Edit Distance con e Senza N-Grams

Niccolò Parlanti

Ottobre 2021

1 Introduzione

Lo scopo dell'esercizio è trovare all'interno di un lessico la parola più vicina alla parola data. Per fare ciò sarà necessario utilizzare la funzione di Edit Distance vista a lezione. Andremo poi ad analizzare anche le differenze tra l'uso e il non uso degli n-gram all'interno della ricerca. Per poter fare questo studio verranno creati alcuni test specifici, ognuno dei quali sarà poi analizzato su i diversi dizionari di parole che ho scelto, da 1000, 60000 e 280000 parole.

2 Teoria

2.1 Edit Distance

L'Edit Distance è un algoritmo di programmazione dinamica che consente di trovare, dato un set di operazioni con dei costi, il minor costo di trasformazione di una parola x in un'altra y. L'utilizzo dell'algoritmo Edit Distance nella correzione ortografica riguarda i campi più svariati, a partire dal suo impiego nella correzione di parole scritte in modo errato all'interno di documenti, fino a suggerire le query all'utente che desidera cercare qualcosa nel web.

Operazioni

Copia il costo è 0 poiche si tratta di lasciare immutato l'elemento in x.

Sostituzione costo 1 poiche corrisponde a sovrascrivere un carattere in x con quello rispettivo in y.

Cancellazione costo 1, si elimina un carattere.

Inserimento costo 1, si inserisce un carattere.

Scambio costo 1, si invertono due caratteri adiacenti.

2.2 Weighted Edit Distance

Corrisponde all'edit Distance introdotto nel punto precedente, con la differenza che in questa variante il costo delle operazioni non dipende solo dalla sua tipologia, ma anche dai caratteri coinvolti, questo aiuta ad adattare l'algoritmo al contesto nel quale viene utilizzato, ad esempio se è implementato in un motore di ricerca è logico attribuire un costo minore alla sostituzione tra due lettere adiacenti nella tastiera.

2.3 Intersezione N-Gram

L'intersezione con gli N-Gram è un approccio che aiuta a diminuire il numero di parole da confrontare con l'edit distance. Data la parola Q da cercare all'interno di L, divido in n-gram la parola Q con n scelto precedentemente, e cerco per ogni parola di L se possiede un numero considerevole di n-gram comuni a quelli di Q in base ad un coefficente chiamato *Coefficiente di Jaccard*, ovvero il quoziente tra la dimensione dell'insieme intersezione e quella dell'insieme unione. Su questo insieme di parole posso quindi calcolare l'Edit Distance e selezionare la parola più vicina.

Si calcola: $JC = \frac{X \cap Y}{X \cup Y}$.

Nel caso che stiamo considerando gli insieme sono gli N-Gram, e il coefficente di Jaccard minimo che accettiamo è $JC \geq 0,8$

3 Confronto Tra Edit Distance Con e Senza N-Grams

3.1 Aspettative

Prima di studiare i risultati pratici che l'esecuzione del programma ci potrà mostrare, possiamo già fare un'analisi riguardo al comportamento che ci aspettiamo. Analizzando la teoria possiamo facilmente dedurre il risultato, poiche l'esecuzione di edit distance senza intersezione N-Gram confronta ogni singola parola del dizionario; anche quelle che risultano decisamente diverse dalla parola considerata. Perciò quello che ci aspettiamo di vedere sono dei tempi dell'algoritmo che utilizza gli N-Gram decisamente minori rispetto a quello che non gli utilizza.

3.2 Descrizione Esperimenti

L'esperimento condotto ha lo scopo di capire quale sia l'andamento dei tempi di ricerca della parola più vicina all'interno di un lessico L ad una certa parola Q data, utilizzando o meno la logica degli N-Gram, ed in aggiunta il coefficente di Jaccard per verificare se tale parola è candidata ad essere la migliore oppure no. Per ognuno dei dizionari presi in considerazione, rispettivamente di 1000,

60000 e 280000 parole, si sceglie randomicamente una parole, e la si testa, utilizzando algoritmi con e senza N-Gram, prima in forma normale, e poi con alcune modifiche:

Aggiunta aggiunta alla parola di una lettera casuale in posizione casuale.

