

# Ultrafast and memory-efficient alignment of short DNA sequences to the human genome

*Ben Langmead, Cole Trapnell,  
Mihai Pop and Steven L Salzberg*

Akshay Sanjeev

May 5, 2025

We are in 2009

# Outline of Bowtie

- ▶ Burrows-Wheeler Transform(Indexing)
- ▶ Exact and inexact alignment
- ▶ Excessive backtracking

# Burrows-Wheeler Transform: Forward Transform

Let  $T = \text{BANANA}$ .  $\text{BWT}(T)$  will be:

\$	B	A	N	A	N	A
A	\$	B	A	N	A	N
N	A	\$	B	A	N	A
A	N	A	\$	B	A	N
N	A	N	A	\$	B	A
A	N	A	N	A	\$	B
B	A	N	A	N	A	\$

# Burrows-Wheeler Transform: Forward Transform

Let  $T = \text{BANANA}$ .  $\text{BWT}(T)$  will be:

$$\begin{bmatrix} \$ & B & A & N & A & N & A \\ A & \$ & B & A & N & A & N \\ N & A & \$ & B & A & N & A \\ A & N & A & \$ & B & A & N \\ N & A & N & A & \$ & B & A \\ A & N & A & N & A & \$ & B \\ B & A & N & A & N & A & \$ \end{bmatrix} \rightarrow \begin{bmatrix} \$ & B & A & N & A & N & A \\ A & \$ & B & A & N & A & N \\ A & N & A & \$ & B & A & N \\ A & N & A & N & A & \$ & B \\ B & A & N & A & N & A & \$ \\ N & A & \$ & B & A & N & A \\ N & A & N & A & \$ & B & A \end{bmatrix}$$

$$\text{BWT}(T) \rightarrow \text{ANNB\$AA}$$

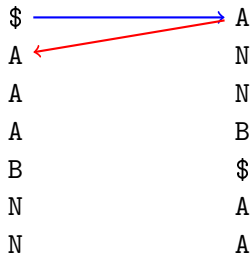
# BWT: Reverse Transform(UNPERMUTE)

\$	A
A	N
A	N
A	B
B	\$
N	A
N	A

# BWT: Reverse Transform(UNPERMUTE)

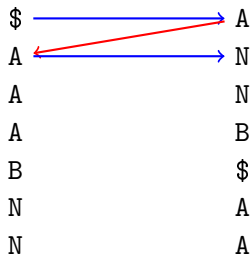
\$	→	A
A		N
A		N
A		B
B		\$
N		A
N		A

## BWT: Reverse Transform(UNPERMUTE)

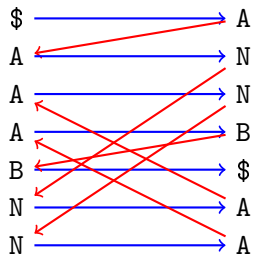




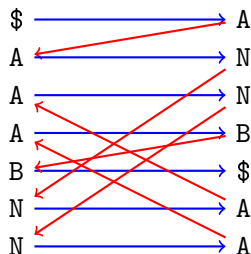
# BWT: Reverse Transform(UNPERMUTE)



# BWT: Reverse Transform(UNPERMUTE)



# BWT: Reverse Transform(UNPERMUTE)



follow the tip of blue arrows to get back the initial message:  
BANANA\$

# EXACTMATCH algorithm

$$\Sigma = B_1A_1N_1A_2N_2A_3.$$

Pattern to find,  $P = \text{NAN}$

\$	$A_1$
$A_1$	$N_1$
$A_2$	$N_2$
$A_3$	$B_1$
$B_1$	\$
$N_1$	$A_2$
$N_2$	$A_3$

# EXACTMATCH algorithm

$$\Sigma = B_1A_1N_1A_2N_2A_3.$$

Pattern to find,  $P = \text{NAN}$

0	\$	$A_1$	1
1	$A_1$	$N_1$	5
2	$A_2$	$N_2$	6
3	$A_3$	$B_1$	4
4	$B_1$	\$	0
5	$N_1$	$A_2$	2
6	$N_2$	$A_3$	3

# EXACTMATCH algorithm

$$\Sigma = B_1A_1N_1A_2N_2A_3.$$

Pattern to find,  $P = \text{NAN}$

top→	0	\$	$A_1$	1
	1	$A_1$	$N_1$	5
	2	$A_2$	$N_2$	6
	3	$A_3$	$B_1$	4
	4	$B_1$	\$	0
	5	$N_1$	$A_2$	2
bot→	6	$N_2$	$A_3$	3

# EXACTMATCH algorithm

$$\Sigma = B_1 A_1 N_1 A_2 N_2 A_3.$$

Pattern to find,  $P = \text{NAN}$

	0	\$	$A_1$	1
	1	$A_1$	$N_1$	5
	2	$A_2$	$N_2$	6
	3	$A_3$	$B_1$	4
	4	$B_1$	\$	0
$N_1 \rightarrow$	5	$N_1$	$A_2$	2
$N_2 \rightarrow$	6	$N_2$	$A_3$	3

# EXACTMATCH algorithm

$$\Sigma = B_1A_1N_1A_2N_2A_3.$$

Pattern to find,  $P = \text{NAN}$

	0	\$	$A_1$	1
	1	$A_1$	$N_1$	5
$A_2N_1 \rightarrow$	2	$A_2$	$N_2$	6
$A_3N_2 \rightarrow$	3	$A_3$	$B_1$	4
	4	$B_1$	\$	0
	5	$N_1$	$A_2$	2
	6	$N_2$	$A_3$	3



# EXACTMATCH algorithm

$$\Sigma = B_1A_1N_1A_2N_2A_3.$$

Pattern to find,  $P = \text{NAN}$

	0	\$	$A_1$	1
	1	$A_1$	$N_1$	5
	2	$A_2$	$N_2$	6
	3	$A_3$	$B_1$	4
$B_1A_3N_2 \rightarrow$	4	$B_1$	\$	0
	5	$N_1$	$A_2$	2
$N_2A_2N_1 \rightarrow$	6	$N_2$	$A_3$	3

# Why and What is Bowtie?

- ▶ Exactmatching performs poorly for DNA short read alignment because of mismatches from sequencing errors and other reasons.
- ▶ New alignment algorithm which conducts a backtracking search to quickly find alignments that satisfy a specified alignment policy.
- ▶ Each character in a read has a numeric quality value( $m_i$ ), with lower values indicating a higher likelihood of a sequencing error.
- ▶ We allows a limited number of mismatches, while trying to minimize  $\sum_i m_i$ , where  $i$  spans over all mismatches.

Bowtie is a quality-aware, greedy, randomized, depth-first search through the space of possible alignments.