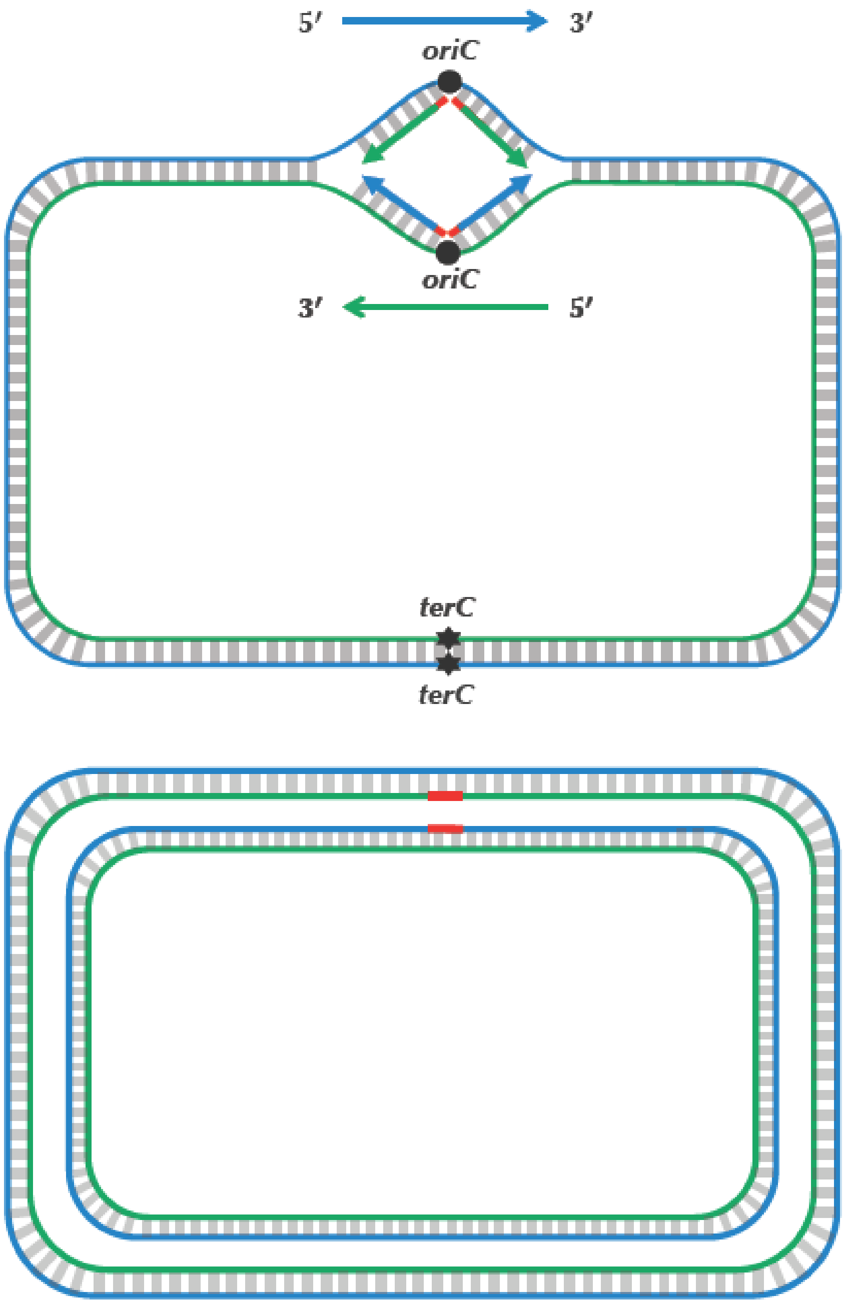
CMSC423

Chapter 1 – DNA replication

Bacterial replication



Outline

- Vague question: find OriC

- Two paradigms
  - look for surprising events
  - leverage biological knowledge

- Computationally
  - Counting letters and words (the kindergarten of CS)

https://en.wikipedia.org/wiki/Count_von_Count

How do we find hidden messages?

- Look for deviations from what we expect
- Random DNA strings do not have long "parts" that repeat nearby each other
- Key idea: find k-mers that are more frequent than expected
  - globally
  - nearby each other (in clumps)
  - allowing for some errors

CS break

- Write pseudo-code that finds number of occurrences of a **given** k-mer.

- Socrative.com (room 187417)

CS break

- Write pseudo-code that finds number of occurrences of all k-mers in a string.

