Genome 540 discussion

January 14th, 2025
Joe Min

Genome Sciences
UNIVERSITY OF WASHINGTON

Agenda

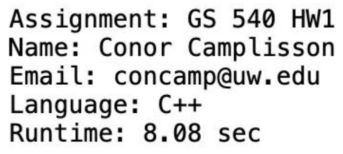
Homework 1 tips and pitfalls

Modeling protein sequence spaces with language models

Homework 1 tips and pitfalls

Anatomy of the output file

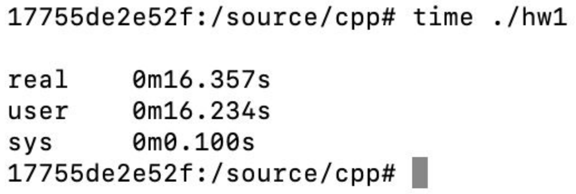
It’s probably easiest to have your program output to a file directly



```
Assignment: GS 540 HW1
Name: Conor Camplisson
Email: concamp@uw.edu
Language: C++
Runtime: 8.08 sec
```

To figure out the runtime, use the `time` command

- Report the “real” runtime

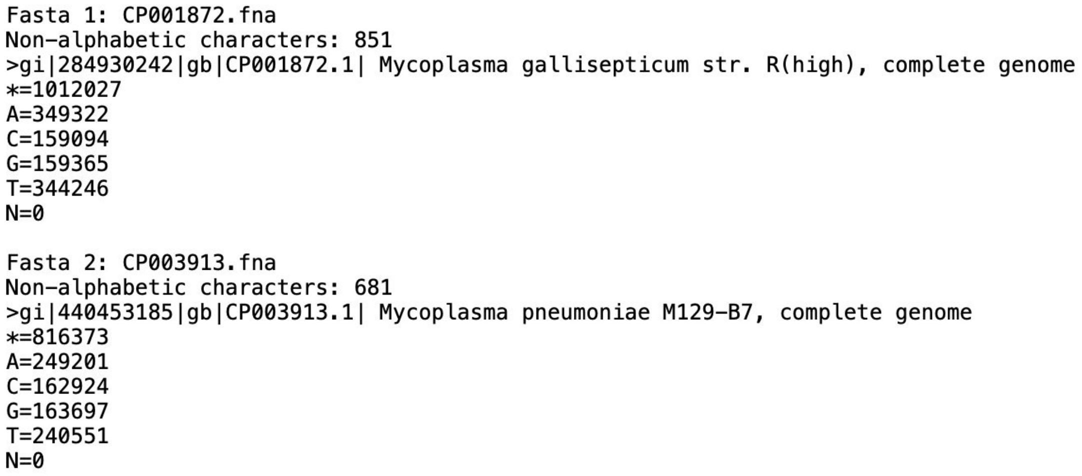


```
17755de2e52f:/source/cpp# time ./hw1

real    0m16.357s
user    0m16.234s
sys     0m0.100s
17755de2e52f:/source/cpp# 
```

