Improvement of Searching Process of Homophone Family Words for a Given English Word from the Constructed Semantic Network Map of Homophones

Vimal P.Parmar [*], Dr. CK Kumbharana [**]

[*] Research Scholar, Dept. of Comp. Sci., Saurashtra University, Rajkot. Gujarat, INDIA
[**] Head, Guide, Department of Computer Science, Saurashtra University Rajkot., Gujarat, INDIA

Abstract- Semantic map is defined as a complex network of words or phrases which are related in predefined way. To search related words or phrases for a given English word from a large database of English language is a massy process of comparison for a computer. Each time to search desired set of words or phrases requires amount of computer processing if the search database is large enough and it is necessary to obtain some kind of solution that makes searching efficient and making maximum usage of the computer. The words and phrases may be related using synonyms, antonyms or homophones which are the words which have similar pronunciation but different spellings and meanings. In this paper researcher has designed and obtain a solution model to search set of homophones for a given English word from a large database of English words. The fast retrieval searching process of homophones requires some kind of data structure suitable for fast searching. It is assumed here that the data structure used here is in form of semantic map network which is directly available. Also to test the phonetic equality about the words whether they are homophones or not we require phonetic algorithms. The paper emphasizes on the searching process with model, algorithm for searching process and comparison of benefits compared to direct searching. We use here indexing like mechanism for fast searching.

Index Terms- Semantic map, Searching Operation, Homophone, Homophone family, Algorithms, Time Complexity, Algorithm Efficiency, Searching Efficiency, Phonetic Equality, Pronunciation

I. Introduction

Searching is widely used operation in the world of computing. We use different searching techniques to search something from the internet through search engine or to search information using queries from large database. Thus search operation is weaved a routine life of human being. Computers search amount of data to retrieve desired information from a database having bulk amount of data. Almost searching is at centre in almost every kind of computer application. To search homophone for a given word from a large list of English word database requires any phonetic algorithm to determine phonetic matching. The process requires performing phonetic comparison between the given word and every word in a database. Many algorithms exist to determine phonetic equality. Two words are said to be having same pronunciation if they sound alike and referred to as homophones[8]. This process of searching group of homophones from a large database of English words, is time consuming and wasting of computing utilization. If searching of homophones is performed frequent other solution must be required.

The effort in this research paper is to design a solution to search from a semantic map of homophone words. It is assumed that the semantic map network of homophone words is already constructed. The file structure of this map is known and how to search from this network of word is modeled which proves performance enhancement of searching process. This technique is more efficient as network related all the homophones with one another. The assumed semantic map records indices of every related homophone. This organization of data structure uses two file. One is the merely listing of the words and the second index file is the semantic map knowledge base of homophone. Both the files are accessed in searching process.

The paper is organized from the phonetic algorithms which are used to determine phonetic similarity, introduction to semantic map structure to relate the words in some of the way, semantic map file structure which is available and which will be searched, searching semantic map model, searching from semantic map algorithm implementation and at last conclusion and performance criteria.

II. Introduction of Phonetic Algorithms

Phonetic algorithms determine the phonetic equality among words. Many algorithms are existed for various languages. Few such algorithms are described as follows[3][4].

Soundex algorithm was originally developed by Robert C. Russell and Margaret K. Odell in 1918. This algorithm yields a four character string according to the given English word where the first character is the first alphabet character of the given word and remaining three characters are digits entirely representing the phonetic encoded string which is compared for phonetic equality[5].

Daitch-mokotoff soundex is a modified version of the original soundex algorithm which was named as D-M soundex which was first designed in 1985 by Gary mokotoff and later improved by Randy Daitch to match surnames of Slavic and German languages. This algorithm returns the six digit numeric code for the given word.

Kolner phonetic algorithm is similar to soundex but was designed for German words.

