

# Back-translation Using First Order Hidden Markov Models

Lauren Lajos

## Abstract

A Hidden Markov Model (HMM) is a well-studied, statistical model which, when given a sequence consisting of observable states, is used to try to estimate a sequence of hidden, or unknown, states. In addition to its extensive, theoretical mathematical role, such a model has real-world applications in a range of topics including speech recognition, financial modeling (e.g. stock market predicting), environmental studies (e.g. earthquake predictions, weather predictions, etc.), and behavioral studies (e.g. homicides, suicides, etc.) [6]. For the purposes of this paper, we will introduce and apply the HMM to the field of bioinformatics. Specifically, we will look into the use of HMMs, and the manipulation of their required training sets, in an attempt to more accurately back-translate a protein sequence into its original genomic coding strand. Since the focus of this paper lies mainly in training set variations, we have chosen to use a program known as Easyback for our HMM model. The plant *Arabidopsis thaliana* will provide the genomic data needed for our study.

## 1 Introduction

In this paper, we look at the Central Dogma of genetics (specifically, the process of translation and a process we will call "back-translation") and attempt to use a particular mathematical model to aid us in understanding and, perhaps improving upon, the method of back-translation. We will begin with a brief background in genetics, regarding these processes, and an explanation of the problem we face with back-translation. In this section of the paper, we will rely greatly on [4].

The processes that a given strand of DNA must undergo in order to form a protein are collectively referred to as the Central Dogma of genetics. Overall, this transformation from DNA to protein involves two major steps: transcription

and translation. Transcription is the process by which DNA is converted into its corresponding mRNA counterpart. Translation is then the conversion of the newly transcribed mRNA strand into a protein; for simplicity and mathematical purposes we will occasionally refer to this process as the function,  $T: \text{mRNA} \rightarrow \text{Protein}$ .

**Definition 1** We will define **back-translation** as the conceptual idea of reversing the translation process (i.e. the process by which an mRNA strand is sequenced, given a particular protein). We will refer to this process as the function,  $B: \text{Protein} \rightarrow \text{mRNA}$ .

Basically, during translation, an mRNA strand is divided into several thousand or even hundred thousand 3-letter subsequences, consisting of any combination of the nucleotides A, G, C, and U. Each of these 3-letter sequences is deemed a codon. Each codon then encodes for a particular amino acid. The chart below shows the relationship between each of the 64 possible codons and the 20 standard amino acids. By reading the chart, we can easily see that the codon, UCA, will always specify the amino acid, Serine (Ser, S).

	U	C	A	G
U	UUU=Phe UUC=Phe UUA=Leu UUG=Leu	UCU=Ser UCC=Ser UCA=Ser UCG=Ser	UAU=Tyr UAC=Tyr UAA=stop UAG=stop	UGU=Cys UGC=Cys UGA=stop UGG=Trp
C	CUU=Leu CUC=Leu CUA=Leu CUG=Leu	CCU=Pro CCC=Pro CCA=Pro CCG=Pro	CAU=His CAC=His CAA=Gln CAG=Gln	CGU=Arg CGC=Arg CGA=Arg CGG=Arg
A	AUU=Ile AUC=Ile AUA=Ile AUG=Met	ACU=Thr ACC=Thr ACA=Thr ACG=Thr	AAU=Asn AAC=Asn AAA=Lys AAG=Lys	AGU=Ser AGC=Ser AGA=Arg AGG=Arg
G	GUU=Val GUC=Val GUA=Val GUG=Val	GCU=Ala GCC=Ala GCA=Ala GCG=Ala	GAU=Asp GAC=Asp GAA=Glu GAG=Glu	GGU=Gly GGC=Gly GGA=Gly GGG=Gly

Let us pause to understand a couple more definitions which will aid in understanding the difficulty of back-translation.  $B: \text{protein} \rightarrow \text{mRNA}$ , as we have defined it, is in essence the inverse function of  $T: \text{mRNA} \rightarrow \text{protein}$ . Unfortunately, this inverse function is nonexistent. For an inverse function (such as  $B$ ) to exist, the original function (in our case,  $T$ ) must meet two requirements: **1**) it must be *one-to-one* and **2**) it must be *onto*.

