Performance Evaluation of Phonetic Matching Algorithms on English Words and Street Names
Comparison and Correlation

Keerthi Koneru, Venkata Sai Venkatesh Pulla and Cihan Varol
Department of Computer Science, Sam Houston State University, 1803 Ave I, AB1 214, Huntsville, TX, U.S.A.

Keywords: Caverphone, Dmetaphone, Information Retrieval, Misspelled Words, Metaphone, NYSIIS, Phonetic Matching, Soundex.

Abstract: Researchers confront major problems while searching for various kinds of data in a large imprecise database, as they are not spelled correctly or in the way they were expected to be spelled. As a result, they cannot find the word they are looking for. Over the years of struggle, relying on pronunciation of words was considered to be one of the practices to solve the problem effectively. The technique used to acquire words based on sounds is known as "Phonetic Matching". Soundex is the first algorithm proposed and other algorithms like Metaphone, Caverphone, DMetaphone, Phonex etc., have been also used for information retrieval in different environments. This paper deals with the analysis and evaluation of different phonetic matching algorithms on several datasets comprising of street names of North Carolina and English dictionary words. The analysis clearly states that there is no clear best technique in general since Metaphone has the best performance for English dictionary words, while NYSIIS has better performance for datasets having street names. Though Soundex has high accuracy in correcting the misspelled words compared to other algorithms, it has lower precision due to more noise in the considered arena. The experimental results paved way for introducing some suggestions that would aid to make databases more concrete and achieve higher data quality.

1 INTRODUCTION

Information deterioration is an intensive problem for every organization in the present era. With the increase in the amount of information saved day by day, there is a desperate need for locating the mistyped data. Organizations are facing great challenge to maintain the quality of data in information systems with various sources of data damage. Whenever the data is assimilated from multiple sources, it is a complicate process to recognize the duplicate records due to the existence of misspelled data for the same record. As a result, the information of organization always ends up at risk.

Databases play a crucial role in almost all of the establishments. Matching records in database is a persistent and well-known problem for years. One of the techniques to improve the data quality, which uses variations of sound to detect the misspelled data, is phonetic matching. The evolution of phonetic matching has come into frame when there is a hardship in the information retrieval. The technique of obtaining words using sounds was used in the US census since the late 1890's, but a concrete solution to this was first proposed and patented by Robert C. Russell in 1912 as Soundex algorithm. Later, many algorithms were developed based on the different specifications and language constraints. Phonetic matching plays a key role in information retrieval in multilingual environments, where diversities in pronunciation or writing styles with same meaning may be present. In such cases, the phonetic matching technique is also used for different languages other than English.

Some of the other prominently used algorithms are Metaphone, DMetaphone, Caverphone, and New York State Identification and Intelligence System (NYSIIS) Phonetic code. This paper provides an overview of five phonetic matching algorithms, specifically Soundex, Metaphone, DMetaphone, Caverphone, and NYSIIS and evaluates their effectiveness on variety of types of strings. The efficiency of these algorithms is calculated by obtaining information retrieval metrics – Precision and Recall.

The rest of this paper is structured as follows.

Koneru, K., Pulla, V. and Varol, C.
Performance Evaluation of Phonetic Matching Algorithms on English Words and Street Names - Comparison and Correlation.
DOI: 10.5220/0005926300570064
In *Proceedings of the 5th International Conference on Data Management Technologies and Applications (DATA 2016)*, pages 57-64
ISBN: 978-989-758-193-9

Related works and description of phonetic matching techniques used in this study are described in Section 2. The main contribution of this paper is presented in section 3 which describes the preparation of the datasets, the metrics used to evaluate the experiments, and the analysis and comparisons. Finally, this paper is concluded and future work is pointed out in section 4.

2 BACKGROUND

2.1 Related Work

Phonetic comparison meticulously obtains the quantitative analysis of pronunciations between speech forms and spellings of words. The different sources of variations can be illustrated as:

(1) Spelling variations

(2) Phonetic variations

(3) Double names or double first names

(4) Change of name (Shah and Singh, 2014).

Due to different criteria mentioned above, rather than looking for exact matching, considering approximate matching would be worthwhile.

The initial algorithm developed for phonetic matching is Soundex, which produces a four digit code retaining its first letter. It is used as a standard feature in applications like mySQL, oracle, etc.

Because of few disadvantages like dependency on the first letter, failure of detection of silent consonants, Soundex can only be used in applications where high false positives and false negatives can be tolerated (Shah and Singh, 2014). Though Shah and Singh, described two of the phonetic matching algorithms being discussed in this paper, they were only able to provide appropriate area of applications for these techniques without any statistical justification.