- Socrative.com (room 187417)

Encodings

- String 2 number
  A – 00, C – 01, G – 10, T – 11

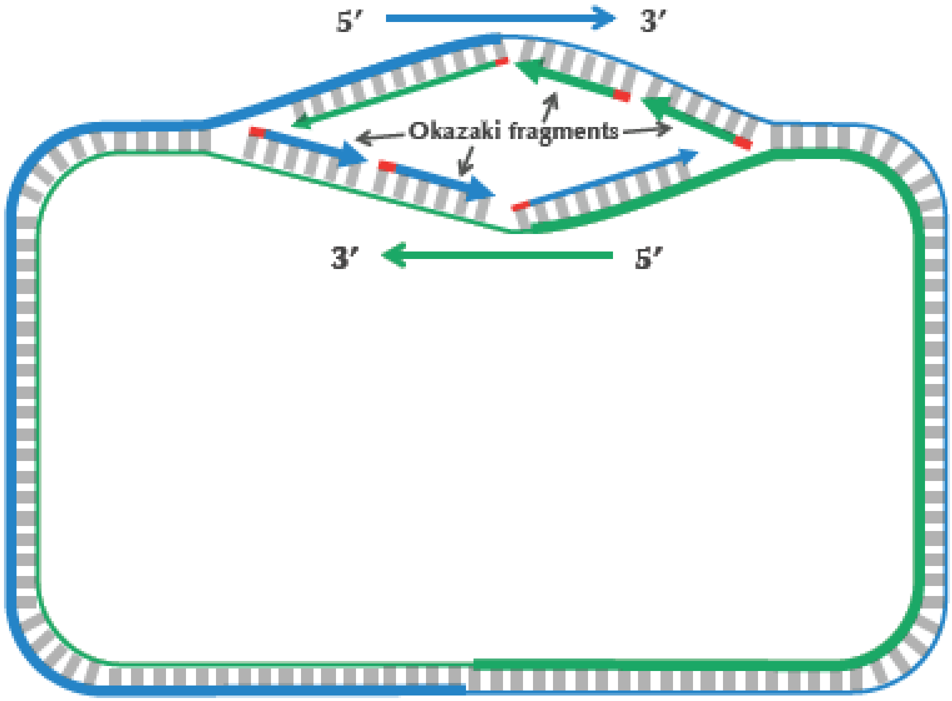
  A C C A
  00010100 =
- Number 2 string
  - simply reverse the process... simple?

- word2vec – a different type of encoding

Some knowledge of biology is helpful

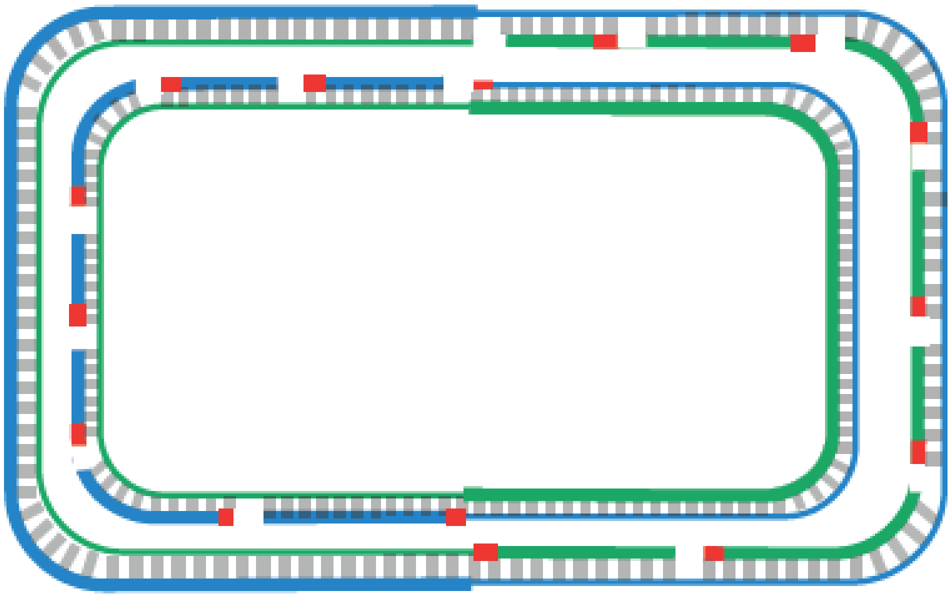
- DNA is double-stranded
- k-mer can occur in either strand
- Algorithms stay the same but need to run twice
- Need to know how to reverse complement

