A Grammatical Inference approach to Transmembrane domain prediction.

Piedachu Peris, Damián López and Marcelino Campos

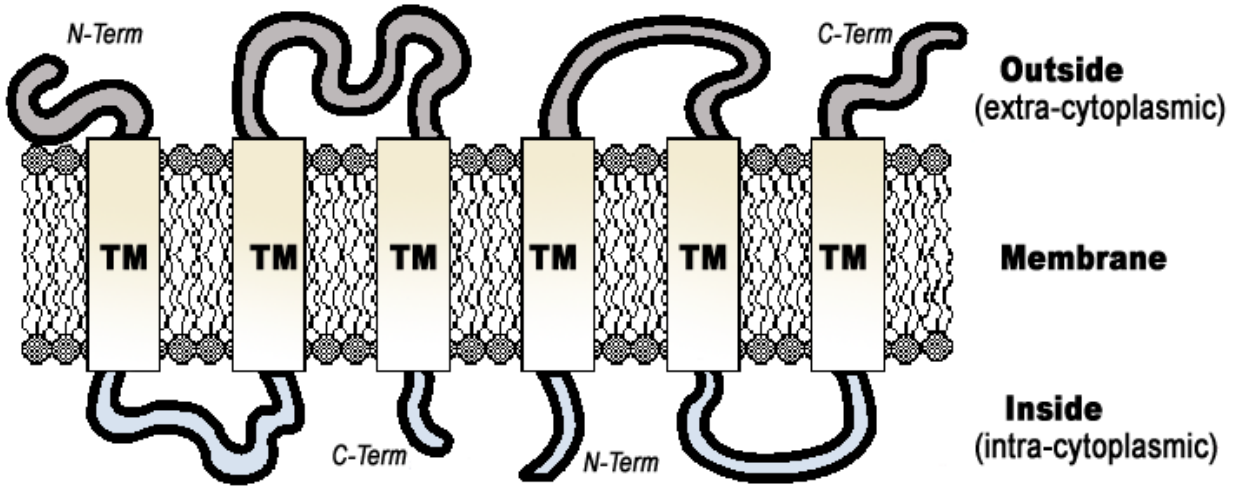
Departamento de Sistemas Informáticos y Computación.

Universidad Politécnica de Valencia.

pperis@dsic.upv.es dlopez@dsic.upv.es mcampos@dsic.upv.es

Introduction

- Transmembrane proteins:



- Involved in:
  - Communication between cells
  - Transport of ions and nutrients
  - Reception of viruses
  - Diabetes, hypertension, depression, arthritis, cancer...

Introduction

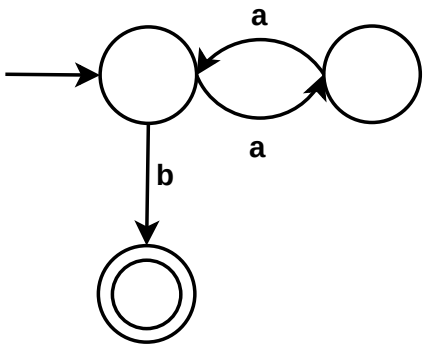
- Prediction of transmembrane regions in proteins.
- Different approaches:
  - Hidden Markov Models:
    - Sonnhammer E. *et al.: TMHMM*
  - Neural Networks:
    - Fariselli P. *et al.: HTP*
  - Statistical analysis:
    - Pasquier C. *et al.: PRED-TMR*
- Our approach (igTM): Based on Grammatical Inference.

