CS 4TB3: Approximate Regular Expressions

Rumsha Siddiqui, Tasnim Noshin, and Umme Salma Gadriwala

March 15, 2019

1 Description

A regular expression is a special text string for describing a search pattern. Let n be the length of the text, m be the length of a regular expression R for the alphabet $\sum$. Further, let d be the number of strings in R, where a string is a sequence of characters connected by concatenation.

The traditional technique to search for an exact regular expression in a text uses $O(mn)$ worst case search time with a space requirement of $O(m)$ states, by converting R into a non-deterministic finite automaton (NFA). An alternative algorithm converts the NFA into a deterministic finite automaton (DFA), and uses $O(2^m)$ states and $O(n)$ search time.

Another interesting problem is approximate regular expression matching, that is searching for a given regular expression in a text allowing a limited number of errors k, where k might be an insertion, a deletion or a substitution of a character by another. There exists a solution for this problem in time $O(mn)$ and a solution for the case $k = 0$ in time $O(dn)$.

To compare the performance of exact regular expression matching to approximate matching, we will implement the NFA-based algorithm and the Myers and Miller's algorithm.

2 Resources

- https://users.dcc.uchile.cl/~gnavarro/ps/wae99.pdf
- https://www.sciencedirect.com/science/article/pii/S1570866712001116?via%3Dihub
- https://www.data-essential.com/approximate-regular-expressions/

3 Division of Work

3.1 Individual Tasks

- Rumsha Siddiqui: Study literature for algorithm comparison and measuring methods; design and run test cases
- Tasnim Noshin: Implement exact regular expression matching algorithm
- Umme Salma Gadriwala: Study literature for algorithm comparison; implement approximate regular expression matching algorithm

3.2 Group Tasks

- Construct hypothesis
- Result analysis
- Final presentation and poster

4 Weekly Schedule

Week	Deliverable
1	Implement algorithms; Design test cases
2	Run test cases; Tabulate results
3	Analyze results; Prepare report
4	Prepare poster and presentation