**Definition 2** A function is **one-to-one** if each element in the domain maps to a unique solution in the range.

**Definition 3** In order for a function to be *onto*, each element in the range must be hit at least once.

In the problem at hand,  $T: \text{mRNA} \rightarrow \text{protein}$  is onto (since each of the 20 standard amino acids is coded for), but not one-to-one.

**Example 1** Consider the amino acid Alanine (Ala, A). Using the chart, we see that the codon GCA specifies for Ala, but so does the codon GCC.

Hence there is no one-to-one correspondence between the mRNA codons and the amino acids they encode. Without the existence of an inverse function,  $B$ , it is then quite difficult to accurately deduce a particular strand of mRNA (and thus, DNA) given a protein sequence. For example, even a short peptide, say 4 amino acids: PRO VAL THR GLY, has 256 possible mRNA strands as its origin.

This is where the role of mathematics comes into play. Though we cannot backtrack to deduce with perfect accuracy the exact mRNA strand which gave rise to a particular peptide, we can attempt to statistically determine which of the possible mRNA strands acts as the most likely predecessor. In [7], it was shown that "the choice of codons for reverse translation can be refined further by taking into account the [amino acids] flanking the [amino acid] of interest in a protein." For example, when studying Ala alone, we find that the probability of the corresponding codon being GCA,  $p(\text{GCA})=.21$ . Similarly,  $p(\text{GCC})=.27$ ,  $p(\text{GCG})=.36$ ,  $p(\text{GCT})=.16$ . Clearly, none of these probabilities is significantly more likely than another; thus our choice in codon would practically be no more accurate than a random guess. However, when studying the peptide sequence Ser-Ala-Ser,  $p(\text{Ser-GCA-Ser})=.164$ ,  $p(\text{Ser-GCC-Ser})=.545$ ,  $p(\text{Ser-GCG-Ser})=.117$ , and  $p(\text{Ser-GCT-Ser})=.174$ . When flanked by Serine, we can thus conclude with a much greater deal of confidence than the original, mere guess that this particular Ala was coded for by the mRNA codon, GCC.

Taking into account these findings, we decided to incorporate the Hidden Markov Model into our attempts at improving back-translation. Hidden Markov Models and our application of them will be discussed in the following section.

## 2 Background

To see the relation between Hidden Markov Models and back-translation, we must first define a Hidden Markov Model in general. For general information on Hidden Markov Models we will depend heavily on [3],[6].

**Definition 4** A *Hidden Markov Model (HMM)* has five major components:

1.  $S=\{s_1, s_2, s_3, \dots, s_N\}$  is the set of  $N$  possible hidden states. A state at a particular time,  $t$ , is typically denoted by  $q_t \in S$ .

2.  $V=\{v_1, v_2, v_3, \dots, v_M\}$  is the set of  $M$  possible observation symbols.  $o_t \in V$  is used to refer to an observation at a particular time  $t$ .
3. An  $N \times N$  state transition probability matrix,  $A$ , s.t.  $a_{ij} = P[q_{t+1} = S_j \mid q_t = S_i]$ , the probability that  $S_j$  is the state at a time  $t+1$  if  $S_i$  was the state at a previous time  $t$ .
4. An  $N \times M$  observation probability matrix,  $B$ , s.t.  $b_{ij} = P[o_t = V_i \mid q_t = S_i]$ , the probability that  $V_i$  is observed at a time  $t$  in a hidden state  $S_i$ .
5. An  $N$  dimensional initial state probability distribution vector,  $\pi$ , s.t.  $\pi_i = P[q_1 = S_i]$ , the probability that  $S_i$  is the initial state.