Rimozione viene rimossa dalla parole una lettera casuale.

Swap scambio di posizione tra due lettere casuali all'interno della parola.

I rislutati presentati sono ottenuti eseguendo più volte il singolo test e facendo una media dei valori finali.

4 Test

4.1 Nessuna modifica

Di seguito sono riportati i tempi di esecuzione degli algoritmi senza alcuna modifica della parola presa casualmente all'interno dei dizionari.

4.1.1 Dizionario da 1000 parole

Tempi con il dizionario da 1000 parole	
Senza N-Gram	0.178737
2-Gram	0.009395
3-Gram	0.008632
4-Gram	0.008234

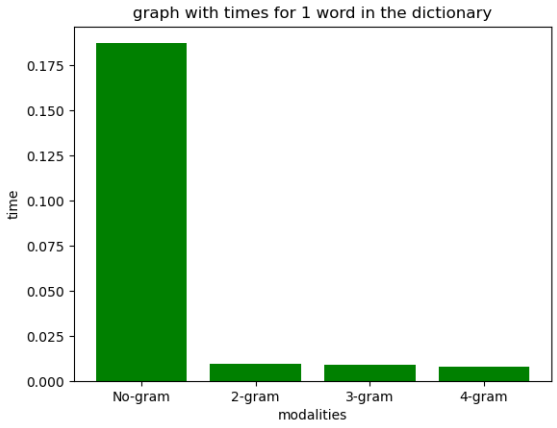

Figure 1: Tempi con una parola nel dizionario

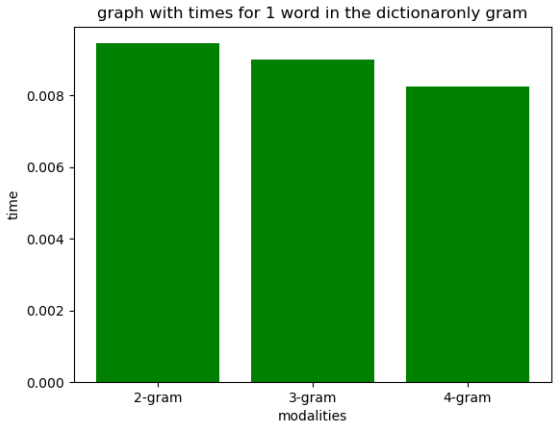

Figure 2: tempi con una parola nel dizionario, confronto tra N-Grams

4.1.2 Dizionario da 60000 parole

Tempi con il dizionario da 60000 parole	
Senza N-Gram	17.609933
2-Gram	0.557127
3-Gram	0.516811
4-Gram	0.485277

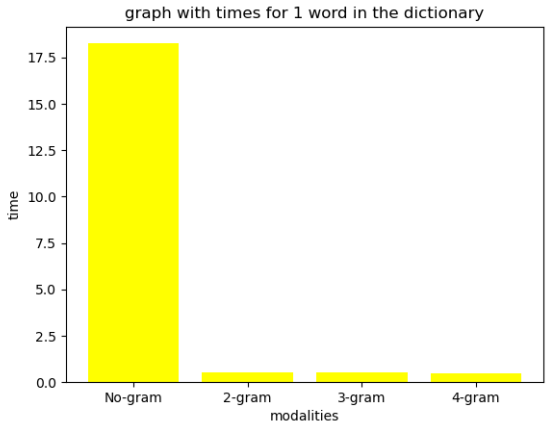

Figure 3: Tempi con una parola nel dizionario

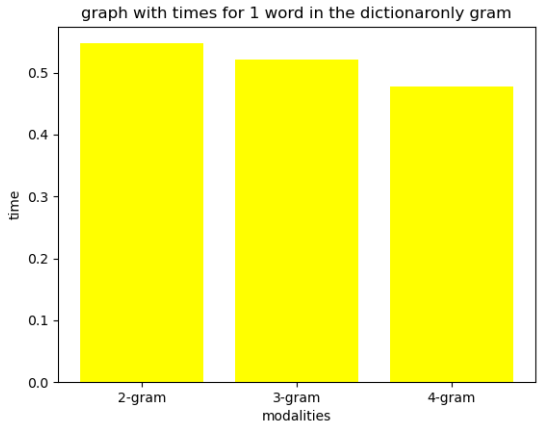

Figure 4: Tempi con una parola nel dizionario, confronto tra N-Grams

4.1.3 Dizionario da 280000 parole

Tempi con il dizionario da 280000 parole	
Senza N-Gram	104.526367
2-Gram	2.637852
3-Gram	2.476425
4-Gram	2.390245

