**Translate gene sequence into** **Gene ontology terms based on statistical machine translation**

Wang Liang, Zhao KaiYong

Tencent Tech, 100080, P.R. China,

Hong Kong Baptist University, Department of Computer Science, HK, P.R. China

\*To whom correspondence should be addressed. E-mail:wangliang.f@gmail.com. Tel: +86-18611846371

**[Abstract]** This paper presents a novel method to predict gene sequence function based on statistical machine translation. We select some model species’ gene sequences and related function descriptions in Gene ontology terms as parallel corpus. Then we use the phrase based translation model to build the gene translation model. Its BLEU score could reach about 0.5 when neglecting the order of Gene ontology words. Its function prediction performance is still not better than search based methods. But it could directly give the function of gene sequence and is more efficient.

[**Keywords**] GFP, gene function prediction, machine translation

# 1 Introduction

Determining the functions of genes is a central problem in biology. Because there have been many databases store the gene sequence and corresponding functions, almost all functional prediction methods rely on the identification, characterization, and quantification of sequence similarity between the gene of interest and genes in available databases. However, sequence similarity does not ensure identical functions. One sequence may get several similar sequence, but there is still no a mature method to choose among similar genes with different functions. And if there is no similar sequence, we also can’t get their functions.

In These years, many machine learning research are applied to deal these problems. For example, some classification methods like SVM and network based methods are applied to predict gene functions [1]. Besides the sequence feature, structure, evolutionary and other information are also applied for prediction [2,3,4].

These problems also happen in languages translation area. But current machine translation research could ideally avoid these disadvantages by building translation model between two ‘languages’. Based on these work, we may also build the translation model between gene sequence and function description.

As far as we know, there seems still no research about the application of machine translation in gene sequence analyzing. For this application, we only need build the parallel corpus about gene and its description. Here we select gene data of several model species and their function description in Go ontology form to build such corpus [5].

Most current translation systems are statistical machine translation system. It tries to generate translations using statistical methods based on bilingual text corpora. Where such corpora are available, good results can be achieved translating similar texts [6,7]. This method follows a simple rule: words and phrases that share similar statistical properties are considered equivalent. At least, the similar sequence will get the similar translation results. So the application of translation technology in gene function prediction is reasonable.

# 2 Parallel corpus

Here we select Human gene data to build parallel corpus firstly. Normally, to give the unambiguous description for gene’s function, we select Gene ontology. It unifies the representation of gene and gene product attributes across all species. Gene Ontology contains three kinds of ‘words’, cellular component, molecular function and biological process. We don’t make distinction for them.

There are about 45,538 human gene sequences in Gene ontology database. We need clean the corpus to avoid the possible problems in the training pipeline. Here the long sentences (gene length>1000, Go ontology sequence >600) and empty sentences are removed. The obviously mis-aligned (length ratio > 9) sequence are also deleted. Lastly, we get about 20,000 parallel corpus for gene sequence (protein form) and function description sequence (Go ontology form). We also build a mixed data set containing several model species (about 100,000 data) and a big data set (about 1,000,000). Here we use the gene database id to identify one corpus. Some different genes may have the same sequence. Here we think it’s the natural distribution and don’t delete the repeated sequence. All corpuses could be downloaded from the additional material parts of this paper.

The gene sequence is not naturally segmented, but most translation research need words or phrase sequence. So we segment/tokenise DNA sequence by an unsupervised segmentation method [8]. Its main idea is to evaluate the probability of all possible words by EM or other statistical methods. Then for a sequence, it selects a maximal probability segmenting form as the segmentation of this sequence.

An example of parallel corpus is shown as follows:

Gene sequence (protein):

***MTMDKS ELVQKA KLAEQA ERYDDM AAA MKAVTE QGH ELSNEE RNLLSV AYKNVV***

Function description (Gene ontology term Id):

***0005737 0006605 0019904 0035308 0042470 0042826 0043085***

# 3 Translation model

Here we use Moses system to build the translation model for DNA sequence to English. We select its phrase model [9]. We use 5% corpus data for testing. The other data are selected for training translation model. We select two groups of corpus. First group only contain the gene data of human. The second group is the mixed data of several model species.