Anatomy of the output file

“Non-alphabetic” does not include spaces

“*” means A, C, G, T, or N

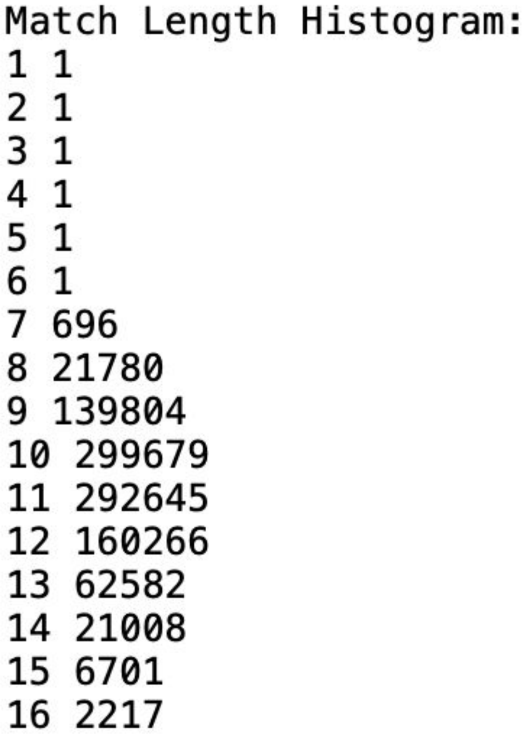


```
Fasta 1: CP001872.fna
Non-alphabetic characters: 851
>gi|284930242|gb|CP001872.1| Mycoplasma gallisepticum str. R(high), complete genome
*=1012027
A=349322
C=159094
G=159365
T=344246
N=0

Fasta 2: CP003913.fna
Non-alphabetic characters: 681
>gi|440453185|gb|CP003913.1| Mycoplasma pneumoniae M129-B7, complete genome
*=816373
A=249201
C=162924
G=163697
T=240551
N=0
```

Anatomy of the output file

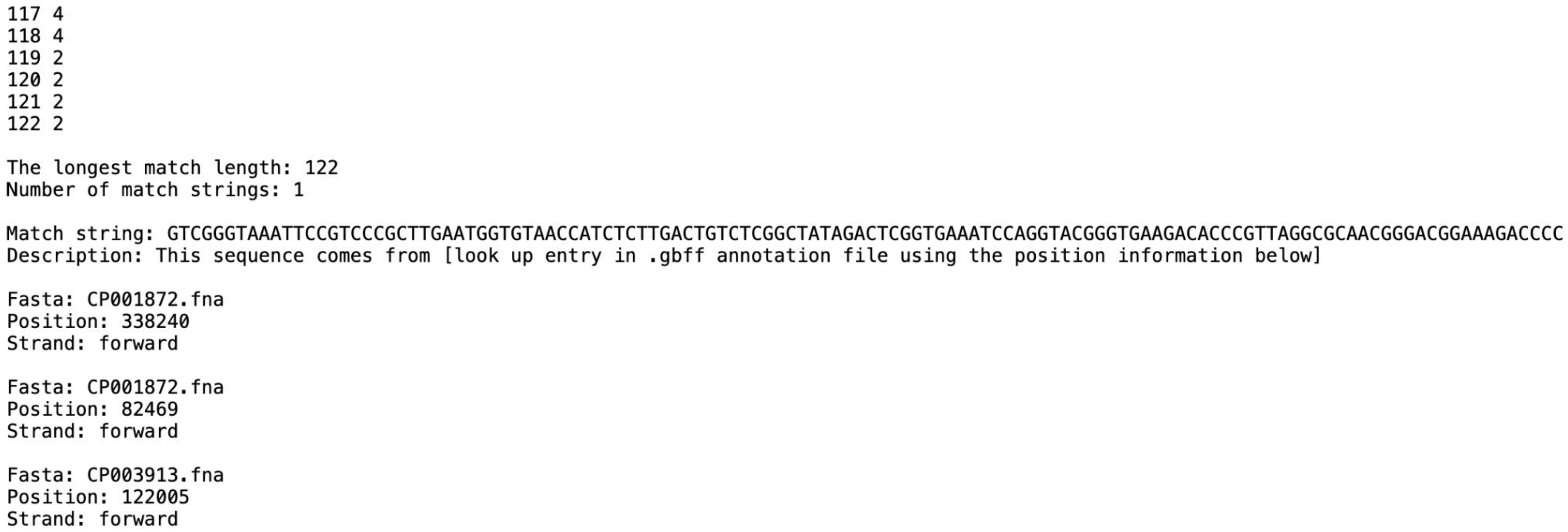
For each suffix in sequence 1:

- Compute its longest match length to any suffix from sequence 2 or its reverse complement
- Add 1 to the bucket corresponding to that match length



```
Match Length Histogram:
1 1
2 1
3 1
4 1
5 1
6 1
7 696
8 21780
9 139804
10 299679
11 292645
12 160266
13 62582
14 21008
15 6701
16 2217
```