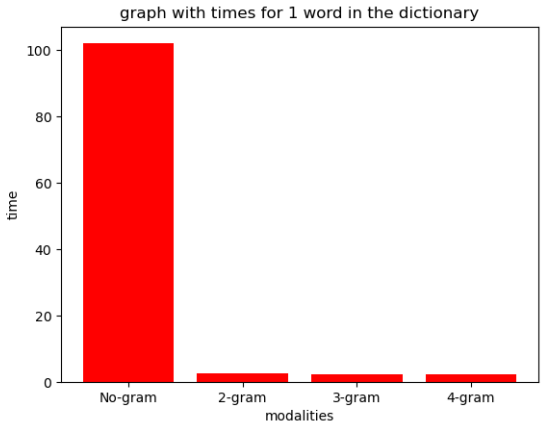

Figure 5: Tempi con una parola nel dizionario

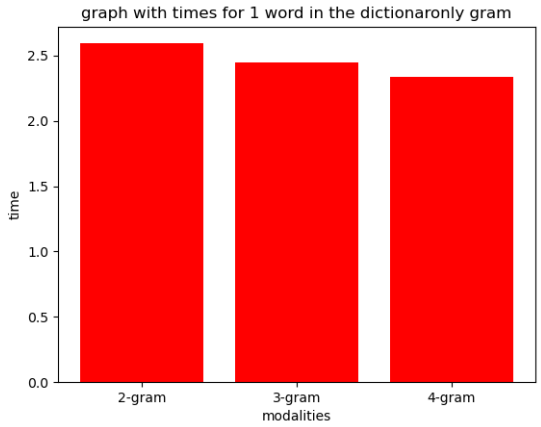

Figure 6: Tempi con una parola nel dizionario, confronto tra N-Grams

4.2 Parola con una lettera aggiunta

Di seguito sono riportati i tempi di esecuzione degli algoritmi con una lettera generata casualmente, e aggiunta alla parola in posizione casuale.

4.2.1 Dizionario da 1000 parole

Tempi con una lettera aggiunta	
Senza N-Gram	0.262535
2-Gram	0.011175
3-Gram	0.010359
4-Gram	0.010279

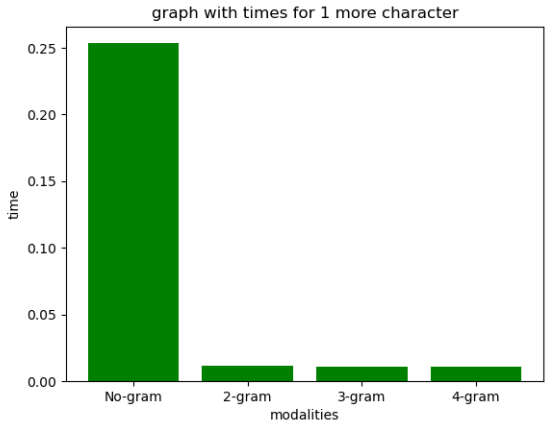

Figure 7: Tempi con una lettera aggiunta

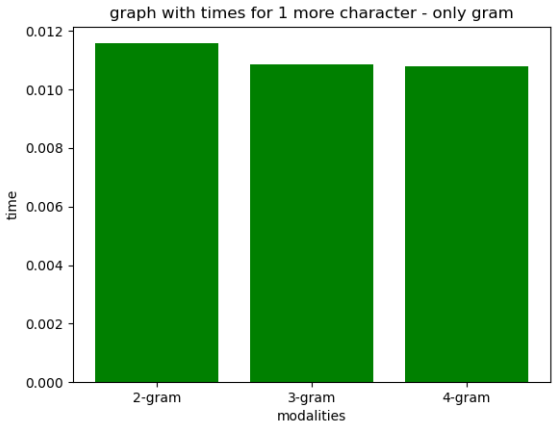

Figure 8: Tempi con una lettera aggiunta, confronto tra N-Grams

4.2.2 Dizionario da 60000 parole

Tempi con una lettera aggiunta	
Senza N-Gram	17.761934
2-Gram	0.546785
3-Gram	0.493999
4-Gram	0.436285

