COMP90049 Project 1

> A Java program to correct words which are mis-spelled, based on Global Edit Distance and Soundex algorithms. This documentation is to discribe the steps.

Getting Start

Take a file 'misspell.txt' with many words which may be misspelt as input. Analyze each one in the file with the dictionary (input as 'dict.txt') via calculating Global Edit Distance and find the word matched best.

```
public static int GED(String target, String misspell) {
        int lq = target.length();
        int lt = misspell.length();
        int[][] F = new int[lq+1][lt+1];
        for(int i=0 ; i<=lq ; i++ ) {
            F[i][0] = i*I;
        }
        for(int j=0 ; j<=lt ; j++ ) {
            F[0][j] = j*D;
        }
        for(int i=1 ; i<=lq ; i++ ){
            for(int j=1 ; j<=lt ; j++){
                int a = F[i-1][j] + I; // insertion
                int b = F[i][j-1] + D; // deletion
                int c = F[i-1][j-1] +
                (target.charAt(i-1) == misspell.charAt(j-1) ? M : R);
                F[i][j] = Math.max(Math.max(a, b), c);
            }
        }
        return F[lq][lt];
    }
```

Increasing Recall

While finding the biggest global edit distance (with [m, i, d, r] = [+1, −1, −1, −1]), try to return all the words which reach the max value instead of returning only the first one. This test increases the recall with sacrificing the precision.

```
for(int i = 0; i < mis.size(); i++){
            int max = Integer.MIN_VALUE;
            ArrayList<String> pickPool = new ArrayList<String>();
            for (int j = 0; j < dict.size(); j++) {
                if (max < GED(dict.get(j) , mis.get(i))) {
                    max = GED(dict.get(j) , mis.get(i));
                }
            }
            for (int j = 0; j < dict.size(); j++) {
```

```
                if (max == GED(dict.get(j) , mis.get(i))) {
                    pickPool.add(dict.get(j));
                }
            }
            corr.add(pickPool);
        }
```

Change Parameters

Change the parameters of Global Edit Distance algorithm to test influence to the precision and recall.

> [m, i, d, r] = [+1, −1, −1, −1]
>
> [m, i, d, r] = [+1, −1, −1, −1]
>
> [m, i, d, r] = [+1, −1, −1, −1]

Import Soundex

Cooperate with Soundex algorithm to try how it will effect the result.

```
for (int j = 0; j < dict.size(); j++) {
    if (max < GED(Soundex(dict.get(j)) , Soundex(mis.get(i)))) {
        max = GED(Soundex(dict.get(j)) , Soundex(mis.get(i)));
    }
}
for (int j = 0; j < dict.size(); j++) {
    if (max == GED(Soundex(dict.get(j)) , Soundex(mis.get(i)))) {
        pickPool.add(dict.get(j));
    }
}
corr.add(pickPool);
```

Store Results

Store results with ***.txt*** file with timestamps to analyze. Including mis-matched lists, matched numbers, returned numbers, correct length and calculated recall and precision.

$$Recall = \frac{Matched\ number}{Total\ number\ of\ relevant\ results}$$ $$Precision = \frac{Matched\ number}{Total\ returned}$$