Anatomy of the output file

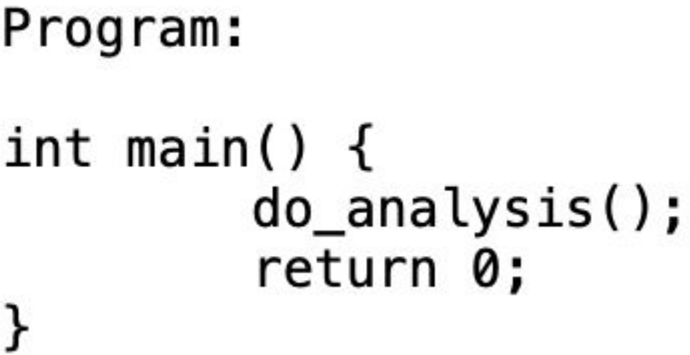


```
117 4
118 4
119 2
120 2
121 2
122 2

The longest match length: 122
Number of match strings: 1

Match string: GTCGGGTAAATTCCGTCCCGCTTGAATGGTGTAACCATCTCTTGACTGTCTCGGCTATAGACTCGGTGAAATCCAGGTACGGGTGAAGACACCCGTTAGGCGCAACGGGACGGAAAGACCCC
Description: This sequence comes from [look up entry in .gbff annotation file using the position information below]

Fasta: CP001872.fna
Position: 338240
Strand: forward

Fasta: CP001872.fna
Position: 82469
Strand: forward

Fasta: CP003913.fna
Position: 122005
Strand: forward
```

“Number of match strings” only counts unique strings

The same match string may occur in multiple places

Anatomy of the output file



```
Program:

int main() {
        do_analysis();
        return 0;
}
```

No need to include build instructions

Can just paste the text from your program file(s)