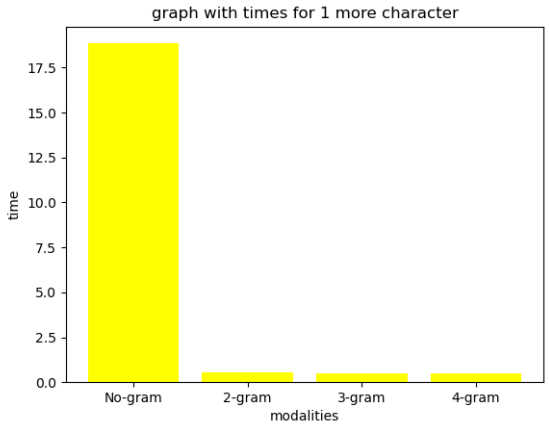

Figure 9: Tempi con una lettera aggiunta

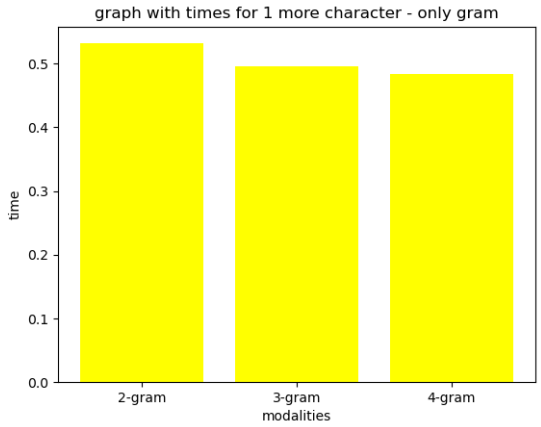

Figure 10: Tempi con una lettera aggiunta, confronto tra N-Grams

4.2.3 Dizionario da 280000 parole

Tempi con una lettera aggiunta	
Senza N-Gram	133.959583
2-Gram	2.505311
3-Gram	2.341726
4-Gram	2.267408

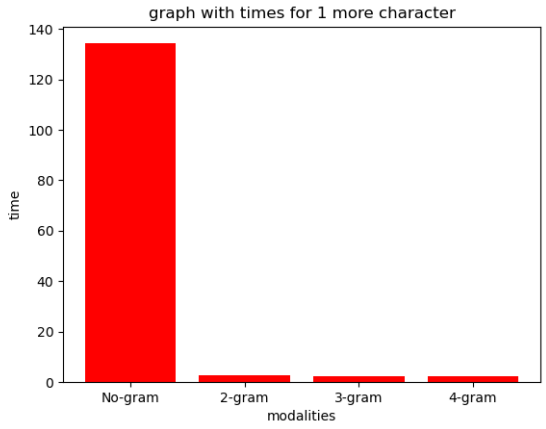

Figure 11: Tempi con una lettera aggiunta

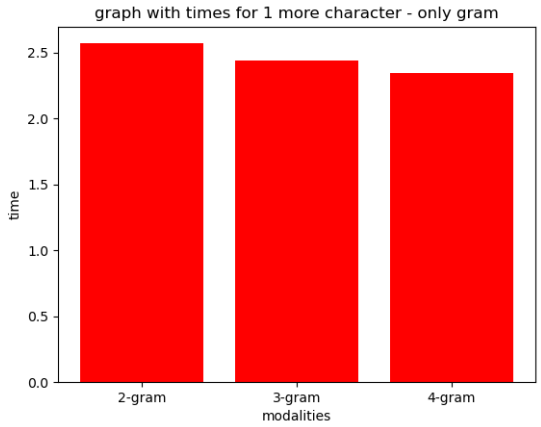

Figure 12: Tempi con una lettera aggiunta, confronto tra N-Grams

4.3 Parole con una lettera rimossa

Di seguito sono riportati i tempi di esecuzione degli algoritmi con una lettera rimossa in maniera casuale dalla parola.

