DNA Sequence Classification in the Presence of Sequencing Errors Project of CSE847

Yuan Zhang ZHANGY72@MSU.EDU
Cheng Yuan CHENGY@MSU.EDU
Computer Science and Engineering Department, Michigan State University, East Lansing, MI, 48823, USA

Abstract

We propose a new method to solve the problem of DNA sequence classification in the presence of sequence errors. This method incorporates sequencing error models into standard Viterbi Algorithm(AJ, 1967) which is used to find the best path that a query DNA sequence is generated by a profile Hidden Markov Model (profile HMM)(SR., 1998). Our method will correct the sequencing errors and then align the corrected query sequence to different protein families. By sequencing error correction our method will improve the prediction accuracy for sequence classification. Moreover, we will classify more sequences to one of the candidate families which cannot be classified by current tools due to sequencing errors.

1. Introduction

Biological sequence classification is an important problem which can help biological researchers to identify regions of similarity between different sequences, predict the function of a protein motif, assign the family label to a novel sequence and dig into the evolution history of a certain biological specie or family. Next-generation sequencing technologies have made it possible to generate large amounts of biological sequence data at low cost, opening great opportunities in life sciences. As genomes are sequenced, a major challenge is their annotation - the identification of genes and regulatory elements, their locations and their functions.

However, DNA sequences generated by the next generation sequencing techniques such as the high throughout and low cost 454 sequencing machine have many sequencing errors such as insertions or deletions of nucleotide bases. This may cause frameshifts when these DNA sequences are translated into protein sequences. In this case, traditional probabilistic generative model can not correctly predict the classification of this translated protein sequence because frameshifts will change the amino acids in the original sequences. Thus an effective and accurate sequence classifier is needed to correct the sequencing errors within a DNA sequence and then classify it to the correct family. In order to better explain our proposed method, we will first introduce some background concepts in bioinformatics.

1.1. DNA Translation

In genetics, DNA translation is the first stage of protein biosynthesis (part of the overall process of gene expression). In translation, messenger RNA (mRNA) produced in transcription is decoded to produce a specific amino acid chain, or polypeptide, that will later fold into an active protein. A DNA sequence is composed of 4 nucleotides which are represented by an alphabet of A, C, G and T. A nucleotide is also called a DNA base. Protein sequences are composed of 20 kinds of amino acids which are represented by an alphabet of 20 Roman letters. Through DNA to protein translation, every three consecutive DNA bases will be translated to one amino acid. And these three bases are called a codon. This translation is governed by a table called codon table. Fig 1 is a typical codon table.

Note that in the codon table there are three special combinations of DNA bases which result in stop signal in DNA translation. We call them stop codons.

When a DNA sequence is translated to a protein sequence, three-frame translation is often used by researchers because in most cases the query DNA sequence is just part of a complete genome. Three-frame translation means a DNA sequence will be translated into three different protein sequences starting from the first, second and the third base of the DNA sequence.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

	T	C	A	G
T	TTT Phe F Phenylalanine	TCT Ser S Serine	TAT Tyr Y Tyrosine	TGT Cys C Cysteine
	TTC Phe F Phenylalanine	TCC Ser S Serine	TAC Tyr Y Tyrosine	TGC Cys C Cysteine
	TTA Leu L Leucine	TCA Ser S Serine	TAA Ochre (*Stop*)	TGA Opal (*Stop*)
	TTG Leu L Leucine	TCG Ser S Serine	TAG Amber (*Stop*)	TGG Trp W Tryptophan
C	CTT Leu L Leucine	CCT Pro P Proline	CAT His H Histidine	CGT Arg R Arginine
	CTC Leu L Leucine	CCC Pro P Proline	CAC His H Histidine	CGC Arg R Arginine
	CTA Leu L Leucine	CCA Pro P Proline	CAA Gln Q Glutamine	CGA Arg R Arginine
	CTG Leu L Leucine	CCG Pro P Proline	CAG Gln Q Glutamine	CGG Arg R Arginine
A	ATT Ile I Isoleucine	ACT Thr T Threonine	AAT Asn N Asparagine	AGT Ser S Serine
	ATC Ile I Isoleucine	ACC Thr T Threonine	AAC Asn N Asparagine	AGC Ser S Serine
	ATA Ile I Isoleucine	ACA Thr T Threonine	AAA Lys K Lysine	AGA Arg R Arginine
	ATG Met M Methionine, *Start*	ACG Thr T Threonine	AAG Lys K Lysine	AGG Arg R Arginine
G	GTT Val V Valine	GCT Ala A Alanine	GAT Asp D Aspartic acid	GGT Gly G Glycine
	GTC Val V Valine	GCC Ala A Alanine	GAC Asp D Aspartic acid	GGC Gly G Glycine
	GTA Val V Valine	GCA Ala A Alanine	GAA Glu E Glutamic acid	GGA Gly G Glycine
	GTG Val V Valine	GCG Ala A Alanine	GAG Glu E Glutamic acid	GGG Gly G Glycine