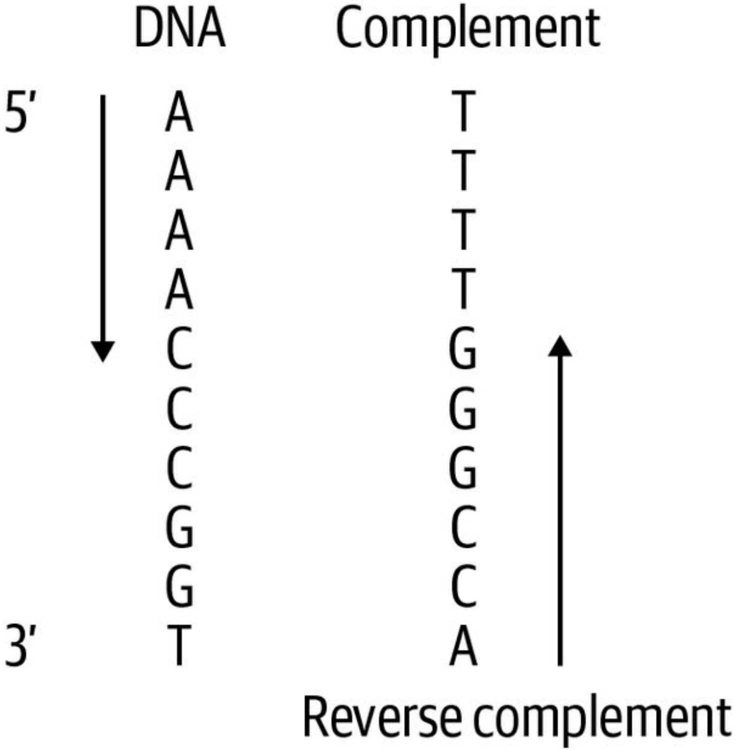
Reverse complement quick refresher

AAAACCCGGT

turns into

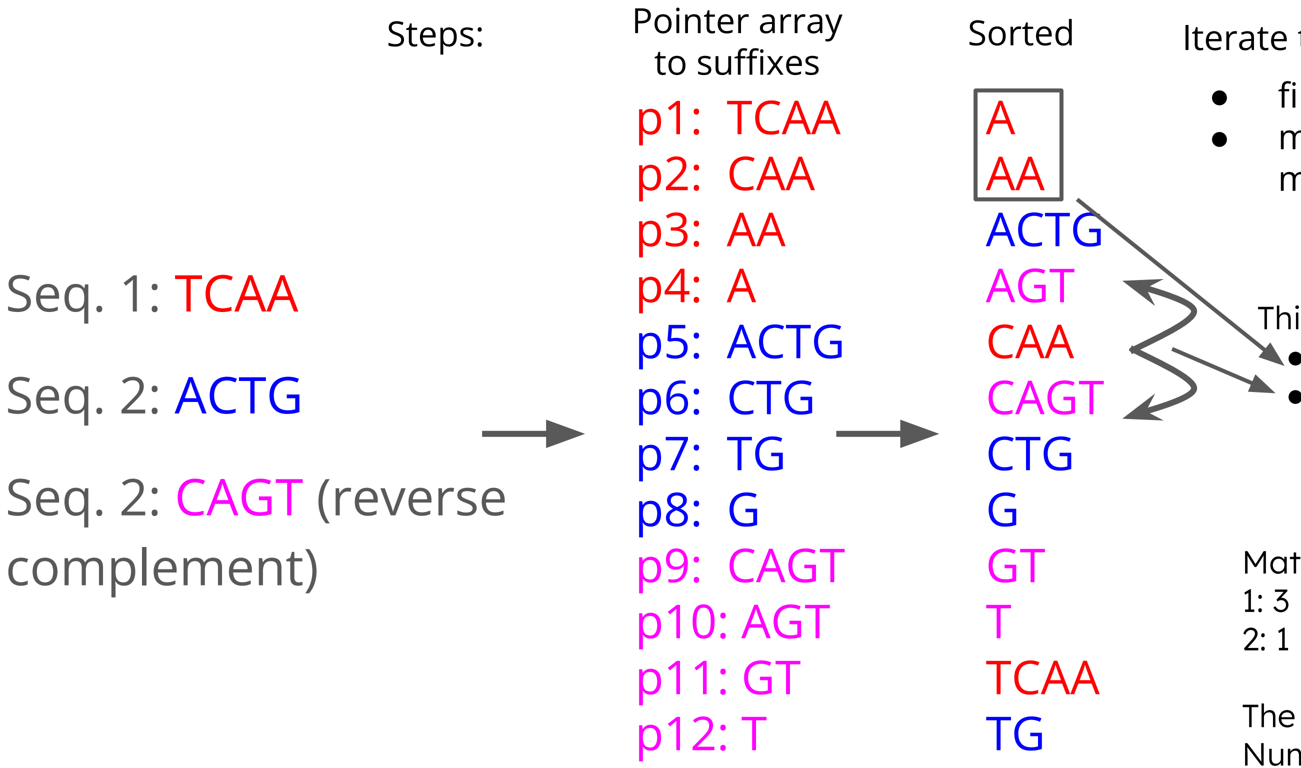
ACCGGGTTTT

- Not TTTTGGGCCA



	DNA	Complement
5′	A	T
	A	T
	A	T
	A	T
	C	G
	C	G
	C	G
	G	C
	G	C
3′	T	A

Small Example



Steps:

Seq. 1: TCAA

Seq. 2: ACTG

Seq. 2: CAGT (reverse complement)

Pointer array to suffixes

p1: TCAA
p2: CAA
p3: AA
p4: A
p5: ACTG
p6: CTG
p7: TG
p8: G
p9: CAGT
p10: AGT
p11: GT
p12: T

Sorted

A
AA
ACTG
AGT
CAA
CAGT
CTG
G
GT
T
TCAA
TG

Iterate through sorted list

- find the longest match to seq1
- make a histogram of seq1 longest matches

Things to consider:

- Multiple seq1 suffixes in a row
- Need to look above and below to find longest match

Match Lengths:
1: 3
2: 1

The longest match length: 2
Number of match strings: 1

Protein sequence language models

What is a language model?