4.3.1 Dizionario da 1000 parole

Tempi con una lettera rimossa	
Senza N-Gram	0.282993
2-Gram	0.011296
3-Gram	0.011561
4-Gram	0.008114

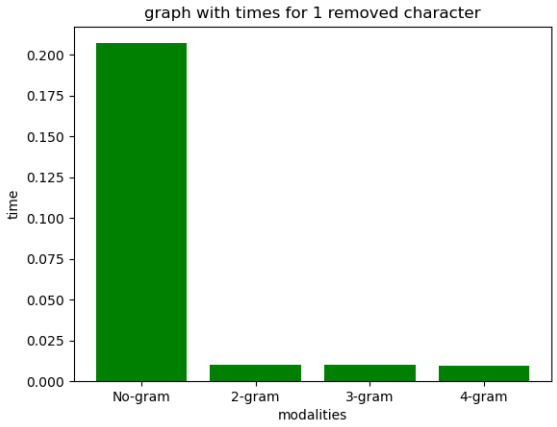

Figure 13: Tempi Con Una Lettera Rimossa

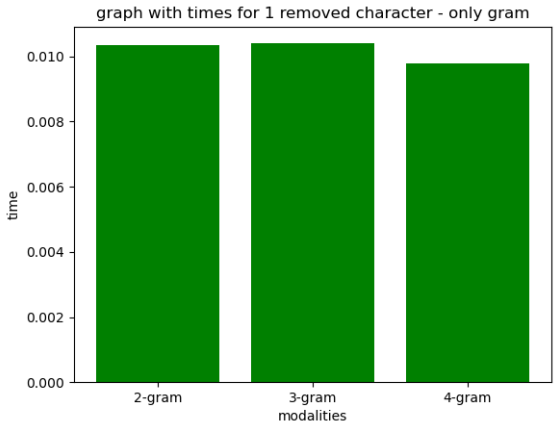

Figure 14: Tempi con una lettera rimossa, confronto tra N-Grams

4.3.2 Dizionario da 60000 parole

Tempi con una lettera rimossa	
Senza N-Gram	15.309161
2-Gram	0.514623
3-Gram	0.483778
4-Gram	0.454593

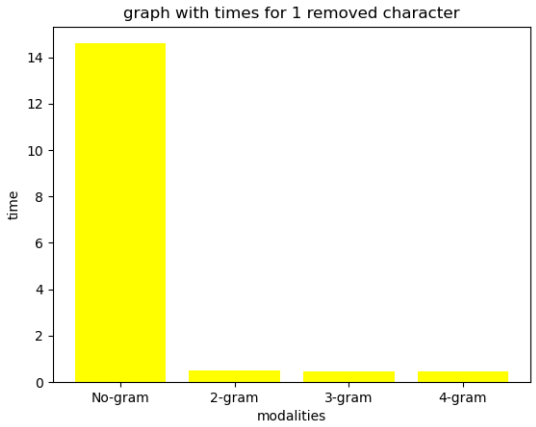

Figure 15: Tempi con una lettera rimossa

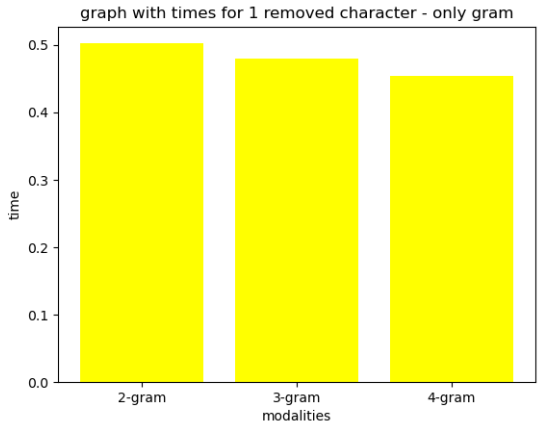

Figure 16: Tempi con una lettera rimossa, confronto tra N-Grams

4.3.3 Dizionario da 280000 parole

Tempi con una lettera rimossa	
Senza N-Gram	112.209150
2-Gram	2.716297
3-Gram	2.459319
4-Gram	2.353745