The only input for Moses is parallel corpus. All parameters are set as default values. The language model for target language is set as 2-grams. Then we could get a gene sequence to Gene ontology translation model. The main process to build gene translation model could be describe by Figure.1:

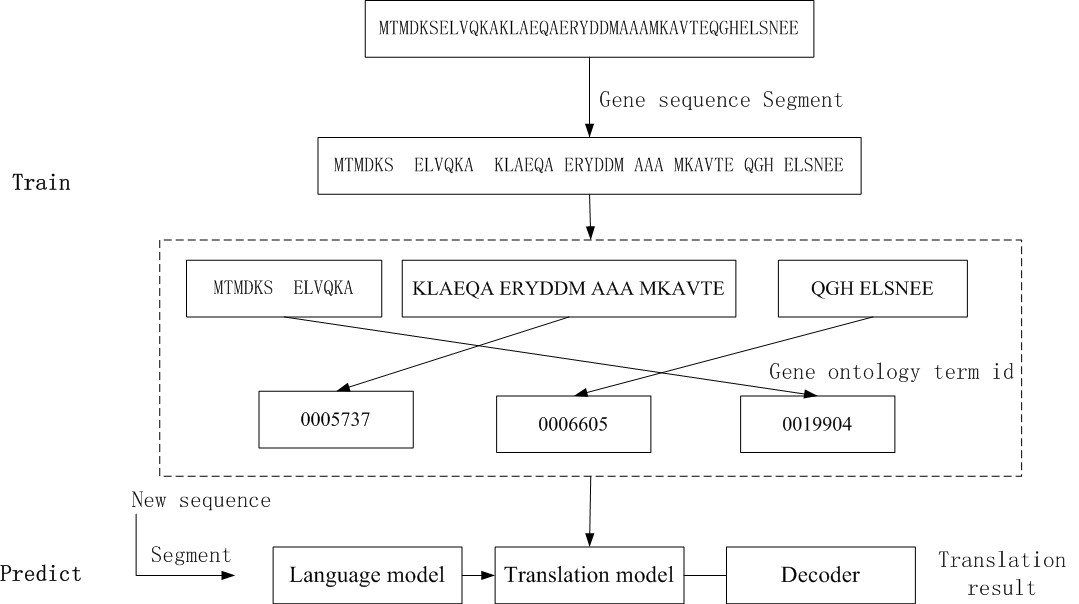


Fig.1 Gene translation system

Normally, we use BLEU score to judge effect of translation system. The main idea of BLEU score could be described by formula.1

 (1)

Where m is number of words from the candidate that are found in the reference, and  is the total number of words in the candidate.

The improved version of BLEU normally considers the n-grams match. Because we don’t care the order of Go ontology words, we only need the unigram word form BLEU.

Because different gene sequence segment method will produce completely different sequences. And the segment methods mainly rely on the selection of maximal word length. So we try the different word lengths. The relation of maximal word length and BLEU score are shown in Fig.2:

fig_2.tif

Fig.2 the relation of maximal word length in segmentation and BLEU score. The red line corresponds to human data set. Blue line is mixed data set and green is big data set

The Fig.3 shows the maximal word length 7 or 8 is the good choices for translation. Its theory explanation is that the 6 length gene ‘words’ has the similar semantic granularity to the terms of Gene ontology. In theory, we need 20^n letters corpus to train an n word length segmenting model. For n=8, we need about 26G protein sequence data. For n=9, we need 512G. There is still no so many annotation data. So a practical word length selection could be 7.

If we don’t segment the gene sequence and treat every protein letter as ‘word’, 20+ protein letters could only express 20+ functions. But from Fig.3, we could still get a translation model if having enough corpuses. It’s mainly because the training system could combine some consecutive letters into ‘phrase’ in alignment process. So such ‘phrase’ could represent more things.

For a usable translation system, its unigram BLEU score should be more than 0.5. The gene translation model has reached this level. So we could compare it with conventional search based method.

# 4 Comparing with search based method

The sections above shows the DNA to function translation is workable. Then we need compare it with current DNA sequence prediction methods.

We design the comparison as follows:

1. Data set: we divide the parallel corpus into 95% training corpus and 5% test corpus.
2. Translation method: we train a gene sequence to Gene ontology translation model based on training corpus and then predict the functions of test gene sequence based on this translation model.
3. Search based method: We index the gene sequence in training corpus by BLAST. To predict the function of test sequences, we search every sequence in BLAST database. The Gene ontology description of best match sequence is regarded as the function of this test sequence.
4. Comparing method. We compare the predicted function of two methods by BLEU score. The results are shown in Table1.

Table.1 BLEU score of search based method and translation method

|  |  |  |  |
| --- | --- | --- | --- |
|  | human data | mixed data | Big data |
| Search method | 0.41 | 0.54 | 0.83 |
| Translation | 0.35 | 0.37 | 0.50 |

From Table.1, we could find the performance of translation model is still not as good as BLAST. There are about 1%-5% sequences whose functions the BLAST method couldn’t predict. But translation methods could predict almost all. Because there are some repeated sequences in corpus, the randomly selected test sequence may exactly match the training sequence. So the better performance of BLAST is in line with our expectation. Here we mainly want to show feasibility of new method. More data sets and translation methods could be tried in the future.

# 5 Summary

Based on statistical machine translation technology, we present a novel method to predict the function of DNA sequence. Although its performance is still not ideal, it shows the application of machine translation in gene function prediction is a workable way. For statistical translation, the more corpuses we have, the better result we could obtain. The Google’s experience has proven that vast amounts of corpus with few rules or even without any rules could produce excellent translation result [10]. Since DNA research also enters the ‘big data’ period, and the gene function prediction problem is so close to ‘language translation’, translation based gene function prediction is likely to be the most potential method.