Figure 1. Codon table

For example, a query DNA sequence is as follows:
AGCCTGTCTAGAGGTaAACGGCTGTAGCTGA
In frame-1 translation, the DNA sequence will be translated this way:
DNA sequence:
AGC CTG TCT AGA GGT AAC GGC TGT AGC TGA
Protein sequence:
S L S R G N G C S Stop

In frame-2 translation, the DNA sequence will be translated this way:
DNA sequence:
GCC TGT CTA GAG GTA ACG GCT GTA GCT
Protein sequence:
A C L E V T A V A

In frame-3translation, the DNA sequence will be translated this way:
DNA sequence:
GCC TGT CTA GAG GTA ACG GCT GTA GCT
Protein sequence:
P V Stop R Stop R L Stop L

1.2. Sequencing Errors and Frameshift

Although the next generation sequencing techniques such as the 454 sequencing machine have high throughout and low cost there are many sequencing errors such as insertions or deletions of DNA bases within the sequences they generate. An insertion means that when a DNA sequence is generated the sequencing machines add one more base which should not exist in the gene of the target biological family. In order to correctly classify this DNA sequence to the family, the insertion should be located and deleted. A deletion means that a base which should exist in the genome is deleted due to sequencing errors. Researchers have shown that sequencing errors are more likely to occur in the homopolymer regions which contain multiple simultaneous copies of a single base (A, C, G or T).

When a sequencing error such as an insertion or a deletion occurs, it will cause frameshift, which means shift in the reading frames or the grouping of the codons. This will result in a completely different translation from the original. Although the redundancy brought by three-frame translation can avoid the false translation of all the DNA sequences. For example, if the first base is an insertion, frame-2 translation will result in a decent translation result because only the first amino acid is wrongly translated. However, a sequencing error which exists near the middle of the DNA sequence may result in a short sequence alignment because at most half of the DNA sequence will be correctly translated no matter what frame we use to do the translation. If this short alignment has score lower than the threshold, this sequence will be misclassified. Here is an example of frameshift (the lowercase letter means an insertion):

DNA sequence: AGCCTGTCTAGAGGTaAACGGCTGTAGCTGA
Frame-1 translation: S L S R G K R L Stop L
Frame-2 translation: A C L E V N G C S Stop
Frame-3 translation: P V Stop R Stop T A V A

Compared to the original three-frame translation results, the insertion results in changes in the translation. This will potentially be the reason of some prediction errors in the DNA sequence classification.

2. Related Works

There are a lot of algorithms and tools which can be used to do sequence homology search and classification.

2.1. Pairwise Sequence Alignment

Pairwise sequence alignment is a way of arranging the sequences of DNA, RNA, or protein to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences. A two dimensional score matrix is used to assign a score to every pair of aligned residues. This score is the log-odds ratio of the probability that these two sequences belong to the same family over that if they are random noise. Then the two sequences will

be aligned using the Naïve Bayes rule that the log-odds ratio of the probability that these two sequences are of the same family over that these two sequences are non-homologous is the sum of the scores of all residues in the sequence. Needleman-Wunsch algorithm(Needleman, 1970) will be used to find the best alignment score. If this score exceeds some pre-defined threshold, then we can say that these two sequences are homologous. However, this method does not work well in the problem of sequence classification. In order to decide whether a query sequence belongs to a family or not, we have to align the query sequence to each sequence of the family in the training set and combine all of the alignment results. It is not effective compared to other methods such as profile HMM.