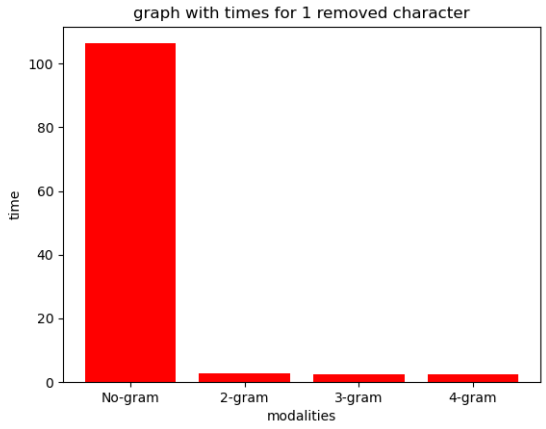

Figure 17: Tempi con una lettera rimossa

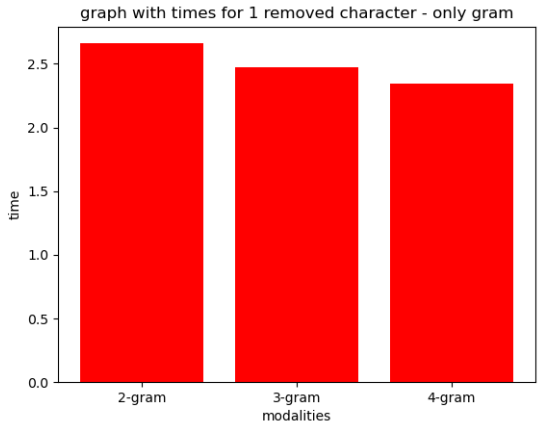

Figure 18: Tempi con una lettera rimossa, confronto tra N-Grams

4.4 Parola con lettere scambiate

Di seguito sono riportati i tempi di esecuzione degli algoritmi con due lettere casuali della parola scambiate tra loro.

4.4.1 Dizionario da 1000 parole

Tempi con due lettere scambiate	
Senza N-Gram	0.280179
2-Gram	0.008948
3-Gram	0.008502
4-Gram	0.007991

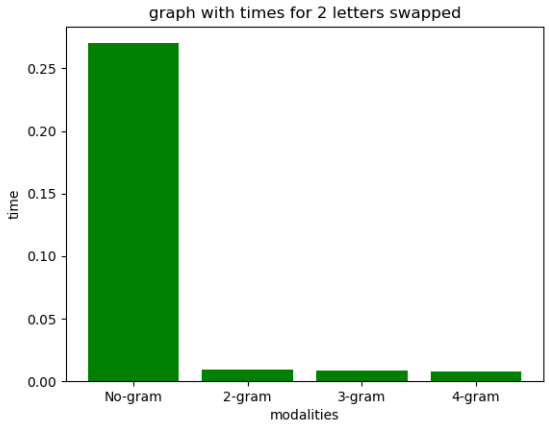

Figure 19: Tempi con due lettere scambiate

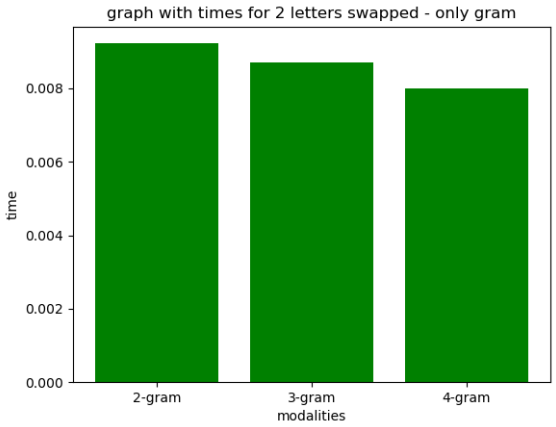

Figure 20: Tempi con due lettere scambiate, confronto tra N-Grams

4.4.2 Dizionario da 60000 parole

Tempi con due lettere scambiate	
Senza N-Gram	17.711446
2-Gram	0.530106
3-Gram	0.487153
4-Gram	0.458731

