ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 447- 450 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

An Evolutionary approach for Candidate Set generation with String Transformation Technique

[1]B. Srinivas Reddy, **[2]V.Vidyasagar**
[1]Final M.TechStudent ,[2]Associate professor
[1,2]Dept of Computer Science and Engineering
[1,2]Pydah College of Engineering, Visakhapatnam, AP,India

Abstract: String Transformation is still an important research issue in the field of natural language processingand search engine optimization, Even though various traditional approaches available for string transformation they are not optimal because of Accuracy and efficiency are the basic parameters to optimize. While generation of the output Strings, Initially we consider the set of similar keywords. In this paper we are proposing an efficient approach of String transformation with Candidate generation and Selection and Query Reformulation, for the generation of the candidate sets we are proposing an evolutionary approach for the correction of the misspelled words(deletion, insertion or substitution) either by single character or substring.

INTRODUCTION

Thispaper addresses string transformation, which is anessential problem, in many Applications. In naturallanguage processing, pronunciation generation, spellingerror correction, word transliteration, and wordstemming can all be formalized as string transformation.String transformation can also be used in queryreformulation and query operators. Here the strings canbe strings of words, characters, or any type of tokens[1].

Each operator is a transformation rule that defines thereplacement of a substring with another substring. Thelikelihood of transformation can represent similarity,relevance, and association between two strings in aspecific application. Although certain progress has beenmade, further investigation of the task is still necessary,particularly from the viewpoint of enhancing bothaccuracy and efficiency, which is precisely the goal ofthis work.

String transformation [5] can be conducted at two different settings, depending on whether or not a dictionary isused. When a dictionary is used, the output strings must exist in the given dictionary, while the size of the dictionary can be very large. Without loss of generality, we specifically study correction of spelling errors inqueries as well as reformulation of queries in web searchin this paper. In the first task, a string consists ofcharacters. In the second task, a string is comprised ofwords. The former needs to exploit a dictionary while thelatter doesnot. Correcting spelling errors in queriesusually consists of two steps: candidate generation andcandidate selection. Candidate generation is used to findthe most likely corrections of a misspelled word from thedictionary. In such a case, a string of characters is inputand the operators represent insertion, deletion, andsubstitution of characters with or without surroundingcharacters, for example, "a"!"e" and "lly"!"ly"[3].

Obviously candidate generation is an example of stringtransformation. Note that candidate generation isconcerned with a single word; after candidate generation,the words in the context (i.e., in the query) can be furtherleveraged to make the final candidate selection, cf.In modern Information Retrieval (IR), search is formalizedas a problem of document ranking based on degree of match-ing between query terms and document terms. Therefore,how to resolve a mismatch between query terms and document terms becomes one of the biggest challenges for IR. Forexample, if a document contains New York Times" whilethe user types many times", typically the document would notbe retrieved at a search system. Spink et al. [2] observethat users have to reformulate their search queries 40% to52% of the time in order to find what they want. In fact,many ill-formed queries can be found from the query logs of web search engines. Queries may contain misspelled words,mistakenly split words, or mistakenly merged words.

More-over, queries may include phrases that should be quoted (using the quotation operator), words that should be properly stemmed, or acronyms that should be expanded.The question then becomes whether we can offer a solutionduring search which automatically reformulates queries, inorder to better represent users' search needs and help users more easily find the relevant information. This is what we mean by query refinement as address in this paper. Notethat for simplicity we only consider replacing the originalquery with the refined query in search (e.g., changing many times" to \new York

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 447- 450 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

times" and searching with it), but not combining the two.There is previous work on query refinement. However,the issue was tackled in separate tasks or by employing generative models. For example, Li et al. conducted spellingerror correction for web search by using a Maximum Entropy model as well as the Source Channel model. Penget al. performed automatic word stemming for web searchby means of a Statistical Language model [7].

RELATED WORK

Even though various approaches available for string transformation they are not optimal in terms of accuracy, traditional approaches like levenshtein distance measure is a time complexity issue because performs the distance with all the keywords in the dictionary and in the log liner model it compares the occurrences of the characters in the source string and target string but not the indexes of the characters

Spelling error correction normally consists of candidate generation and candidate selection. The formertask is an example of string transformation. Candidategeneration is usually Sometimes a dictionary is utilizedin string transformation in which the output strings mustexist in the dictionary, such as spelling error correction,database record matching, and synonym mining. In thesetting of using a only concerned with a single word. Forsingle-word candidate generation, a rule-based approachis commonly used. The use of edit dictionary, we canfurther enhance the efficiency. Specifically, we index the dictionary in a tree, such that each string in the dictionarycorresponds to the path from the root node to a leaf node [4][6].

When we expand a path (substring) in candidate generation, we match it against the tree, and see whether the expansions from it are legitimate paths. If not, we discard the expansions and avoid generating unlikely candidates. In other words, candidate generation is guided by the traversal of the tree.

Query reformulation involves rewriting the original query with its similar queries and enhancing the effectiveness of search. Most existing methods manage to mine transformation rules from pairs of queries in the search logs. One represents an original query and the first identifies phrase-based transformation rules from query pairs, and then segments the input query into phrases, and generates a number of candidates based on substitutions of each phrase using the rules. The weights of the transformation rules are calculated based on log likelihood ratio. A query dictionary is used in this case [8][9].