A language model learns some **meaningful embedding space** that is well-behaved under vector arithmetic

- E.g., the vector for “woman” + the vector for “monarch” may result in the same vector as “queen”
- This allows us to perform math on words and understand relationships between them

Uh… relevance??

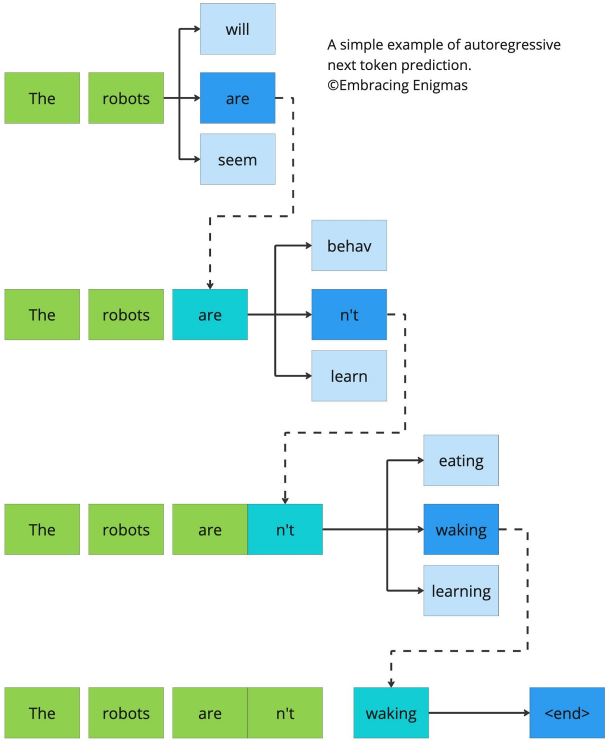
DNA or protein sequence models are extensions of the idea of site/background models

- “site” might be an actual DNA site, or a protein sequence of interest
- “background” might be non-functional protein sequences

How are language models trained?

Next token prediction

- “The cat sat on the … [???]”
- Assign probabilities to possible next tokens given previous token window
- ChatGPT basically uses this approach



A simple example of autoregressive next token prediction.
©Embracing Enigmas

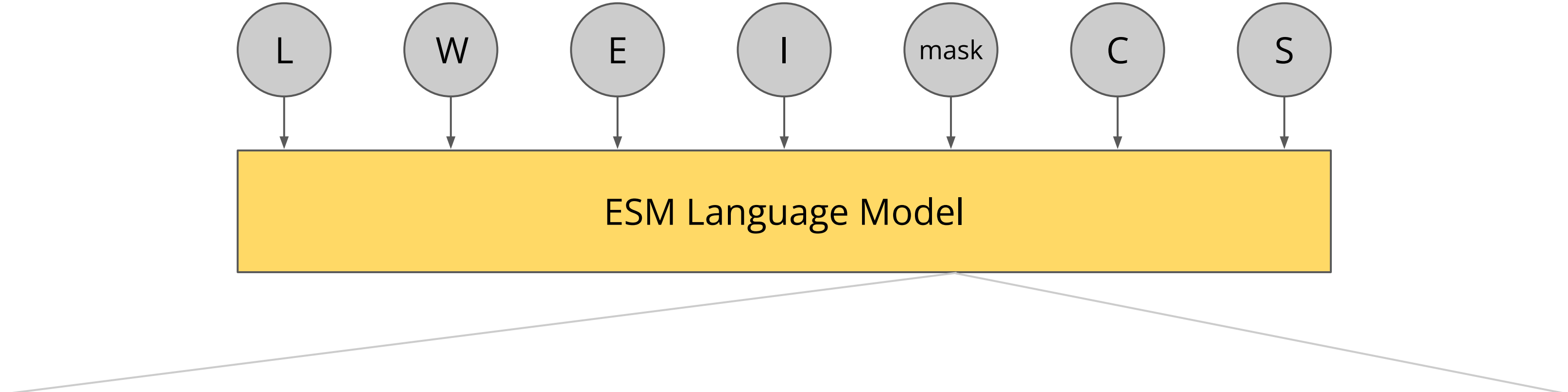
How are language models trained?