2.2. Support Vector Machine

Support Vector Machine (SVM) is also used to do sequence classification in bioinformatics. Geometrically, the SVM modeling algorithm works by constructing a separating hyperplane with the maximal margin. Compared with other standard classifiers, SVM is more accurate on moderately imbalanced data. However, in bioinformatics, the datasets are usually greatly unbalanced(N. Japkowicz, 2002). For example, in DNA sequence classification. There are often much fewer sequences which belong to a certain biological family than those which do not. An SVM classifier can be sensitive to high class unbalance, resulting in a drop in the classification performance on the positive class. It is prone to generate a classifier that has a strong bias towards the majority class, resulting in a large number of false negative (R. Akbani, 2002).

2.3. Profile Hidden Markov Model

HMMs are useful probabilistic models to parameterize complex position-specific models, to integrate multiple sources of information consistently, and to frame the homology detection problem more formally and powerfully as a statistical inference problem with log-likelihood statistics. It is based on the following hypotheses: is a target sequence x more likely to be a homolog of our query sequence (or query alignment) y (call this hypothesis H_y) or is x more likely to be a nonhomologous "random" sequence (call this hypothesis R, our null hypothesis)? Theory says if this is the problem, we should aim to calculate a log-odds likelihood score:

$$S(x|y) = log\frac{P(x|H_y)}{P(x|R)}$$

Profile HMM is proposed to align a query sequence to a family. The family is represented by a multiple sequence alignment (MSA) which aligns all the training sequences which belong to the family. A profile HMM is built on the MSA using maximum likelihood estimation. Profile HMM is a special HMM in that it models three states called match state, insertion state and deletion state, for every position in MSA. Every state has an emission probability distribution for the symbols of the alphabet and a transition probability distribution to all the states. All these parameters vary from position to position depending on the distributions of the observed symbols in the training MSA. Viterbi algorithm is used to calculate the probability that a query sequence is generated by the profile HMM. This is also the probability this sequence belongs to the family which the profile HMM is based on.

HMMER3 is a fast and accurate tool based on profile HMM. Compared to BLAST, FASTA, and other sequence alignment and database search tools based on older scoring methodology, HMMER aims to be significantly more accurate and more able to detect remote homologous because of the strength of its underlying mathematical models. The outstanding performance of HMMER3 shows the advantages of profile HMM to do sequence homology search and sequence classification. However, HMMER3 cannot align a DNA sequence to a protein family. Also it cannot handle sequencing errors brought by new sequencing technology such as 454 sequencing machine. When sequencing errors occur in a query sequence, frameshifts happen. This will dramatically change the translated protein sequence. HMMER3 will thus align the wrong protein sequence. This problem is even worse when any of the errors exist near the middle of the DNA sequence because three-frame translation can at most have half of the right sequence. This will bring about extreme difficulty for HMMER3 to classify this sequence. Our method has successfully addressed the problem of sequencing errors by incorporating error detection methodology in our algorithm.

3. Methodology

3.1. Sequencing Error Model

In our algorithm we have made used of a sequencing error model which is built on a large number of DNA sequences generated by 454 Titanium sequencing machines. Table 1 shows the error rate for both insertion errors and deletions errors as well for homopolymer regions and non-homopolymer regions.

Table 1.

Error Type \ Region Type	Homopolymer region	Non-homopolymer region
Insertion error	0.0044	0.0007
Deletion error	0.0044	0.0007

3.2. Algorithm

Input: a DNA sequence x and a profile HMM M from Pfam DB.

Output: the optimal alignment between this DNA sequence x and M using the Viterbi algorithm.

Given any three DNA bases $x_{i-2}x_{i-1}x_i$, we use the function $T(x_{i-2}x_{i-1}x_i)$ to denote the translated amino acid.

