Definitions

Token: One unique unit of information

- example: `A`

Vocabulary: A set of tokens

- example: `{A, B, C, ...}`

Token Sequence: An ordered list of tokens from a vocabulary

- example: `[A, D, C, C, A, B]`

Path: A sequence of actions that can be iteratively applied to a token sequence

- example: `[REPLACE(0, D), INSERT(2, A), DELETE(3)]`

Actions

Token Replacement: Set the token at a specific position to a new token

- example:
  - `[A, B, C, D]`
  - `REPLACE(0, D)`
  - `[D, B, C, D]`

Token Insertion: Insert a new token at a specific position

- example:
  - `[A, B, C, D]`
  - `INSERT(2, A)`
  - `[A, B, A, C, D]`

Token Deletion: Delete a token at a specific position

- example:
  - `[A, B, C, D]`
  - `DELETE(3)`
  - `[A, B, C]`

Assumptions

- The size of the vocabulary is on the order of 10,000 to 100,000 tokens
- Sequence lengths are on the order of 100 to 1,000 tokens

Problem

Consider an initial sequence S

Consider a target sequence T

Let x be path such that applying x to S results in T

Let X be the set of all possible paths x

Define a path $x^$ *to be optimal if* $|x^| \leq |x|$ for all $x \in X$

Let A be the set of actions containing the first action of all optimal paths

What algorithm can most efficiently find A given S and T?