Masked token prediction

- “The cat [???] on the mat”
- Assign probabilities to some set of tokens given the surrounding context window
- ESM, a protein language model, was trained this way

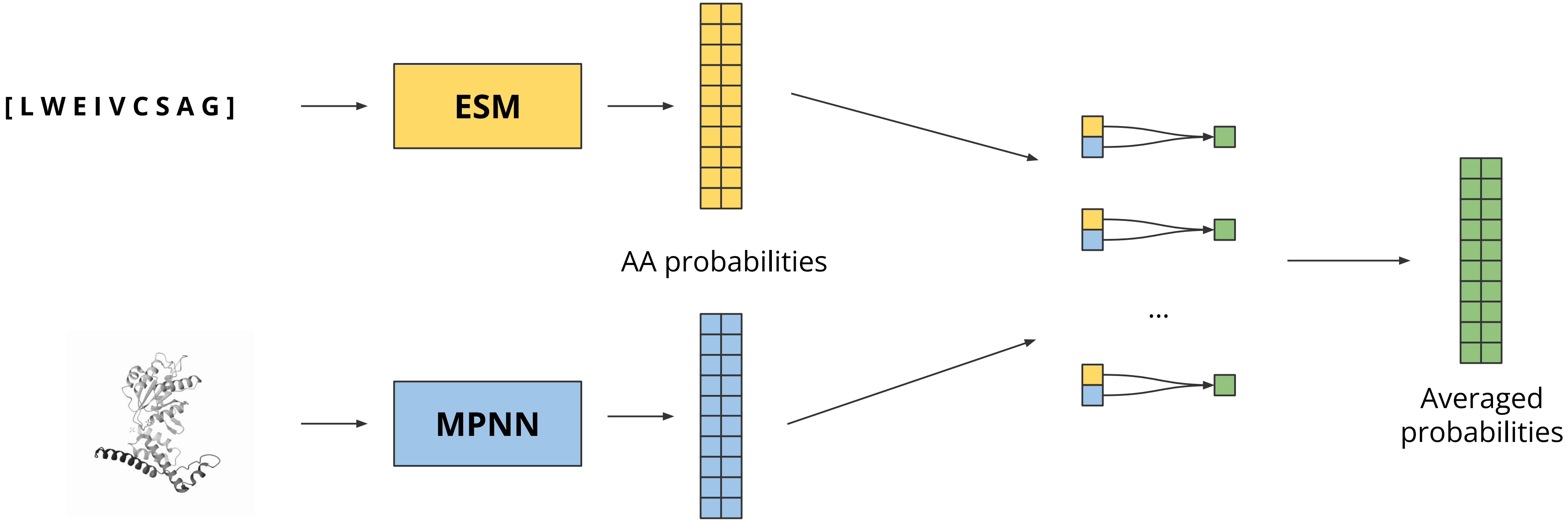
ESM has learned the natural protein space

Evolutionary Scale Model (ESM) is trained on UniRef using masked token prediction



A	C	D	E	F	G	H	I	K	L	M	N	P	Q	R	S	T	**V**	W	Y
.14	.01	.06	.02	.03	.02	.01	.02	.04	.04	.02	.03	.01	.01	.05	.01	.02	.42	.03	.01

How I use ESM for protein design



Office hours (30m)

Feel free to hang out and work on the homework, ask questions, or leave!