Metaphone family of algorithms are suitable for most of the English words and these algorithms are used for many English spell checkers and dictionaries. First metaphone algorithm was developed by Lawrence Phillips in 1990. Later variation of metaphone by him was double metaphone and incorporating other languages also. In 2009 he released the third version of metaphone which achieves accuracy of 99% of English words[6]. NYSIIS meaning that New York state Identification and Intelligence System which is known as NYSIIS phonetic algorithm was developed in 1970 and has achieved increased accuracy over soundex algorithm.

The match rating Approach (MRA) is a phonetic algorithm which was developed by Western Airlines in 1977 for indexing and comparing homophonous names. MRA uses distance calculation between two words. It can compare maximum of 12 character words.

The Caverphone phonetic algorithm was developed by David Hood at the University of Otago in New Zealand in 2002 and revised in 2004. It was created for data matching between late 19th century and early 20th century electoral rolls to commonly recognize the names and surnames.

All these algorithms have their own advantages and characteristics. Any algorithm or combination of these algorithms can be used for better accuracy for determining phonetic equality. Use of more than one algorithm proves better performance for identifying homophones. By using these algorithms it is possible to search the family of homophones. For this, it is necessary to bind homophones together in form of semantic map which can also be called as network of homophone words. In this paper we have already such a network of word formed as semantic map. This semantic map is searched using a derive algorithm.

III. Semantic Map Introduction

Semantic map relates the words and represented in form a network of words[13]. This graphical representation of the words is more suitable to understand the concept and relationship among the words. The semantic map can be defined as network of words or word web for some relation that binds them together. Semantic map can be useful for increasing the vocabulary, clearing the concepts, simplify the solution that can be used to implement the artificial intelligence in computer applications. Here adopted semantic map consists of number of circles representing the word and connected with the other circled words through arcs which represents the homophone relationship. Other possible semantic maps may be constructed for other kind of relationships like synonyms, antonyms or phrases which have similar conceptual and contextual meaning[12][13]. This research paper uses such a semantic map of homophones and searching is made for the given word from a large database of English words. General form of semantic map is depicted in following figure[10].

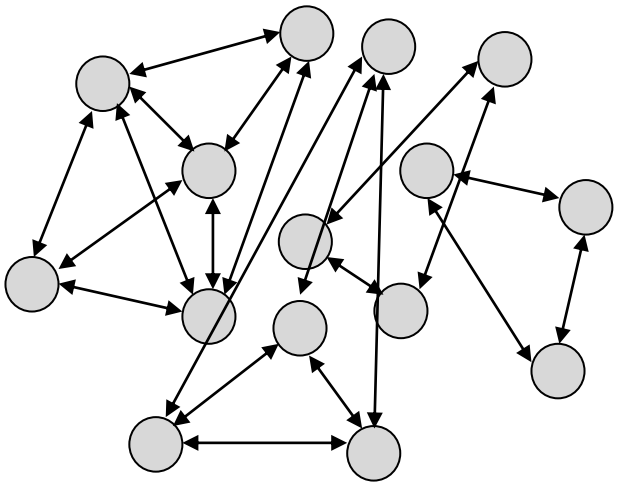

Figure 1 : General Semantic map

The arcs are bidirectional joining circled words representing the relationship and binding them together. The circles or nodes are united through the arcs in map using the indices. There may be isolated nodes which have no homophone. Many such semantic maps of words can be possible based on the relationship among the word. But the basic mechanism remains the same to bind all the related words together. Searching operation can efficiently be applicable and movement from one word to another related word can directly be possible using the indices of each word in the semantic map.