**Note** : the following three conditions must be satisfied:

1.  $\sum_{j=1}^N a_{ij} = 1, 1 \leq i \leq N$
2.  $\sum_{j=1}^M b_{ij} = 1, 1 \leq i \leq N$
3.  $\sum_{i=1}^N \pi_i = 1$

A given HMM  $\lambda=(A,B,\pi)$  is often applied to solving one of three central problems:

1. **Decoding problem:** given a HMM  $\lambda = (A,B,\pi)$  and an observation sequence  $O=O_1O_2 \dots O_T$  (where each  $O_t$  is an element in  $V$ ), find the most probable corresponding hidden state sequence  $Q=q_1q_2 \dots q_T$ .
2. **Evaluation problem:** given a HMM  $\lambda= (A,B,\pi)$  and observation sequence  $O=O_1O_2 \dots O_T$ , find  $P(O \mid \lambda)$  or the probability that the observation sequence was constructed by the HMM.
3. **Learning Problem:** given a set of observation sequence  $\{O_1O_2 \dots O_n\}$ , determine the HMM  $\lambda=(A,B,\pi)$  that most accurately explains the observation sequences (i.e. find the values of  $\lambda$  which maximize  $P(O \mid \lambda)$ ).

Understanding this definition and its basic applications, we can now manipulate it to fit the problem at hand. Recall that, in back-translation, we are given a sequence of proteins and our goal is to find the most likely corresponding DNA sequence. Here, the known sequence of proteins will correspond to the sequence of observable states,  $O$ , and the DNA (or mRNA) sequence we wish to deduce will correspond to the hidden state sequence  $Q$ . Thus, the problem of back-translation can be related to HMMs and is, in general, a type of decoding problem.

In my research, I luckily came across a study in which a software program, Easyback, was developed for precisely this application. The set up for the HMM used was as follows:

Let  $q$  be an amino acid input sequence with unknown back-translation, let  $T$  be a training set containing many already-known DNA (or mRNA, so long as the content of  $T$  is consistent) sequences for similar organisms, and then

1.  $S = \{s_1, s_2, s_3, \dots, s_{64}\}$ , which is the set of 64 possible codons, will constitute the set of hidden states
2.  $V = \{v_1, v_2, v_3, \dots, v_{20}\}$ , which is the set of 20 standard amino acids, will be the set of observation symbols
3.  $A = 64 \times 64$  matrix where  $a_{ij} = P[q_{t+1} = S_j \mid q_t = S_i]$  for some codons,  $S_i$  and  $S_j$ . Easyback creators define a *transition state* from  $S_i$  to  $S_j$  to be two consecutive amino acids coded by  $S_i$  and  $S_j$ , respectively. In other words, each " $S_i S_j$ " found in  $q$  is a transition state from  $S_i$  to  $S_j$ . They then defined the *transition probability* of  $S_i$  and  $S_j$  by:

$$P = \frac{\# \text{ of occurrences of "S}_i\text{S}_j\text{" in T}}{\# \text{ of occurrences of } S_i \text{ in T s.t. } S_i \text{ is not followed by a stop codon}}$$

4.  $B = 64 \times 20$  matrix where  $b_{ij} = P[o_t = V_i \mid q_t = S_i]$ . Creators named this probability that a codon  $S_i$  generates an amino acid,  $a$ , the *emission probability* and defined it by:

$$P = \frac{\# \text{ "a" coded for by } S_i \text{ in T}}{\# \text{ "a" in T}}$$

In their paper, they explain that for stop codons the emission probability will be zero, which is obvious since stop codons don't code for any amino acids. Even more interesting and something certainly worth noting, is that, based on the general knowledge of codons and the amino acids that they generate (which is summarized in the chart given previously), the majority of the entries in this matrix will in fact be zero. This is because only about 3 or 4 codons code for each particular amino acid, rather than all 64. For example,  $P[Phe_t | q_t = UUA] = 0$  since the codon, UUA, never codes for the amino acid, Phe.

5.  $\pi$  is the initial state vector where  $\pi_i = P[q_1 = S_i]$ . The initial state vector used by the creators of Easyback is not clearly specified, however it is most probable to assume that they used something along the lines of  $\pi$  where  $\pi_i = P[q_1 = S_i] = \frac{1}{64}$  since the likelihood of any one of the 64 possible codons being the first codon in  $q$  is simply 1 in 64 chance.

