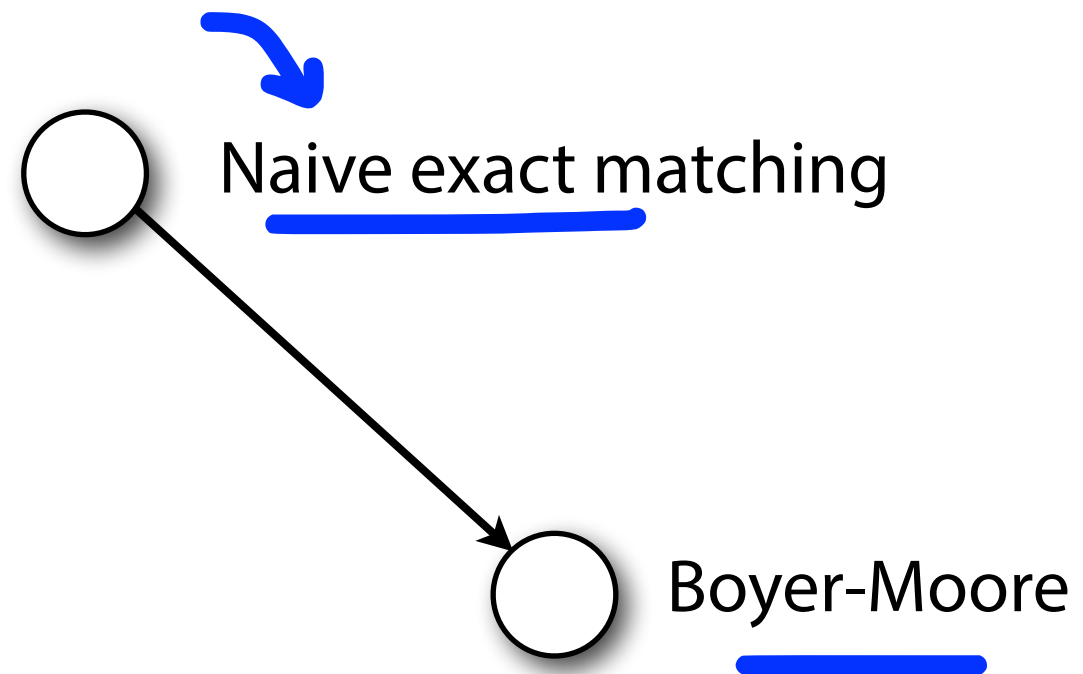


~~Boyer-Moore~~



Exact matching: better naïve algorithm

P : word

T : There would have been a time for such a word

..... word



u doesn't occur in P , so we can skip next two alignments

P : word

T : There would have been a time for such a word

..... word

word skip!
word skip!
word



Boyer-Moore

Learn from character comparisons to skip pointless alignments

Try alignments in left-to-right order, and try character comparisons in right-to-left order

P : word

T: There would have been a time for such a word

A diagram showing the word "word" on a horizontal dashed line. The letters are colored: 'w' is grey, 'o' is red, 'r' is green, and 'd' is grey. A dashed arrow points to the left below the 'o', and a solid arrow points to the right at the end of the line.

Boyer, RS and Moore, JS. "A fast string searching algorithm."
Communications of the ACM 20.10 (1977): 762-772.

Boyer-Moore: Bad character rule

Upon mismatch, skip alignments until (a) mismatch becomes a match, or (b) P moves past mismatched character

Step 1:

T : G C T T C T G C T A C C T T T T G C G C G C G C G G A A
 P : C C T T T T G C

Step 2:

T : G C T T C T G C T A C C T T T T G C G C G C G C G G A A
 P : C C T T T T G C

Step 3:

T : G C T T C T G C T A C C T T T T G C G C G C G C G G A A
 P : C C T T T T G C

Boyer-Moore: Good suffix rule

Let t = substring matched by inner loop; skip until (a) there are no mismatches between P and t or (b) P moves past t