IV. Importance and Necessity of Related Homophone Semantic Map Network

We can use more than one phonetic algorithm for searching set of homophones for a given English word from a large English word database for improved performance. The given word must be phonetically compared using these algorithms with each and every word in the large word database. The process of such direct searching homophones is time consuming and requires more computing. This kind of process can be implemented using linear search approach. Although the word may be arranged in alphabetical order, possibly binary search can also be not applicable as all the words in the database must be compared with the given word. So it is not possible to take advantage of the computing power. For an example if database has English word list of 72000 words approximately for simplifying calculation then the given word must be compared with all the 72000 words using selected phonetic algorithms. If search frequency is more than this direct searching technique is an inefficient searching involving more computation. But the same kind of searching can be applied using some other solution to improve the performance. One solution is to use semantic map. We assume that a semantic map of homophone is available to us and then we require a procedure to search from this semantic map. Although the process of constructing a semantic map is time consuming process which is acceptable because it is only one time processing and after construction searching can be performed efficiently as any number of times. The prepared network of homophone words semantic map then can be treated as

knowledge base having the knowledge of related words. Database file organization and structure of the available semantic map network of homophone words is depicted in following figure 2.

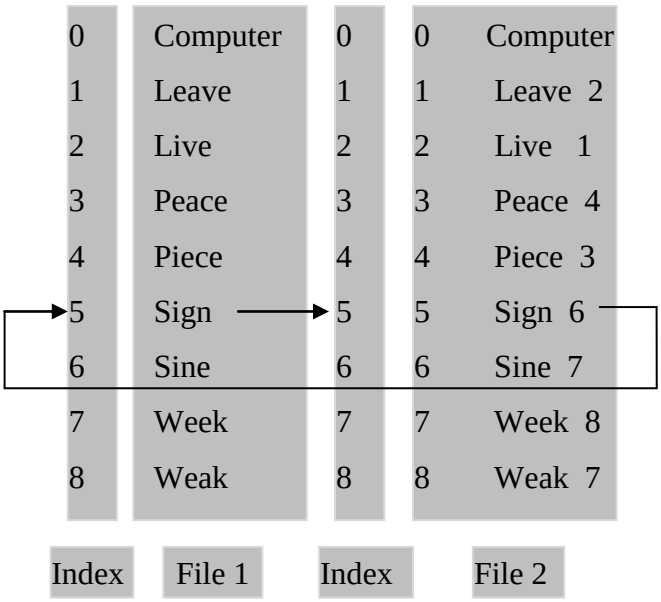

Figure 2 : Original and Semantic map File Structure

For sake of simplicity a set of homophones are listed in two files. First file just consists of words each in a separate line. Second file representing the complex semantic map which ties the homophones using indices. Whenever search for homophones is made for a given word, it is searched from the first file by applying the linear search algorithm and phonetic matching algorithm. Once the match is found its index is recorded and the word with the same index is located in second file. Now in second file, record contains a list of indices separated by comma at the end of the word. These indices are parsed and one by one indices are fetched and directly locating the word at that index from the first file which is the homophone for a given word. Further, in this example only two homophones are displayed, but in actual implementation may have more related words. Also it is possible to apply search directly using the second file only but in this case parsing process may be increased. Here the effort is being made for achieving the desired performance with the cost of duplicate content file, which offers significant performance. In many computing applications, time and space complexities are in inverse proportional to each other. If we try to save time, space may increase, and if we try to save space time may increase. Here, the space increase cost can be acceptable comparing to the efficient performance is achieved in searching process.