Define three functions:

- $V_j^M(i)$ is the score of the best path matching subsequence $x_{1..i}$ to the submodel up to state j, ending with x_i being the third base of a codon, which is emitted by state M_j.
- $V_j^I(i)$ is the score of the best path ending in x_i and $T(x_{i-2}x_{i-1}x_i)$ is emitted by I_j.
- $V_j^G(i)$ is the score of the best path ending in x_i being emitted by state G_j.
- $V_j^D(i)$ is the score of the best path ending in state D_j.
- $insert(x_i)$ = the probability that base x_i is an inserted base.
- $delete(x_i)$ = the probability that there is a deletion after base x_i.

$$
\begin{aligned}
V_j^M(i) = max\{ & \\
& e_{M_j}(T(x_{i-2}x_{i-1}x_i)) \times V_{j-1}^M(i-3) \times a_{M_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-2}x_{i-1}x_i)) \times V_{j-1}^I(i-3) \times a_{I_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-2}x_{i-1}x_i)) \times V_{j-1}^D(i-3) \times a_{D_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-2}x_{i-1}x_i)) \times V_{j-1}^G(i-3) \times a_{G_{j-1}M_j}, \\
& \\
& e_{M_j}(T(x_{i-3}x_{i-2}x_i)) \times insert(x_{i-1}) \times V_{j-1}^M(i-4) \times a_{M_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-3}x_{i-2}x_i)) \times insert(x_{i-1}) \times V_{j-1}^I(i-4) \times a_{I_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-3}x_{i-2}x_i)) \times insert(x_{i-1}) \times V_{j-1}^D(i-4) \times a_{D_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-3}x_{i-2}x_i)) \times insert(x_{i-1}) \times V_{j-1}^G(i-4) \times a_{G_{j-1}M_j}, \\
& \\
& e_{M_j}(T(x_{i-3}x_{i-1}x_i)) \times insert(x_{i-2}) \times V_{j-1}^M(i-4) \times a_{M_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-3}x_{i-1}x_i)) \times insert(x_{i-2}) \times V_{j-1}^I(i-4) \times a_{I_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-3}x_{i-1}x_i)) \times insert(x_{i-2}) \times V_{j-1}^D(i-4) \times a_{D_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-3}x_{i-1}x_i)) \times insert(x_{i-2}) \times V_{j-1}^G(i-4) \times a_{G_{j-1}M_j}, \\
& \\
& e_{M_j}(T(x_{i-2} * x_i)) \times delete(x_{i-1}) \times V_{j-1}^M(i-3) \times a_{M_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-2} * x_i)) \times delete(x_{i-1}) \times V_{j-1}^I(i-3) \times a_{I_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-2} * x_i)) \times delete(x_{i-1}) \times V_{j-1}^D(i-3) \times a_{D_{j-1}M_j}, \\
& e_{M_j}(T(x_{i-2} * x_i)) \times delete(x_{i-1}) \times V_{j-1}^G(i-3) \times a_{G_{j-1}M_j}, \\
& \\
& \textit{Repeat the above three equations for} \\
& \textit{both } e_{M_j}(T(*x_{i-1}x_i)) \textit{and } e_{M_j}(T(x_{i-1} * x_i)) \\
& \}
\end{aligned}
$$

$$
\begin{aligned}
V_j^I(i) = max\{ & e_{I_j}(T(x_{i-2}x_{i-1}x_i)) \times V_j^M(i-3) \times a_{M_jI_j}, \\
& e_{I_j}(T(x_{i-2}x_{i-1}x_i)) \times V_j^I(i-3) \times a_{I_jI_j}\}
\end{aligned}
$$

$$
\begin{aligned}
V_j^D(i) = max\{ & V_{j-1}^M(i) \times a_{M_{j-1}D_j}, \\
& V_{j-1}^D(i) \times a_{D_{j-1}D_j}\}
\end{aligned}
$$

$$
\begin{aligned}
V_j^G(i) = max\{ & insert(x_i) \times V_j^M(i-1) \times a_{M_jG_j}, \\
& insert(x_i) \times V_j^G(i-1) \times a_{G_jG_j}\}
\end{aligned}
$$

4. Experiments

4.1. Experimental Dataset Description

We used two datasets to test performance of our method. The first dataset is from NifH family which is involved in biological fixation of nitrogen to assimilable ammonia. A set of training protein sequences is provided to build the profile HMM. A set of testing DNA sequences is used to predict how many sequences belong to NifH family, which is of great biological significance because the knowledge of genes which are involved in nitrogen fixation is important to agriculture research. These DNA sequences are generated by 454 sequencing machines. We compare the performance of

Table 2.

Method	# of seq	# of classified seq	Classification accuracy	# of seq with errors	# of seq with predicted errors	Error prediction accuracy
Genewise	3997	3689	92.29%	455	197	43.30%
Our Method	3997	3861	96.60%	455	438	96.26%

Table 3.