An improvement of Soundex is implemented by Beider and Morse to reduce the number of false positives and false negatives, known as Beider-Morse Phonetic Matching (BMPM). Beider and Morse, (2010) has also mentioned that the algorithm is extended to languages other than English, with the application of some generic rules to obtain the phonetic codes. Varol and Talburt (2011) discussed BMPM as a hybrid technique with a 6-letter encoded code in which the percentage of irrelevant matches can be abated by 70%.

Phonex is a technique in which words are pre-processed before encoding. In order to overcome defects of Phonex, Phonix has been introduced with a number of transformations in the beginning, ending and in the middle of the word (Varol and Talburt, 2011).

NYSIIS algorithm was developed in 1970 as a part of New York State Identification and Intelligence System project headed by Robert L. Taft (Hood, 2004). The algorithm produces a canonical code similar to Soundex, but generates only alphabetic code (Balabantaray et al., 2012). Balabantaray et al., mentioned that unlike Soundex, NYSIIS retains information regarding vowels (Balabantaray et al., 2012).

Though sounds are taken into consideration, all the above mentioned algorithms consider the phonetics of each letter. A new technique which considers diphthongs (combination of two or more letters) of words was first developed by Lawrence Philips in 1990 known as Metaphone. Bhattacharjee et al., (2013) has stated that this technique is mainly used for data cleaning in the text files to remove erroneous data.

Pande and Dhami (2011) detailed that the Metaphone has its extended usage in stemming, which improves performance in Information Retrieval (IR). In stemming, natural language processing tools like Levenshtein Edit Distance (LED) algorithm conflated with phonetic matching algorithms like Metaphone are used for greater accuracy (Pande and Dhami, 2011). David Hood cited that though the algorithm is sensitive to combination of letters like ‘TH’, it is not subtle enough with the vowels especially at the postvocalic L and R (Hood, 2004). Zobel and Dart demonstrated different phonetic matching and string matching algorithms on personal names. However, the paper was published two decades ago which does not include the newer techniques and substantial changes in some of the existing algorithms (Zobel and Dart, 1996).

Double Metaphone, popularly known as DMetaphone, is an enhancement to Metaphone algorithm by Lawrence Phillips in 2000. Unlike Soundex, which encodes letter by letter, DMetaphone encodes groups of letters called diphthongs according to a set of rules (Varol and Talburt, 2011). Carstensen, mentioned that the algorithm is more effective while matching proper names and short sentences in databases (Carstensen, 2005).

In pace, the specified algorithms are not suitable for a particular database, named Caversham, which is mainly used for data source linkage. The algorithm, known as Caverphone, which is analogous to Metaphone with some rules subsequently applied, is enforced by David Hood in 2002 to encode the data of Caversham database (Hood, 2004). The algorithm is later improvised in 2004 to Caverphone 2.0, to increase its accuracy and efficiency by applying more set of rules. David Hood, (Hood, 2004) also stated that the algorithm is efficient by giving precise matches when compared to Soundex and Metaphone algorithms for linking data sources.

One of the major applications of phonetic matching algorithms is its appliance to different languages. Advanced techniques with collaboration of two or three algorithms have paved the way for obtaining codes to the phonetics of a word in different languages (Varol and Talburt, 2011).

There are other works in the literature (Chan et al., 2015; Christen, 2006) describes about various phonetic matching algorithms. However, the area of interest on these studies are either related to string matching algorithms or the patterns seen in misspellings.

2.2 Phonetic Matching Techniques

The experiments in this work are based on five popular phonetic matching techniques, namely, Soundex, NYSIIS, Metaphone, DMetaphone, and Caverphone. The functionality of these five phonetic matching algorithms is illustrated in this section. All the algorithms other than Soundex, have larger rule set which cannot be explicitly fit to this paper. Hence, an overview of the functionality of algorithms is described below:

2.2.1 Soundex Algorithm

The steps for generating phonetic code using the Soundex algorithm are as below:

1. Retain the first letter of the word.
2. For the remaining letters, numbers are to be assigned as reflect in the table below:

Table 1: Soundex Translation Table.

a, e, h, i, o, u, w, y	b, f, p, v	c, g, j, k, q, s, x, z	d, t	l	m, n	r
0	1	2	3	4	5	6

3. From the string obtained by the above manipulations, remove all pairs of same digits that occur beside each other.
4. All zeroes, obtained from the above step, are removed from the string.
5. The first four characters are considered to be Soundex code, and are right padded with zeroes if the string is deficit of four characters (Shah and Singh, 2014).

2.2.2 NYSIIS Phonetic Algorithm

NYSIIS is a computational algorithm mainly designed for American names. The algorithm initially transforms beginning and ending of words. It is then followed by set of rules to encode the remaining letters. The vowels are retained by converting all of them to ‘A’.

There are some special cases, where ‘AY’ is changed to ‘Y’; removing ‘A’ if they are existing at the end of encoded word, etc. The NYSIIS phonetic code, in general, truncated to 6 letter word, but it is optional based on the user defined requirement (Philips, 2000).