V. Search Procedure Model From a Semantic Map Network Structure

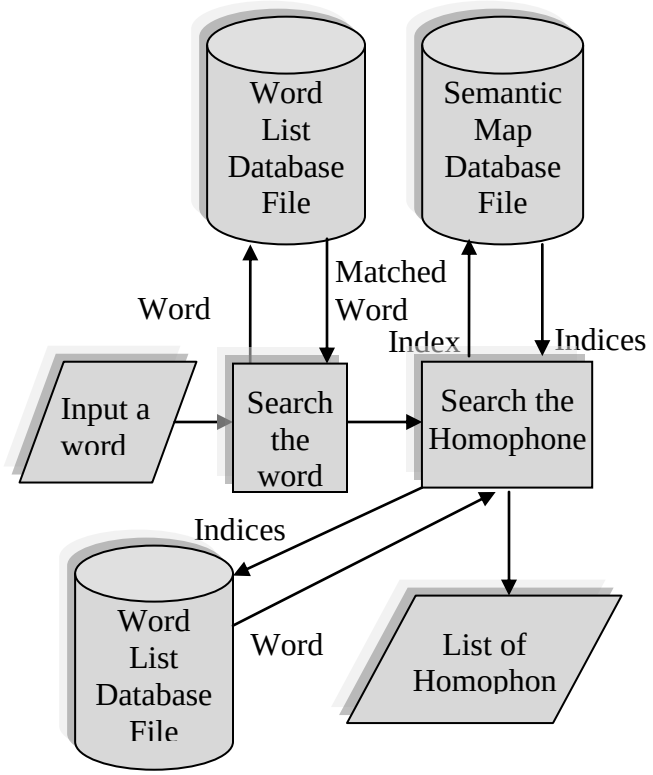

Figure 3 : Searching homophone model from Semantic map network of homophones

The model is designed and developed for searching homophones from a given semantic map network. Word list of nearly 72000 words is used that forms a semantic map network. First an input word is taken to search a set of homophone words. This word is compared and searched from the first database file having listing of words. Once a phonetic match is found may be the same word or similar pronunciation word, its index entry is noted. The record with that index is directly located in second file. The record at that index in semantic file contains a set of indices. Entire record is parsed and homophone word indices are used to find out the word directly from the database. The list of indices are used one by one locate the word in the first file again. Using indexing scheme, search process becomes faster. All the words at found indices are the homophone words and returned as searching result. The prime condition here is that, the semantic map must have been constructed prior to the searching is applied. It is also possible to search using a single semantic file but the overhead of parsing becomes cumbersome. So the process is simplified and searching becomes more efficient.

From the example of the file structure in figure 1, a search of homophones for a word “sign” is performed. First the input word “sign” or its any homophone is phonetically compared with all the words in first database file. Here the match is found at word index at 5. The word at index 5 is located in second file representing semantic map of homophones. The record at this location consists of comma separated indices. In this example at the end, it contains index 6 meaning that word at index 6 is the homophone of this word. Now all these indices are parsed and again located words with these indices in first file. In this example word at location 6 is “sine” is recorded as homophone. All such words whose indices are at the end of the found word are returned back as a result.

This searching is efficient because only first match is required to obtain the index of the word itself or its first homophone and not necessary to compare with every homophone word. This reduces the redundant processing of comparisons. For example if for a given word has say 10 homophones as its family homophones then it requires to scan first file only once until a first match is found. Once the match is found, its homophone indices are available in second semantic map file and directly located the indexed words. Without this approach it is necessary to compare each word in the first file using suitable set of phonetic algorithms. Number of comparison is reduced and hence the performance is improved.

The entire process encompasses of two procedures. One to create semantic homophone map and second is to search using this constructed semantic map. This paper is about the searching procedure and assumed that semantic map with structure given in figure 2 is already available. The procedures can be implemented using any programming language supporting database handling or file handling capabilities. .

Algorithm For Searching Homophones From Semantic Map Homophone Word Network

Algorithm for searching homophones for a given word is listed as follows.

Step 1 : Start for preparing, connecting and fetching the first original word list from the database file and second an index file which contains the index followed by the word followed by the homophone indices. Count the total number of words say N in the original word list file to process all the words. Receive a word WORD for which homophones are to be searched.
Step 2 : Repeat Step 3 for i = 0 to N - 1.
Step 3: Start comparing the given WORD with all the words word(i) in a first file. Once a match found record its index i as INDEX.
Step 4: Locate the record with index INDEX in second semantic map file.
Step 5 : Parse the located record and store the comma separated indices in an index array ind [ ] and store the total number of indices in TOTAL.
Step 6 : Repeat for i = 0 to TOTAL – 1
Locate record ind[i] in first file and record the word at that index in a string array homophone[ ].
Step 7 : The content of the string from homophone[i] to the element homophone[TOTAL-1] represents homophone family words for a given word WORD.
Step 8 : Finished

Algorithm processes by initiating connection with both the files, first original word database file for reading all the words and second semantic map homophone word file containing indices. Searching starts from first file by searching homophone for a given word. Once the match is found, it index is recorded in variable INDEX. The record with the index INDEX is located in second semantic file. Then the record is parsed and only the indices at the end are stored in an array representing the indices of homophones for a given word. Parsing mechanism requires separating each index which is comma separated. Once indices are parsed and stored, the records at those indices are located one by one in first file and recorded in a string array. The strings in this array represent the homophone family of the given word and the algorithm is terminated by closing the files and connections.