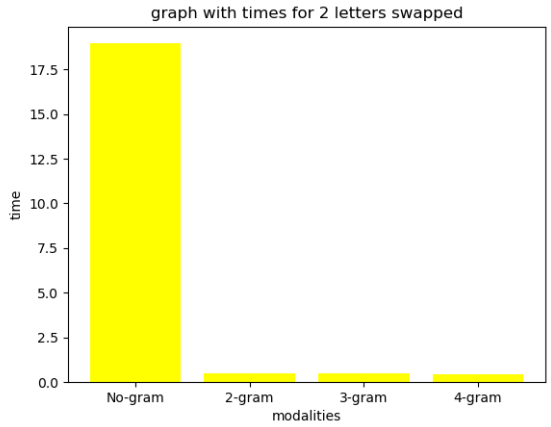

Figure 21: Tempi con due lettere scambiate

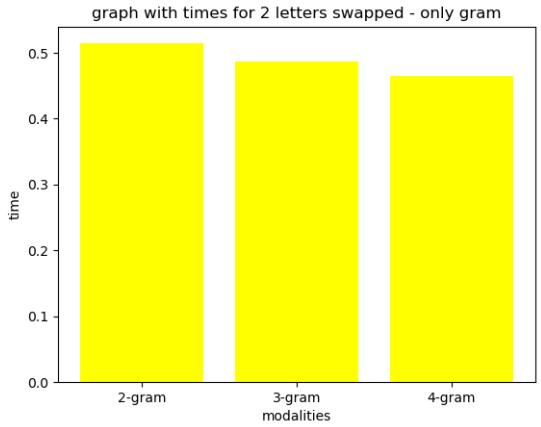

Figure 22: Tempi con due lettere scambiate, confronto tra N-Grams

4.4.3 Dizionario da 280000 parole

Tempi con due lettere scambiate	
Senza N-Gram	87.806007
2-Gram	2.580576
3-Gram	2.3936613
4-Gram	2.257300

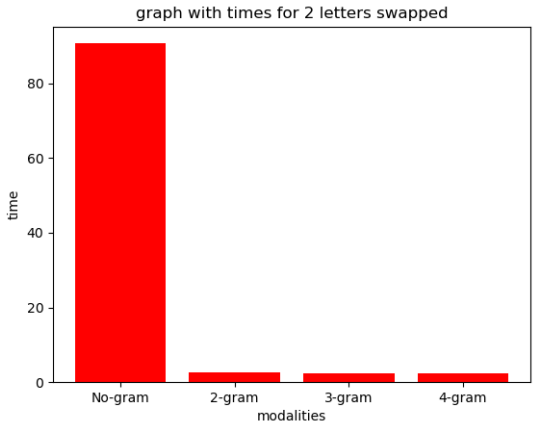

Figure 23: Tempi con due lettere scambiate

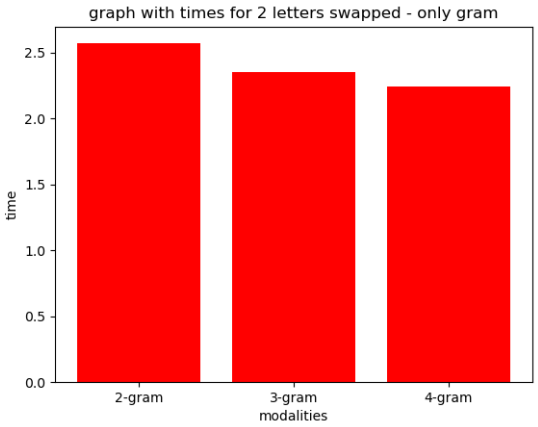

Figure 24: Tempi con due lettere scambiate, confronto tra N-Grams

5 Conclusioni

Come previsto nei paragrafi precedenti, la differenza di velocità di esecuzione tra l'approccio con e senza l'utilizzo degli N-Gram è molto ampia, ed aumenta sempre di più all'aumentare del numero di parole nel dizionario. Questo è dovuto, come spiegato anche in precedenza, all'analisi degli N-Grams che grazie al coefficiente di Jaccard permettono di escludere le parole che non sono minimamente simili a quella che stiamo considerando, e velocizzare di molto il processo di controllo delle parole.
La grandezza dell'N-Gram influisce in maniera marginale rispetto all'esecuzione senza di esso; possiamo notare che in quasi tutti i casi aumentando la grandezza dell'N-Gram diminuiscono anche i tempi di esecuzione.