Preliminary concepts (I)

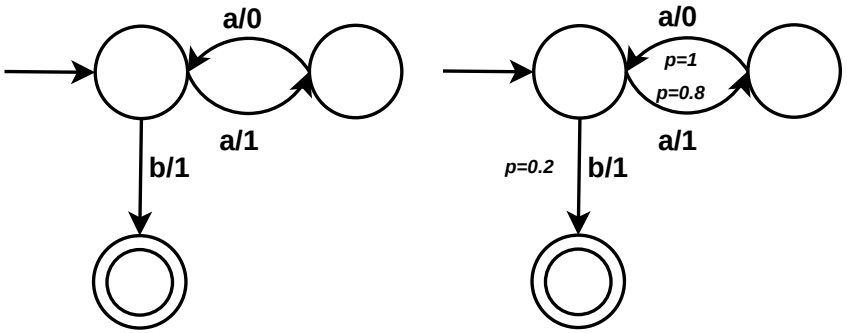
- Alphabet:
  $\Sigma = \{a, b, c, d, e, f, g\}$
  $\Delta = \{A, B, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y, Z\}$
- Word:
  $u = abababab$
  $w = abcddabfgedfc$
  $v = MNYIFDLSILLVVA$
- Language:
  $L_1 = \{a^n b^n : n \geq 1\}$
  $L_2 = \{transmembrane\ proteins\ sequences\}$
  $L_3 = \{df^m a^n : m \geq 1 n \geq 0\}$

Preliminary concepts (II)

- Finite automaton:



- Transducer:



Grammatical Inference (GI)

- Goal: Learn a language from a sample of words.

$$S = \{aab, aaaab, aaaaaab\}$$

- Different GI algorithm $\rightarrow$ different language:

$$L_a = \{a^n b : n \geq 1\}$$

$$L_b = \{a^n b : n \geq 2\}$$

$$L_c = \{(aa)^n b : n \geq 1\}$$

- Greater alphabet $\rightarrow$ more difficult to learn a language.

Method

1. Words: Set of proteins (sequences of amino acids)
   $W = \{MDAIKKM, GDAVKK, MDAAIKKM\}$

2. Alphabet reduction: Dayhoff

   `MDAIKKM` `GDAVKK` `MDAAIKKM`

   `ecbedde` `bcbedd` `ecbbedde`

3. Domain and topology annotation:

   `ecbedde` `bcbedd` `ecbbedde`

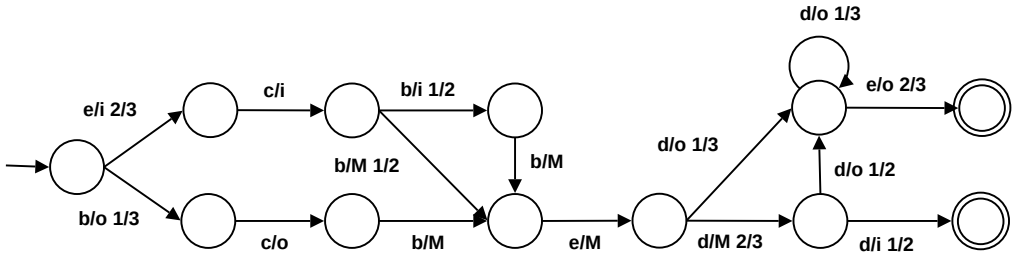
   `iiMMMoo` `ooMMMi` `iiiMMooo`

Amino acid	Dayhoff
C	a
G, S, T, A, P	b
D, E, N, q	c
R, H, K,	d
L, V, M, I	e
Y, F, W	f
B, Z	g

Method (II)

4. GI process: Inference of a probabilistic transducer:
   input: protein + annotation (each symbol related to its label):

   `[ei] [ci] [bM] [eM] [dM] [do] [eo]`
   `[bo] [co] [bM] [eM] [dM] [di]`
   `[ei] [ci] [bi] [bM] [eM] [do] [do] [eo]`



   output: annotation of words (proteins): iiMMMoo ooMMMi iiiMMooo

5. Test phase: returns the transduction that is most likely produced by the input string.
   input: MDAIKKKHL → ecbedddde
   output: iiiMMoooo

Databases

We used three datasets to train and test our method:

- TMHMM database: set of 160 transmembrane proteins, available at:
  `http://www.cbs.dtu.dk/~krogh/TMHMM`.
- TMPDB: set of 302 transmembrane proteins, available at:
  `http://www.genome.jp/SIT/tsegdir/whatis_tmpdb.html`.
- 101-pred-TMR db: Set of 101 transmembrane proteins, used to elaborate the pred-TMR prediction method. We downloaded each of the proteins from Uniprot web page.