Query refinement is a problem already identified and studied in IR. Much of the previous work, however, only focused on one task of refinement or resorted to generative models. Li et al. proposed a method for spelling error correction by using a Maximum Entropy (ME) models well as the Source Channel model. They utilized distributional similarities between the query word and its correction candidate as features in the ME model. Cucerzan and Brilladdressed a more generalized spelling correction task using a Source Channel model and query log data. They madeuse of bigrams as the source model and Weighted Edit Distance as the channel model. However, it is less possible to extend their approach to handle other alteration types, e.g. phrase segmentation. Peng et al. proposed a method for conducting stemming on head words of queries. They employed a Statistical Language model in stemming candidate selection. Risvik et al. proposed a method for phrase segmentation using the so-called convexity measures". A\segmentation score" is computed from convexity values and used as a criterion for segmentation. (See also [1][5][3]).

PROPOSED WORK

In this paper we are proposing an efficient string transformation technique with identification of possible correct keywords and extraction of the likely keywords from the dictionary then finds the edit distance over likely keywords. Query reformulation feature enhanced by providing elimination word bag and for optimal results we proposed index based comparison for retrieve the top candidate sets for input query.

Accuracy can be calculated in terms of correct number of candidate set generations for a user query or keyword with previous approach and proposed approach by using graphical user representation. Number of candidate sets can be generated for the user query, Initially making the corrections in misspelled user query or keyword by the evolutionary approach, In this approach random characters or substring can be substituted which are available in dictionary dataset .

Index based comparison

In the traditional approach source string and target string can be compared with occurrence of the character not with index based ocuurence,so in this project we are comparing every character in source string and target string should be identical with respect to index and initial priority given to highest order of dictionary target string

Evolutionary Approach

Input: Input Source string 'S'

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 447- 450 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

Dict_words D (w_1,w_2…….w_n)

Likely_set (l_1,l_2……...l_n)

Output:

Candidateset_ ListC (c_1,c_2……c_n)

Step1: Find likely keywords for source string and compute edit distance

Step2: Get minimum number of edit operations with respect to likely strings (l_1,l_2……..l_n)and input string

Step3: if (min_editdistance<= Threshold value)

Add $l_i$to C.

Step4: Compute Query _Reformulation (S,W_i)

Step5: for i=0 ; i<D.length ; i++

Counter=0;

If String_diff (S,W_i) < = threshold

Add Wi to C

Next

Step 6: Store candidate sets for 'S'.

The following pseudo code shows index based comparison and maintains the order with respect to source and target string

Index Based Comparison

String_Diff(S,W)

{

Compare counter :=0

For i=0; i<S.length;i++

If S[i]== W[j] Then

Compare_counter:+1

Next

If Comparecounter> Threshold value then

Add order wise W_i to C

}

To generate more accurate and efficient candidate sets, we are performing index based comparison between source string and target string, it compares individual character along with their index, if both are equal it will set to '1',continues this process until it reaches source string maximum size

Query Reformulation

In Query Reformulation, We can generate the output strings for the input String "IEEE" as Institute of Electrical and Electronics Engineers by maintain the semantics of the respective keywords to generate the optimal set of keywords, after the generation of the keywords semantics of the respective keywords also integrated to existing the output strings.

In traditional query reformulation technique it simply tokenize the string and compares with respective keywords but fails with additional set of words like articles, prepositions etc. ,to resolve this issue we are maintain word bag to eliminate the unnecessary keywords

Query Reformulation

Void Query _Reformulation (S,W_i)

{

T : =W_i.gettokens()

Count=0;

forint i=0;i<s.length;i++

if T[i] not available in Eliminate words Then

if S[i]== T[i] then

Count :=+1;

End if

End if

Next

If count==T.count then

Add Wi to C.

}

Accuracy Computation

Accuracy can be calculated in terms of correct number of candidate set generations for a user query or keyword with previous approach and proposed approach by using graphical user representation.

ISSN 2278-3091

International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 447- 450 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

CONCLUSION

We are concluding our research work with efficient string transformation technique by generating accurate and more number of candidate sets for input query through index based comparison. Query reformulation can be efficiently handled with string tokenization and words of bag,our experimental results shows optimal results than traditional approaches.

REFERENCES

[1] F. Ahmad and G. Kondrak. Learning a spelling error model from search query logs. In Proceedings of EMNLP 2005,pages 955–962, 2005.

[2] D. Beeferman and A. Berger. Agglomerative clustering of a search engine query log. In Knowledge Discovery and Data Mining, pages 407–416, 2000.

[3] S. Bergsma and Q. I. Wang.Learning noun phrase query segmentation. In Proceedings of EMNLP-CoNLL 2007, pages 819–826, 2007.

[5]Top K Pruning Approach to String TransformationA. Meenahkumary

[4]A Unified and Discriminative Model for Query RefinementJiafengGuo \6] S. Cucerzan and E. Brill. Spelling correction as an iterativeprocess that exploits the collective knowledge of web users.In Proceedings of EMNLP 2004, pages 293–300, 2004.

[7] A. Feuer, S. Savev, and J. A. Aslam.Evaluation of phrasalquery suggestions.In Proc. of CIKM '07, November, 2007.

[8] W. Frakes and R. Baeza-Yates. Information Retrieval:Data Structures & Algorithms. Prentice Hall, EnglwoodCliffs, New Jersey, 1992.

[9] K. Jarvelin and J. Kekalainen.Cumulated gain-basedevaluation of ir techniques. ACM Trans. Inf. Syst.,20(4):422–446, 2002.