Technische Universität Braunschweig

Hannover Medical School

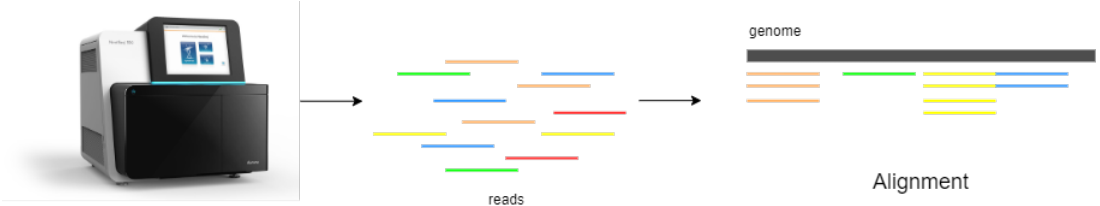
PETER L. REICHERTZ INSTITUTE FOR MEDICAL INFORMATICS

Final Project 2
Burrows-Wheeler Algorithm

Lisa-Marie Bente
Division Data Science in Biomedicine
Peter L. Reichertz Institute for Medical Informatics
of TU Braunschweig and Hannover Medical School
lisa-marie.bente@plri.de, www.plri.de

Usage of the Burrows-Wheeler Transform

- Originally: data compression
- Search of substrings in transformed string
- Bioinformatics: map reads to genome
- Decompression without data loss possible



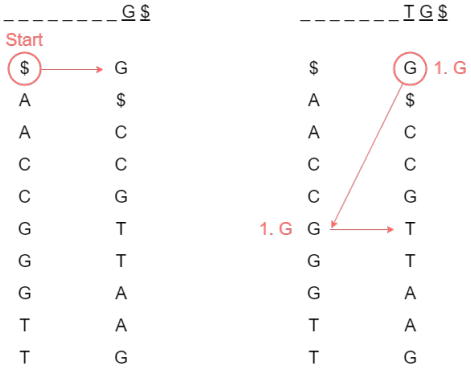
The Algorithm: Transformation

1. Append $ to template T
2. Generate all cyclic rotations of T and enumerate them
3. Sort rotations lexicographically
4. BWT(T) is the last column of the sorted matrix

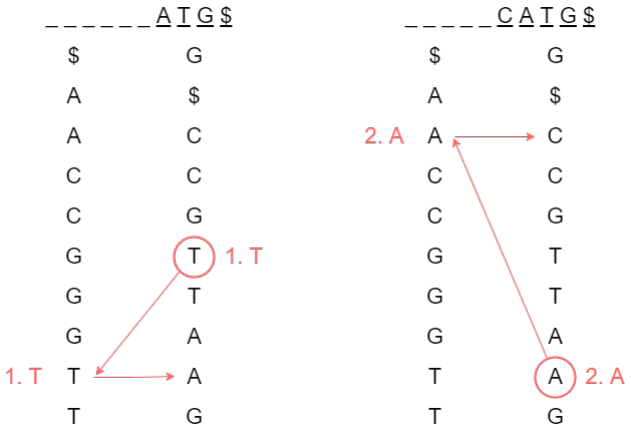
```
T = AGTGCCATG$

$AGTGCCATG 9            $AGTGCCATG 9
G$AGTGCCAT 8            AGTGCCATG$ 0
TG$AGTGCCA 7            ATG$AGTGCC 6
ATG$AGTGCC 6            CATG$AGTGC 5
CATG$AGTGC 5    -->     CCATG$AGTG 4
CCATG$AGTG 4            G$AGTGCCAT 8
GCCATG$AGT 3            GCCATG$AGT 3
TGCCATG$AG 2            GTGCCATG$A 1
GTGCCATG$A 1            TG$AGTGCCA 7
AGTGCCATG$ 0            TGCCATG$AG 2

BWT(T)= G$CCGTTAAG
```

The Algorithm: Inverse BWT

1. Sort BWT lexicographically to get first column of matrix
2. Use 'last first'-assignment to reconstruct original template



The Algorithm: Inverse BWT



The Algorithm: Inverse BWT

	_ _ _ _ C C A T G $	
	$	G
	A	$
	A	C
1. C	C	C
	C	G
	G	T
	G	T
	G	A
	T	A
	T	G

1. C

	_ _ _ G C C A T G $	
	$	G
	A	$
	A	C
	C	C
2. C	C	G
	G	T
	G	T
	G	A
	T	A
	T	G

2. C

The Algorithm: Inverse BWT

	_ _ T G C C A T G $	
	$	G
	A	$
	A	C
	C	C
	C	G
	G	T
2. G	G	T
	G	A
	T	A
	T	G

	_ G T G C C A T G $	
	$	G
	A	$
	A	C
	C	C
	C	G
	G	T
	G	T
	G	A
	T	A
2. T	T	G

The Algorithm: Inverse BWT

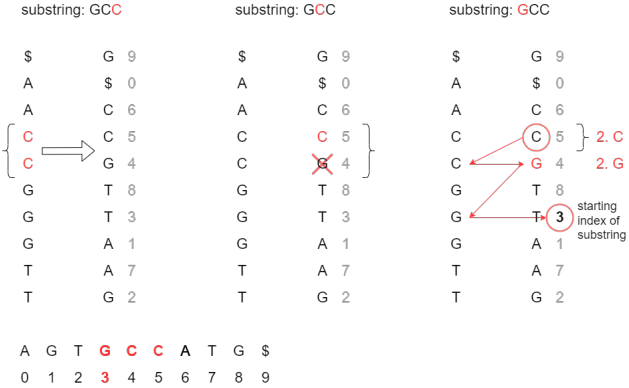
	A G T G C C A T G $	
	$	G
	A	$
	A	C
	C	C
	C	G
	G	T
	G	T
3. G	G	A
	T	A
	T	G 3. G

	A G T G C C A T G $	
	$	G
1. A	A	$ Stop
	A	C
	C	C
	C	G
	G	T
	G	T
	G	A 1. A
	T	A
	T	G

PLRI PETER L. REICHERTZ INSTITUTE FOR MEDICAL INFORMATICS

The Algorithm: Substring Search

1. Determine interval in first column with last letter of substring
2. Look for second to last letter of substring in this interval
3. Repeat until you reach the first letter of the substring



Can A Neural Network Learn BWT?

1. Implement the BWT, inverse BWT, and substring search in Python
2. Implement a neural network and train it for BWT with substring search using data generated with your implementation
3. Compare the two methods
4. Develop an autoencoder that achieves a better compression than Burrows-Wheeler
5. Write a short report (3 - 5 pages) & prepare a presentation on your project