# References and Note

1. Yuanfang Guan, Chad L Myers, David C Hess, Predicting gene function in a hierarchical context with an ensemble of classifiers. Genome Biology. **9(Suppl 1)**,(2008)
2. Pavlidis P, Gillis J. (2012) Progress and challenges in the computational prediction of gene function using networks [v1; ref status: indexed, http://f1000r.es/SqmJUM] F1000Research 2012, 1:14 (doi: 10.12688/f1000research.1-14.v1)
3. Jonathan A. Eisen1, Phylogenomics: Improving Functional Predictions for Uncharacterized Genes by Evolutionary Analysis. Genome Research. **8**,163-167(1998)
4. Whisstock JC, Lesk AM, Prediction of protein function from protein sequence and structure.Q Rev Biophys. **6(3)**,307-40(2003).
5. Ashburner M, Ball CA, Blake JA, Gene Ontology: tool for the unification of biology, The Gene Ontology Consortium*.* Nature Genet. **25**, 25-29(2000).
6. Peter F Brown, John Cocke, Stephen A, A Statistical Approach to Machine Translation, Computational Linguistics.**16(2)**,79-85(1990).
7. D. Chiang. Hierarchical phrase-based translation. Computational linguistics. **33(2)**, 201-228(2007).
8. Wang Liang, Zhao Kaiyong. Segmenting DNA sequence into `words'. <http://arxiv.org/abs/1202.2518>
9. Philipp Koehn, Hieu Hoang, Alexandra Birch. Moses: Open Source Toolkit for Statistical Machine Translation, *Annual Meeting of the Association for Computational Linguistics (ACL),* *demonstration session, Prague, Czech Republic*. (2007).
10. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In Proceedings of Workshop at ICLR.(2013).

# Data and source code

In attached files:

**1 Directory “/data”: contain the parallel corpus, include:**

human.pr : human gene sequence (protein form) for training, which has been segmented(set maximal word length as 7, same for other gene corpus)

human.go: human gene function description in Gene ontology terms, for training

human.pr.test : human gene sequence (protein form) for test, has been segmented

human.go.test : human gene function description in Gene ontology terms, for test

mixed.pr : mixed gene sequence (protein form) for training

mixed.go: mixed gene function description in Gene ontology terms, for training

mixed.pr.test : mixed gene sequence (protein form) for test, has been segmented

mixed.go.test : mixed gene function description in Gene ontology terms, for test

big.pr : big gene sequence (protein form) for training

big.go: big gene function description in Gene ontology terms, for training

big.pr.test : big gene sequence (protein form) for test, has been segmented

big.go.test : big gene function description in Gene ontology terms, for test

**2 Directory “/code”: contain the source and example to train the gene translation model, we use mix data set to train the translation model, include:**

readme.txt: direction about how to train the model

Content in readme:

**1 System requirement**

(1) You need install moses system. http://www.statmt.org/moses/?n=Development.GetStarted

(2) You'd better run the baseline of Moses. <http://www.statmt.org/moses/?n=Moses.Baseline>

(3) Python 2.7.

Here we use irstlm to train language model, as this baseline suggest.

**2 parallel corpus**

for train:

gene.6.pr : gene sequence, have been segmented, maximal word lenght 6

gene.6.go:  gene ontology sequence

for test:

gene.6.pr.test:  gene sequence

gene.6.go.test: gene ontology sequence

**3 train language model of target sequence, here we train gene ontology model**

file: train\_lm.sh

(attention: please revise the moses\_path and irstml\_path to your own path!)

run:

./train\_lm.sh gene.6.go

You will get gene.6.go.blm.

**4 train translation model**

File: train\_ts.sh

(attention: please revise the moses\_path and  giza\_path to your path!)

run:

./train\_ts.sh gene.6 gene.6.go.blm

Wait about 4 hours (1 CPU. 10+G memory requirement)

you will get a "train" directory.

the train/model/moses.ini is just your translation model file

**5 test**

File: get\_bleu.sh

(please revise the moses\_path)

run:

./get\_bleu.sh gene.6.go.test gene.6.pr.test

it will translate gene.6.pr.test and then compare it with gene.6.go.test. If there is no error, you will get the result:

*BLEU = 26.38,* ***47.4****/30.0/21.4/15.9 (BP=1.000, ratio=1.571, hyp\_len=48369, ref\_len=30786)*

Because we don't consider the order of go terms, 47.4 is just its BLEU score (unigram). You could still divide the test corpuses into test part and tuning part, and then run the tuning process. It could bring some improvement for bleu score, but it may take several days.

Moreover, if you want to test segment methods or build your own corpus from original database source, please refer to our open source project

https://code.google.com/p/dnasearchengine/downloads/list