two things.}
\]

References and Notes


**Reviewer reports:**

Reviewer #1: The manuscript describes the application of a method used in a search engine to solve problems of computational biology. It is the primary aim of the authors to use this unsupervised method of text segmentation to predict the 2D structure of proteins.

In many cases computational biology utilizes more general methods of computer science. Thus, it is reasonable to consider novel algorithms to solve problems of biology. It is obvious that a
segmentation method will create segments. This means that simply applying a method is not sufficient, the authors have to convince the reader that their method creates interesting and useful results in the application domain.

The authors create a segmentation of proteins sequences and assess the agreement of their segmentation with those ones determined by several methods like DSSP, KAKSI, etc. As a quality measure, an F-score is computed. However, it is not clear whether a score of 45.8% is a good or a bad performance because the authors do not assess state-of-the-art methods for the prediction of 2D structure like PHD or PSIPRED. For a comparison, this data are necessary. Moreover, the "meaning" of the symbols in the output of DSSP and the other programs that deduce 2D structure from the known 3D structure is different and therefore it would be necessary to weight the positive and negative cases according to their "meaning". For example, residues belonging to beta-strands are characterized with different symbols, to indicate subtle environmental differences; see Fig. 3. This is why programs that predict 2D structure from sequences usually distinguish only three 2D elements, which are alpha-helices, beta-strands, and loops. As the performance values of these programs indicate, it is difficult enough to precisely predict such a crude segmentation and a more detailed one is beyond current in silico methods. If the authors want to improve their performance measure, I would recommend to study the classical literature which discusses adequate methods, see e.g. the paper of Rost and Sander (1993).

This method creates "some" segments of proteins sequences. However, what is there biological meaning? Is it useful with respect to the function of a protein? The authors only show a single example, which does not make plausible that the segmentation helps to understand a biologically relevant problem. In Fig. 3, DSSP and STRIDE deduced for the studied protein two alpha-helices connected by a short loop. Both methods utilized by the authors predict at least eight segments. DSSP-ACC additionally indicates, which residues are solvent exposed. However, I do not see a correlation of the authors' segmentation and these more detailed 2D data. So, these segments seem to be chosen arbitrarily.

Moreover, homologous proteins from phylogenetically unrelated species are encoded by quite different protein sequences. This means, that most proteins, that have the same 3D structure are represented by many sequences that can differ in 60% or more of their residues. Supervised state-of-the-art methods of 2D prediction reach their performance only, if a large multiple sequence alignment can be analyzed to deduce "common themes" of amino acid usage for each residue position. What is the outcome of the authors' approach if different sequences that encode the same protein are analyzed? Is the segmentation robust with respect to sequence variations? If not, what is then the BIOLOGICAL meaning of differing segmentations?

This variation of such "words" representing the same 2D element makes intractable the improvements proposed by the authors: How to generate a corpus of words or a vocabulary in this context? I feel that this is not possible. The authors should study databases like PFAM to get an impression of sequence variability in homologs. In proteins, "words", e.g 2D elements with the same "meaning" can differ dramatically, which distinguishes this language from a spoken one,
which does not allow for this variability.

Moreover, the proposed segmentation of DNA is highly questionable. It seems to me that the authors are not aware of the existence of codons, introns, and exons. Otherwise they would NOT propose the coding of a protein in a DNA sequence consisting of short pieces (multiples of trimers, i.e. codons) separated by single nucleotides. Genes are NOT organized in this manner. I would recommend to calculate the probability of finding in the human genome randomly chosen words separated in the same way as proposed by the authors. I expect that the probability for finding them is high.

So in summary, this manuscript describes the application of a novel method which generates "some" results. However, the authors have not shown that the application of this method advances the understanding of proteins or DNA. Therefore, I cannot recommend to publish this manuscript.

Reviewer #2: In this manuscript, the authors attempt to use word segmentation methods from natural language processing for predicting protein secondary structures ab initio, i.e., without having known pairs of sequence and secondary structure available for training. As far as I can tell, the authors do not contribute any new methodology, but compare different existing methods originally designed for different tasks on a new application domain, and it appears to me that the manuscript does in the end not predict protein secondary structures at all but merely segments protein on a sequence level.

In any case, I do not recommend this manuscript for publication as it already lacks the basic requirements of a sound report of a scientific study.

Below there is a -- certainly not complete -- list of major flaws.

Presentation:
(1) The manuscript is very hard to read, as the language would clearly benefit from professional copy-editing. Moreover, the structure of the manuscript is poor as paragraphs are only loosely connected, the authors jump between different topics without clear motivation, and a straight storyline is not recognizable. The "Results" section is only one page long, whereas many empirical results actually appear within "Discussion". If it's impossible first to present all results, then perhaps both sections should be combined into one logical unit. Also subsections could be beneficial.

(2) New terms are rarely properly defined and introduced so a lot of guesswork is needed to follow the argumentation. Example: At the beginning of the Results section, the authors refer to algorithms called "DSSP, STRIDE, etc". What do the abbreviations mean? What are the algorithms doing? Precise references are here *mandatory*, as well as a brief description of the features of he individual methods.
(3) The authors present concepts rarely formally in an abstract manner, but chose to predominantly rely on examples. While examples are always a good addition to further illustrate a concept, they can never be a replacement for formal definitions and derivations, as it is impossible for a reader to understand the general case from a single example.

Content:

(1) The key idea, i.e., perceiving prediction of protein secondary structures perceived as a segmentation problem, is not motivated. Is it really a problem that existing algorithms require training data? I would expect that nowadays a large number of associations between protein sequences and secondary structures are available, and choosing to ignore this information appears to me questionable. Under which conditions may a ab initio method be actually useful? Related to that, I do not see substantial empirical evidence that the unsupervised approach could be meaningful. What need to be shown is at least a few comparisons of the unsupervised approach versus the best supervised method existing to date on practical relevant prediction problems that demonstrate that at least in some cases the unsupervised approach yields a similar performance or a performance loss that is still in acceptable range.

(2) The study that translates the protein segmentation back to the DNA sequence level appears to me fundamentally flawed. The presented example on page 11 seems to neglect the existing of an open reading frame. *IF* the first fragment (translated to MVL) was actually protein coding, then the second (S) cannot be. Hence, all following discussion and the drawn conclusions (more protein coding sequences in regions to be considered noncoding) appear to me wrong.

(3) It is not acceptable to use two different methods (HDP+soft counting), but to describe in the Methods part only one -- especially if it is the more simple one. *Either* both methods are considered to be known from literature, then appropriate references and a good intuitive explanation may suffice. *Or* both methods are not expected to be known by the target audience, but then they have to be presented equally. If the description would be too lengthy or technical for the main manuscript, there is the possibility to use a supplement.

(4) Related to the previous comment, I find the description of the EM algorithm not helpful, as it solely relies on one example. What needs to be done in order to present it properly: (i) Defining the observed variables, (ii) defining the unobserved/hidden variables, (iii) defining the model parameters, (iv) writing down the likelihood in the general case, (v) stating that EM methodology is applied, and (vi) presenting the final results of the derivation of E step (expected sufficient statistics of unobserved variables) and M step (parameter estimation based on weighted data).