Performance measures

- Sensitivity (Sn) $S_n = \frac{TP}{TP+FN}$
- Specificity (Sp) $S_p = \frac{TP}{TP+FP}$
- Correlation coefficient (CC)
  $CC = \frac{(TP \times TN) - (FN \times FP)}{\sqrt{(TP+FN) \times (TN+FP) \times (TP+FP) \times (TN+FN)}}$
- Average conditional probability (ACP)
  $ACP = \frac{1}{4}\left[\frac{TP}{TP+FN} + \frac{TP}{TP+FP} + \frac{TN}{TN+FP} + \frac{TN}{TN+FN}\right]$
- Approximated correlation (AC) $AC = (ACP - 0{,}5) \times 2$

Experimentation

- Encoding and annotation of an example sequence for each different experimental configuration:

```
Sequence:    MRVTAPRTLLLLLWGAVALTETWAGSHSMR
Dayhoff:     edebbbdbeeeeefbbebebcbfbbbdbed
TM domains: 4-10, 20-25
exp1:        edebbbdbeeeeefbbebebcbfbbbdbed...MMMMMMM.........MMMMMM.....
exp2:        edebbbdbeeeeefbbebebcbfbbbdbedoooMMMMMMMiiiiiiiiiMMMMMMooooo
exp3:        edebbbdbeeeeefbbebebcbfbbbdbedoooNNNNNNNiiiiiiiiiPPPPPPooooo
exp4:        edebbbdbeeeeefbbebebcbfbbbdbedOOONNNNNNNiiiiIIIIIPPPPPPooooo
exp5:        edebbbdbeeeeefbbebebcbfbbbdbedooCMMMMMMMDiiiiiiiAMMMMMMBoooo
exp6:        MRVTAPRTLLLLLWGAVALTETWAGSHSMRoooMMMMMMMiiiiiiiiiMMMMMMooooo
```

Results - TMHMM database

		TMHMM database		
		Sn	**Sp**	**AC**
igTM	**exp2**	0.795	0.808	0.692
	exp3	0.820	0.794	0.703
	exp4	0.748	0.801	0.656
	exp5	0.808	**0.810**	0.702
	exp6	0.819	0.796	**0.707**
TMHMM		0.900	0.879	<u>0.827</u>
Pred-TMR		0.786	<u>0.898</u>	0.767
S-TMHMM		0.832	0.854	0.768

Results - TMPDB

		TMPDB		
		Sn	**Sp**	**AC**
igTM	**exp1**	0.675	0.757	0.538
	exp2	0.690	0.751	0.542
	exp3	0.670	0.741	0.530
	exp4	0.601	0.735	0.476
	exp5	0.683	0.750	0.539
	exp6	0.710	**0.759**	**0.557**
TMHMM		0.739	0.831	0.659
Pred-TMR		0.777	<u>0.899</u>	<u>0.756</u>
S-TMHMM		0.737	0.829	0.659

Results - 101-PRED-TMR-DB

		101-PRED-TMR-DB			
		Sn	**Sp**	**CC**	**AC**
igTM	**exp2**	0.810	0.811	0.702	0.702
	exp3	0.758	0.781	0.667	0.652
	exp4	0.693	0.795	0.640	0.618
	exp5	0.793	**0.821**	0.697	0.692
	exp6	0.801	0.820	**0.855**	**0.709**
TMHMM		0.899	0.871	0.822	0.817
Pred-TMR		0.814	0.909	0.792	0.795
WaveTM		-	-	0.77	-
HMMTOP		-	-	0.82	-
S-TMHMM		0.831	0.840	0.772	0.760

Conclusions and future work

- Results in line with those existing in literature
- This system does not need any biological knowledge.
- Method can be tested online at:
  http://esparta.dsic.upv.es:8080/code/igtm.php
- Future work:
  - use this method together with another one, based on HMM, to perform better.
  - train this method with another (larger if possible) databases (e.g.: http://opm.phar.umich.edu/)
  - new inference algorithms

Thank you!

Any question?