VI. Conclusion

The algorithm can be implemented using any programming language supporting database and file handling capabilities. Prerequisite of this algorithm is the semantic map of homophone must be available, as it works on the file structure designed as given in figure 2. Exact amount of time is not calculated but the time required using this search technique is less compared to direct searching entire database file of English words. Preparing semantic map requires hours of time depending on the amount of words in database but once it is created searching a family of homophones becomes more efficient using this algorithm.

Further, this technique can be employed to other computer applications where direct searching using linear search or binary search is not applicable or not efficient. One of the search applications which employ such technique is search engine which filters the database and returns only the related search result. In general using such technique there may be possibly many scope exist where text searching is at the center.

References

[1] Analysis and Comparative Study on Phonetic Matching Techniques Rima Shah, Dheeraj Kumar Singh International Journal of Computer Applications (0975 – 8887) Volume 87 – No.9, February 2014

[2] Name and Address matching strategy – White Paper Series Truth Technologies December 2010

[3] Vimal P. Parmar, Dr. CK Kumbharana "Study Existing Various Phonetic Algorithms and Designing and Development of a working model for the New Developed Algorithm and Comparison by implementing it with Existing Algorithm(s)" International Journal of Computer Applications (IJCA) ISSN: 0975 – 8887 Volume 98 / Number 19 (ISBN: 973-93-80883-19-1) DOI : 10.5120/17295-7795

[4] Phonetic algorithm meaning and description
http://en.wikipedia.org/wiki/Phonetic_algorithm

[5] Soundex algorithm description and working mechanism
http://en.wikipedia.org/wiki/Soundex

[6] Metaphone algorithm description and working mechanism,
http://en.wikipedia.org/wiki/Metaphone

[7] Vimal P. Parmar, Apurva K. Pandya, Dr. CK Kumbharana "Determining the Character Replacement Rules and Implementing Them for Phonetic Identification of Given Words to Identify Similar Pronunciation Words" Futuristic Trends on Computation Analysis and Knowledge Management (ABLAZE) 2015 International Conference at Greater Noida, India Pages : 272-277 Print ISBN : 978-1-4799-8432-9 DOI : 10.1109/ABLAZE.2015.7155010 Publisher : IEEE

[8] Homophone and related linguistic concepts
https://en.wikipedia.org/wiki/Homophone

[9] The Semantic Map of the Spatial Domain and Related Functions- Wei Wang University of California, Los Angeles- Language and Linguistics 16(3) 465–500 DOI: 10.1177/1606822X15569169

[10] The representation of homophones: Evidence from remediation- Britta Biedermann, Gerhard Blanken, Lyndsey Nickels- APHASIOLOGY, 2002, 16 (10/11), 1115–1136

[11] The homophone effect during visual word recognition in children: an fMRI study- Sharlene D. Newman, Psychological Research DOI 10.1007/s00426-011-0347-2

[12] 3rd grade context clues - http://kc3rd.pbworks.com

[13] Semantic relationship - www.iva.dk/bh/lifeboat_ko/CONCEPTS/semantic_relations.htm

AUTHORS

First Author – Vimal P.Parmar, Research Scholar, Dept. of Comp. Sci., Saurashtra University, Rajkot. Gujarat, INDIA parmarvimal1976@yahoo.co.in
Second Author – Dr. CK Kumbharana, Head, Guide Department of Computer Science, Saurashtra University Rajkot. Gujarat, INDIA, ckkumbharana@yahoo.com