Easyback provides three solving-strategies to choose from: *simple*, *binary*, and *reliable*. *Simple* uses the ordinary Blast-similarity strategy. *Binary* uses the smallest necessary training set. *Reliable* allows for analysis of forward and posterior probability diagrams to optimize prediction quality; forward diagrams suggests the smallest necessary training set size for a reliable prediction, and

too much oscillation in a posterior graph indicates that low percentages of amino acids have been correctly deduced. For this posterior decoding, Easyback allows for the application of the Viterbi and/or the Forward-Backward algorithm to the model when back-translating  $q$ . Both the Viterbi and the Forward-Backward algorithms are standard algorithms used to solve HMMs.

### 3 Results

For our study of this topic, we chose to focus on the training set used to back-translate a particular protein input sequence. Specifically, our main interest was in discerning the effect that regional variations in the training set had on the accuracy of back-translation. Using the MPICao2010 data center, we downloaded several genomes for various strains of the well-studied plant, *Arabidopsis thaliana* [2]. A simple program was written and implemented to extract the entire protein coding sequence of the the second chromosome of each of the selected strains. We then chose a random protein (the 2nd protein on the 2nd chromosome) from a Spanish strain of *Arabidopsis thaliana*, labeled Pra6, and back-translated it using Easyback with three regionally different training sets: one set from the same region as the input, one set from a drastically different geographical region than the input, and one set consisting of strains from different regions than the input, but falling roughly within the same latitudinal line. The three training sets created were:

- the entire protein encoding mRNA sequence on the 2nd chromosome for each of 5 Spanish strains of *Arabidopsis thaliana*
- the entire protein encoding mRNA sequence on the 2nd chromosome for each of 6 Russian strains of *Arabidopsis thaliana*
- the entire protein encoding mRNA sequence on the 2nd chromosome for each of 6 strains of *Arabidopsis thaliana* from various countries but the same latitudinal region

Though each training set included only 5 to 6 strains or items, rather than the literature’s recommended 85 items, each of our training sets had plenty of information since each item consisted of practically an entire chromosome. In fact, each of our training sets had far more data than the training sets studied by the Easyback group. Their provided training sets had a file size of anywhere from 50 kB to a 100 kB, whereas the files containing our sets required between 13 MB and 15 MB of storage space.

With the vast amount of information in our training sets and the extreme similarity between the input (a particular strain of *Arabidopsis thaliana*) and the various training sets used to back-translate it (composed of different strains

of the same species of plant), we expected a minuscule error rate in our back-translated output in each of the three scenarios. Further, we predicted that the Spanish training set would give the most accurate back-translation of the three sets and the Russian set would give the least accurate.

Surprisingly, our results did not mirror the  $> 70\%$  accuracy that the Easy-back group claimed and, at the very least, we had expected. Rather, each of training sets provided a back-translated mRNA sequence with 25.8% accuracy. Although the results were better than chance and also consistent across training sets, they were entirely sub-par given the amount of information and similarity of input to training set. To clarify, we defined accuracy as :

$$\frac{\# \text{ of precisely back-translated codons in the output sequence}}{\text{total } \# \text{ of codons in the original input sequence}}$$

where *precisely back-translated* means those back-translated codons which exactly match the original codon, by content and order of that content.

After creating two additional training sets, one using the AGC Kinase gene family of *Arabidopsis thaliana* and the other using the Basic Region Leucine Zipper (bZIP) Transcription Factor gene family, we back-translated a randomly chosen kinase gene and bZIP gene using their respective training sets [1],[5]. The kinase training set included 38 items, and the bZIP training set included 73 items. The back-translation results for these two cases were better; the kinase family back-translation was performed with 52 % accuracy and the bZIP back-translation was performed with 63 % accuracy. The discrepancy between these two resulting percentages should have been slightly larger due to the number of items in the training set. Perhaps, in the bZIP case, we simply chose a particular gene which was more distantly related to the other genes of its gene family, thereby giving the somewhat less-than-favorable result.