A lot more biology



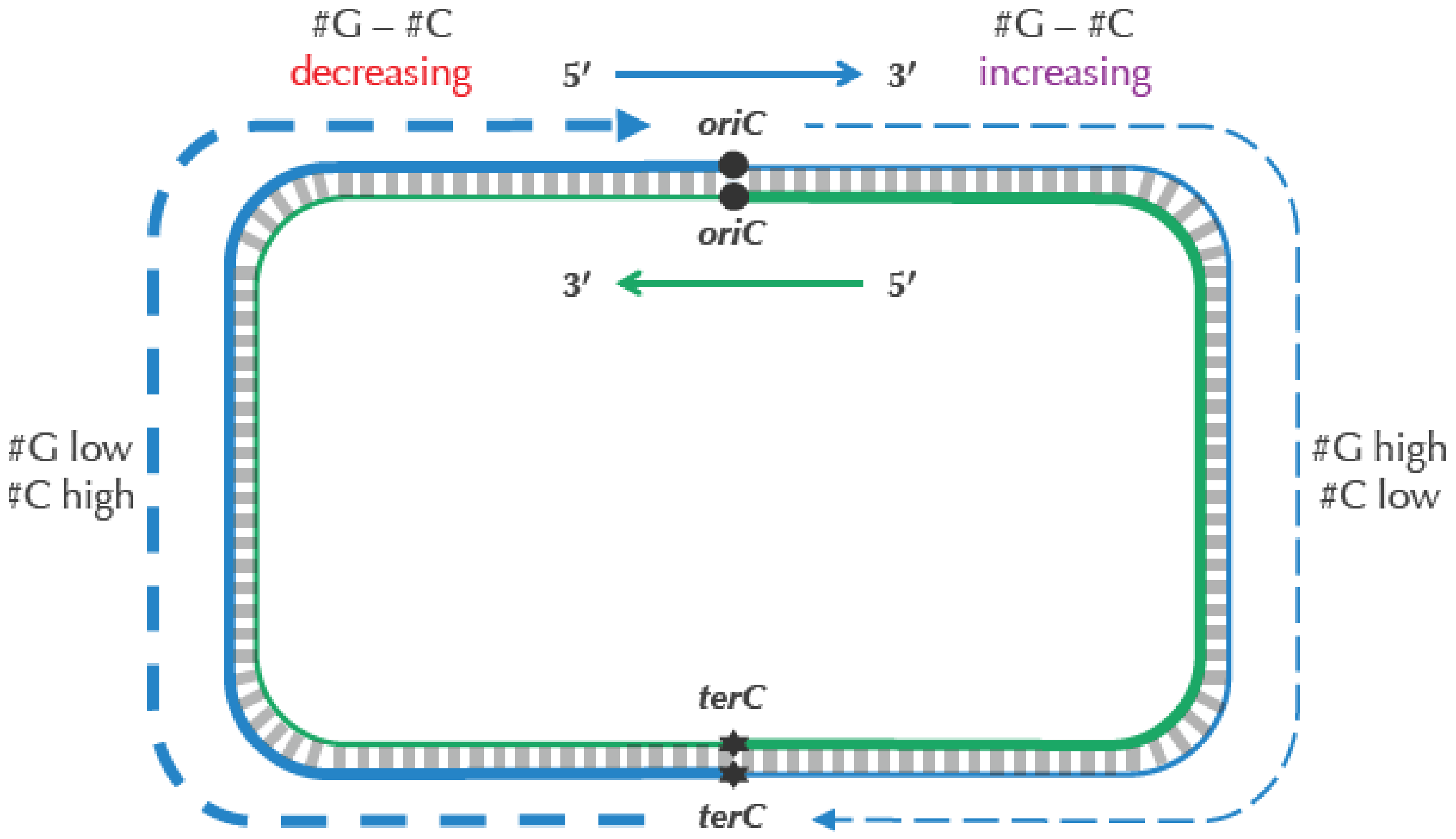
Deamination: C -> T mutation
more frequent in single stranded DNA

Also occurs with time (ancient DNA)



Interesting patterns...

- Simply count "skew" between G and C.



Later in the class

- Finding a pattern efficiently
- Finding patterns with mismatches/errors

- How fast?