# Information Retrieval and Extraction

**ASSIGNMENT 1** - Archit Jain (2019101053)

I am planning to add context sensitive spelling correction technique to reduce unrelated search results if they are greater than some limit (can be 10 or more) but now our problem reduced to correct spelling of single word by treating them isolated word.

So I am planning to use both edit distance along with k-gram(here I am using k=2) index but now we have to decide its sequence, here I mean we can used edit distance algorithm for given query word in our sets of word and thinking that first and last alphabets of word are correct and find to filter out more words we will use k-gram technique, for this the selected words must have at least R(starting from 3) bigram other than first and last word this is the one way but it can be time consuming because the set of word will be too large and calculating edit distance for each and then filtering out is expensive so we can swap them to decrease some time but how much it will solely depends on the set words in which we are looking for a correct spell word. I also added a edge condition if a word matches exactly then also I am finding max 5 words which will have edit distance '1' just to add a feature.

Now if result if less than 10 repeat k-gram ans edit diastane with R=2,then 1 than 0

If no result found return "NO RESULT FOUND"

Pseudocode:

```
dictionary_set =[...]
new_set=[...]

def edit_distance(query,required_distance):
    if required_distance==1 :
        return list of all words with edit distance 1
    else :
        return list of words along with edit distance


def k-gram(k,query,R):
    genarate sets of words with k-gram
    make new set with all word have atlest R gram same along woth first and la
st word same
    return the array of words

def init(query):
    if query present in dictionary_set:
        new_set = edit_distance(query,1)
        if length of new_set more than 5 trim it to 5
    else:
        new_set = filter(dictionary_set,(for x in dictionary_set has fisrt and last alphabet same))
        new_set = k-gram(2,query,3)
        new_set = edit_distance(query,-1)
        new_set = jaccard_cooficeint(query)
        if length of new_set more than 10 trim it to 5
        else repeat with R=2 than with R=1 then R=0

        if new_set is empty
            return "NO RESULT FOUND"
    return new_set
```

**ADDING CONTEXT SENSETIVE CORRECTION**

If we have huge list of result we can add a scoring function to calculate which one is better we will prefer max score in our result.

Example like :

query string = a b c d e

let say result for b if 1000 then we can find list of matched words and there distance form b we can use this formula but not sure that its best because it is not derive and pure based on my observation

| | a | b | c | d | e |
|---|---|---|---|---|---|
| Number of Word related | 10 | 1000 | 8 | 20 | 100 |
| distance | -1 | 0 | 1 | 2 | 3 |

```
def score(table with realted word and distance and edit distace for the realted words ) :
    sum_of_edi_distance_of_related_words*log(distance + 1)*(-1)
```