Method	# of seq	# of classified seq	Percentage of classified seq
Genewise	2486	1293	42.00%
Our Method	2486	1110	52.01%

our method with that of Genewise which is based on pairwise alignment on the first dataset.

The second dataset is from bacteria families obtained from soil sample. The training proteins are from three known protein groups with labels. The testing DNA sequences are generated by 454 machines targeted to bacterial genes from the same environment. Thus, error may be present in the testing DNA sequences. We compare our method with HMMER3 on this dataset. Although these two methods use the same theoretical models, which are both based on profile HMM, our method can locate and correct sequencing errors as well as classify more sequences.

4.2. Experimental Results

In the first experiment, all the testing sequences are known to be member of the NifH family. However, due to the presence of the sequencing errors of 454 sequencing machine, it is difficult to classify all sequences. Table 2 shows the comparison between Genewise and our method in terms of classification accuracy and error prediction accuracy.

In the second experiment, the testing sequences either belong to one of the three families or do not belong to any of them at all. It is also possible that some sequences which belong to one of the families cannot be classified due to serious sequencing errors. Thus, we do not know the actual prediction accuracy. However, if the alignment score exceeds some predefined threshold, we have a high confidence to classify the query sequence to the family. The higher the alignment score is, the more confidence we have to classify this sequence. So in this set of experiment, our comparison focuses on the alignment score as well as how many sequences can be aligned. Table 3 shows the comparison between our method and HMMER3.

HMMER3 can classify 42.00% sequences to one of the three protein groups. Our method can classify 52.01% sequences to one of the three protein groups. This shows that by error correction, we can find 10.01% more sequences which cannot be found by HMMER3. Among the 42.00% sequences both methods can classify, our method has significantly longer average alignment length than HMMER3. To be specific, in all the sequences both methods can classify, our method reports errors in 36% of them and we have longer alignment length and higher alignment score than HMMER3. For those sequences our method does not report errors, we have the same alignment results as HMMER3.

5. Conclusion and Future Work

In this project we show that sequencing errors will cause serious problems to the classification of DNA sequences to protein families, especially when the errors are near the middle of the DNA sequences. We have proposed our algorithm which incorporates sequencing error detection into the well known Viterbi algorithm. Two sets of experiments have been conducted and the experimental results show that by error correction, our method not only obtains longer sequence alignment and higher alignment score but also classifies more sequences which cannot be found by neither Genewise nor HMMER3.

Our future work will focus on two aspects. For one thing, although we have aligned more sequences than Genewise and HMMER3, the improvement is not very significant. We will build more accurate sequencing error models to enhance the performance of our program. For the other, sometimes our method will try to correct sequencing errors which do not actually exist within the query sequence. This is because we always choose the best path a sequence is generated by the profile HMM. However, the latent DNA sequence may not necessarily be the best one because our method sometimes tries to tailor the sequence to match the model. This will increase our false positive rate. More efforts have to be made to make our algorithm more robust.

References

AJ, V. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory*.

C. Yan, D. Dobbs, V. H., & Dobbs, D. (2004). A two-stage classifier for identification of proteinprotein interface residues. bioinfromatics. *Bioinfromatics* (pp. 371–378).

Krogh A, Brown M, M. I. S. K., & D, H. (1994). Hidden markov models in computational biology. applications to protein modeling. *J Mol Biol*, *5*, 1501–31.

Mardis, E. R. (2008). The impact of next-generation sequencing technology on genetics. *Trends in Genetics*, *24*, 133–141.

N. Japkowicz, S. S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis* (p. 429C449).

Needleman, Saul, W. C. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, *58*, 443–453.

R. Akbani, S. Kwek, N. J. (2002). Applying support vector machines to imbalanced datasets. *Conference on Machine Learning* (pp. 429–449).

R. Durbin, S. R. Eddy, A. K., & Mitchison, G. J. (1998). *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*. Cambridge UK: Cambridge University Press.

SR., E. (1998). Profile hidden markov models. *Bioinformatics*, *15*, 755–63.

Yuchun Tang, Y.-Q. Z. (2007). Svms modeling for highly imbalanced classification. *Journal of Latex class file*, *1*.