The drastic improvement when comparing the results found using the first three training sets to the results found using the two, gene family, training sets, lends us to believe that perhaps the huge amounts of information in the first three training sets, in fact, detracted from the accuracy of the overall result. One thought is that, although the strings and strings of sequence information at first appeared to be optimal, the first three training sets contained *too much* information, not specific to the protein being back-translated, causing the large amounts of data used to provide an accurate result to be canceled out, in effect, by the large amounts of unrelated protein sequences. It seems then that, not only must well-predicting training sets use similar species, they must use genes related to the protein being back-translated. From our study, we have seen that although too little information in the training set can hinder accuracy of back-translation (as in the kinase case), using too much information which is less specified can lead to even less accurate results. Finally, the geographical location of the strains of plants used to back-translate a protein in a plant of the same species seemed to have little to no effect on the accuracy of back-translation.

## 4 Future Work

The initial three training sets we created were not as accurate in back-translating as we would have hoped. In the future, perhaps we can look into discerning the annotated genomes for the various strains that were selected, and create training sets with more meaningful, knowingly-related data rather than the general data used here. With additional time and research, we might also look into creating a back-translating HMM of our own. It would be interesting to see if, in addition to the improved accuracy of back-translation using first-order HMMs like Easyback, back-translation accuracy might be improved further using a second-order model. Similar to a first-order model showing that "codon usage is not a property of isolated codons ...[but that] the bases immediately upstream or downstream affect the translation," a second-order model would show whether or not codon usage is dependent on the two codons just previous or just subsequent to the codon being back-translated [7].

## 5 Acknowledgments

I would especially like to thank Dr. Jonathan Kujawa for his patience, and for his contributions to and oversight of this honors research project. I would also like to give special thanks to my father, Jaroslav Lajos, for his insight on the subject of Hidden Markov Models. Lastly, I would like to thank the VerTEX research group members Ore Adekoya, Rhyker Benavidez, Alyssa Leone, and Logan Maingi for their aid in this research.



## References

- [1] Bögre, L., Ökrsz, L., Henriques, R., and Anthony, R. G. (2003) Growth signalling pathways in Arabidopsis and the AGC protein kinases. *Trends Plant Sci.* 8. 424-431. [http://personal.rhul.ac.uk/ujba/110/agc/agc\\_genes1.htm](http://personal.rhul.ac.uk/ujba/110/agc/agc_genes1.htm), The Bögre Lab: <http://personal.rhul.ac.uk/ujba/110/bogrelab.htm>, <http://www.arabidopsis.org/browse/genefamily/AGC.jsp>.
- [2] Cao, J., Schneeberger, K., Ossowski, S., Gnther, T., Bender, S., Fitz, J., Koenig, D., Lanz, C., Stegle, O., Lippert, C., Wang, X., Ott, F., Mller, J., Alonso-Blanco, C., Borgwardt, K., Schmid, K. J., and Weigel, D. Whole-genome sequencing of multiple Arabidopsis thaliana populations. *Nature Genetics* 43, 956963 (2011). <http://1001genomes.org/projects/MPICao2010/index.html>.
- [3] Ferro, A., Giugno, R., Pigola, G., Pulvirenti, A. Pietro, C. D., Purrello, M., Ragusa, M. (2007) Sequence similarity is more relevant than species specificity in probabilistic backtranslation. *BMC Bioinformatics* 8, no. 58.
- [4] Hartwell, L. H., Hood, L., Goldberg, M. L., Reynolds, A.E., Silver, L.M., Veres, R. C. (2008) Genetics: From genes to genomes. 3rd edition. *McGraw Hill*, 255-300.
- [5] Jakoby, M., Weisshaar, B., Droge-Laser, W., Vicente-Carbajosa, J., Tiedemann, J., Kroj, T., Parcy, F. (2002) bZIP Transcription Factors in Arabidopsis. *Trends In Plant Science.* 7: 106-111. Max-Planck-Institut fuer Zuechtungsforschung <http://www.mpiz-koeln.mpg.de/>, <http://www.arabidopsis.org/browse/genefamily/bZIP-Jak.jsp>.
- [6] Lajos, J., George, K. M., Park, N. (2011) A six state HMM model for the S AND P 500 stock market index. *WORLDCOMP '11*.
- [7] Sivaraman, K., Seshasayee, A., Tarawater, P. M., Cole, A. M. (2008) Codon choice in genes depends on flanking sequence information—implications for theoretical reverse translation. *Nucleic Acids Research* 36, no. 3, e16.