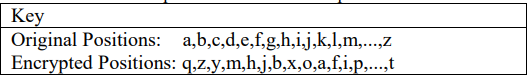
Genetic Algorithm in Breaking Substitution Cipher

1. Substitution Cipher

Thay đổi từng kí tự (không phải kí tự đặc biệt) bằng 1 kí tự khác dựa theo 1 map table

Example:





1. Genetic Algorithm

1, Markov model:

-một mô hình ngẫu nhiên mô tả 1 tập các sự kiện có thể 1 cách tuần tự khi mà xác xuất xảy ra của trạng thái tiếp theo chỉ phụ thuộc vào trạng thái trước đó.

-nói cách khác, xác suất tại bước thứ n+1 chỉ phụ thuộc vào bước thứ n chứ k phải bước nào trước n.

2, Markov assumption

-cung cấp nền tảng cho việc mô hình hoá việc ra quyết định trong các tình huống mà kết quả là một phần ngẫu nhiên và 1 phần dưới sự điều khiển của người ra quyết định

-hữu dụng trong việc giải quyết 1 bài toán bằng quy hoạch động và học tang cường

3, N-gram:

-N-gram là quá trình phân tách 1 đoạn văn bản của mỗi N kí tự liên tiếp làm thành 1 cụm từ.

-một cách đơn giản, chúng ta có thể lấy N-1 gram và dự đoán từ cuối cùng dựa trên xác xuất phụ thuộc vào bản gốc N-gram

-chúng ta sẽ dung bigram và unigram để xây dựng markov model

4, [Chain rule](https://en.wikipedia.org/wiki/Chain_rule_(probability)):

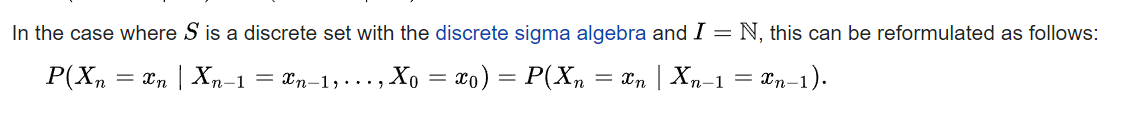
Text

Description automatically generated

-ví dụ: gọi p(word) là xác suất để từ “word” có thể được phân tách như sau:

p(word) = p(d | wor) \* p(r | wo) \* p(o | w) \* p(w)

= p(d | r) \* p(r | o) \* p(o | w) \* p(w) ([markov assumption](https://en.wikipedia.org/wiki/Markov_property))



Bước 1: chọn cấu trúc NST biểu diễn lời giải bài toán:

dna\_Set

Bước 2: khởi tạo tập lời giải ban đầu

Hàm \_\_init\_\_()

Bước 3: chọn hàm đánh giá mức độ tốt của lời giải:

Hàm population\_Score()

Bước 4: thiết kế toán tử di truyền (toán tử đột biến)

Hàm next\_Gen()

Bước 5: xác định các tham số cho bài toán:

num\_Iters, num\_Children, population

update\_prob\_mats\_with\_sents: keep pieces of the “higher likelihood score” maps and change the “lower likelihood score” maps

get\_word\_prob: sử dụng hàm log để chống lại các xác suất rất nhỏ vì không gian mẫu rất lớn

keepdims=true: return a 2d array

np.sum(axis=1, keepdims=true): sum of each rows, then display each sum as a 1x1 matrix

A picture containing timeline

Description automatically generated