Relazione su Edit distance

$Autore : Georgiev\ David\ mat : 1043306$

Introduzione

In questa relazione vengono esaminate le prestazioni di un algoritmo che calcola la distanza di edit tra due stringhe considerando solo due operazioni possibili, ovvero la cancellazione di un carattere oppure l'inserimento di uno. In particolare viene messa in evidenza l'utilità della programmazione dinamica che è fondamentale per un implementazione efficiente di questo algoritmo.

Ciò che ottimizza l'algoritmo è il meccanismo di memoization; la matrice che memorizza i risultati intermedi impedisce alle chiamate ricorsive, chiamate sulle stesse sotto-stringhe, di ripetersi più di una volta e quindi ridurre drasticamente il numero di chiamate ricorsive totali effettuate dall'algoritmo. Questo miglioramento del calcolo in termini di tempo è pagato con un leggero costo temporale per l'inizializzazione della matrice e con lo spazio in memoria che essa occupa però questi compromessi sono di gran lunga preferibili al ricalcolo di molte sub-routine tutte uguali fra di loro.

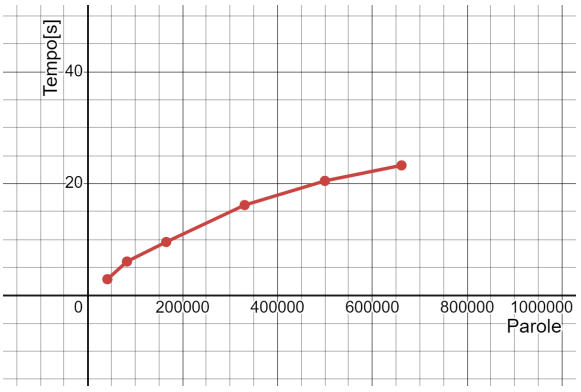
Analisi dei tempi di calcolo

L'algoritmo è stato esaminato su un testo da correggere e un dizionario da cui controllare la correttezza ortografica di tutte le parole. Sono stati eseguiti test variando la lunghezza del dizionario mantenendo lo stesso testo da correggere (vedi figura 1), in questo caso si può notare come il tempo di esecuzione aumenti col aumentare del numero di parole nel dizionario semplicemente perché l'algoritmo viene chiamato più volte per ogni parola da esaminare nel dizionario. Da notare che il numero di parole errate aumenta perché il dizionario contiene meno parole, la lista di parole corrette con edit distance minima varia anche in base alla dimensione del dizionario.

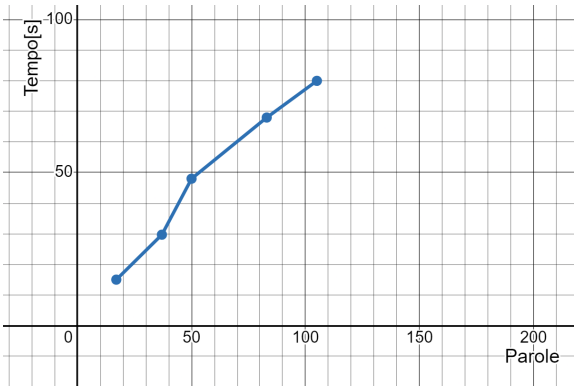
Il secondo tipo di test eseguito è stato eseguito variando il numero di parole nel testo da correggere e mantenendo invariata la dimensione del dizionario (vedi figura 2). Inoltre la proporzione di parole ortograficamente sbagliate è stata tenuta fissa per ogni test eseguito (circa il 25% delle parole nei vari testi

conteneva errori). Come si può notare aumentando il numero di parole nel testo si ha un incremento maggiore rispetto all'aumentare le parole nel dizionario ma questo è dovuto principalmente alla piccola dimensione del testo usato nel primo test rispetto alla dimensione del dizionario usato nel secondo test. Inoltre si osservi che non sempre le parole trovate con edit distance minima da quella errata possono essere parole adatte a essere inserite nella frase al posto dell'errore. (Es. squola ha edit distance uguale a 1 da suola e 2 da scuola).

Queste due osservazioni suggeriscono che questa applicazione sarebbe inefficace se usasse tutto il dizionario italiano per la correzione di testi e sarebbe meglio utilizzare strategie alternative sempre basate sul calcolo del edit distance.



1. Edit distance nel variare della dimensione del dizionario



2. Edit distance nel variare della dimensione del testo

Osservazioni aggiuntive

Provando a eseguire l'applicazione con un algoritmo ricorsivo che non fa uso della programmazione dinamica le prestazioni calano notevolmente e non si riesce a ottenere una risposta in tempo ragionevole. Infatti un algoritmo del genere effettua molte chiamate ricorsive di cui si è già calcolato il risultato in precedenza e già con una edit distance di 20 caratteri questo algoritmo impiega tempi enormi per concludere l'esecuzione. (tempi relativi alla macchina ma comunque molto più grandi rispetto all'alternativa realizzata con la programmazione dinamica).