2.2.3 Metaphone Algorithm

Metaphone is an algorithm, which considers set of letters as an alternative to letter by letter encoding, to identify the phonetic variations and inconsistencies in words. The algorithm initially performs transformations using diphthongs such as changing MB to B if at the end of the word, SCH to K, CIA to X, and drops all the vowels in the encoded word. It shows that the phonetic sound of vowels combined with the consonants is considered instead of individual consonant or vowel sounds. Metaphone code length varies from 4-letter code to 12-letter code. The metaphone code used in this paper has a 12 letter encoded code as it improves the precision (Bhattacharjee et al., 2013).

2.2.4 DMetaphone (Double Metaphone) Algorithm

DMetaphone is a sound indexing system which groups letters not only by spellings but also by different pronunciations (Carstensen, 2005). Like Metaphone, the double metaphone also produces character code. But the major difference is the latter produces secondary key along with a primary encoded word to identify the most common native pronunciation.

Double metaphone has an extensive encoding for letter ‘C’, ‘G’ and ‘S’, as they have major differentiation in pronunciations on combining with other vowels and consonants (Philips, 2000). The algorithm retains only the first vowel sound to same character ‘A’ while all other vowel sounds are dropped. Later few other transformations are done on the remaining letters based on the letters present in the successor index and predecessor index.

2.2.5 Caverphone Algorithm

The algorithm for Caverphone 2.0 has several sequential transformations based on the characteristics of the word. Initially, all the letters are converted to lowercase and then it removes ‘e’ at the end of the word. The encoding uses the numbers ‘2’ and ‘3’ to encode few phonetic sounds. Later these numbers are again converted into alphabetic phonetic encoding. Unlike Soundex, the vowel sound is retained either as ‘A’, if at the beginning, or as 3, elsewhere.

The obtained word is followed by few other transformations such as ‘y’ is converted to ‘Y3’ upon appearance at the start of the word, s, t, p, k, f, m, n are converted into uppercase (if present as group of consecutive letters), ‘r’ present at the end of the word, is converted to 3. Later, ‘2’ are removed and ‘3’ is converted to ‘A’ at the end of the word. After all transformations the encoded word is truncated to ten letters and is appended by ‘1’, if necessary (Nikita, 2013).

Overall, Table 2 shows the phonetic codes for the words “Phonetic” and “Matching” generated by the discussed five algorithms.

Table 2: Sample Conversions.

Algorithm	**Phonetic**	**Matching**
Soundex	P532	M325
NYSIIS	FANATA	MATCAN
Metaphone	FNTK	MCHXNK
DMetaphone	FNTK	MXNK
Caverphone	FNTK111111	MKNK111111

3 EXPERIMENTAL RESULTS

In our experiment, we analyzed the effectiveness of five prevalent phonetic matching algorithms on two types of datasets, namely, street names of North Carolina (NC Master Address Dataset, 2014) and English dictionary words (Lawler, 1999). For street names four different sizes of datasets are considered from 200 to 800. At a particular size, names are analyzed by generating synthetic data having four types of errors such as single error in each string (swapping of letters, misspelled letter, and absence of letter), addition of a random letter in each string, strings having mixed errors (strings having more than two errors that includes swapping of letters, absence of letter, having additional letter, etc.), and two errors in each string. Similarly, for the English dictionary, four different data sizes are considered ranged from 200 to 800. Each of the synthetic dataset contains 100% errors.

3.1 Dataset Preparation

Various experiments have been conducted on finding phonetic matches for misspelled words of personal names. Moreover, various string matching algorithms were applied on the English dictionary words but there is only little exploration in finding the phonetic matches of street names or English words in the literature. Hence, in this paper we mainly concentrated on obtaining the phonetic matches for misspelled words of street names and English dictionary words.

The datasets for the experiment are prepared as follows. For the street names, the datasets are based on the North Carolina (NC) Master Address Dataset (NC Master Address Dataset Project, 2014). Initially, all street names are extracted and a list is created having correct, non-duplicate names. This list is used as a reference list for obtaining the suggestions for misspelled street names. From the clean list, different datasets are generated which contains correct data and corresponding manipulated datasets having misspelled data.

According to Kukich (1992), nearly 80% of problems of misspelled words can be addressed either by addition of a single letter, or replacement of a single letter or swapping of letters. Therefore, manipulated datasets are obtained by executing addition, deletion, swapping, and replacement of letters. A total of forty eight datasets are generated with four datasets at each type of error and data size, ranging from 200 to 800.

By the same token, the English dictionary word list is extracted without any duplicate or erroneous values. Like street names, forty eight synthetic datasets and corresponding reference datasets are generated with data size ranging from 200 to 800.

3.2 Evaluation Metrics

The performance of phonetic matching algorithms used for information retrieval is evaluated by calculating precision, recall, and F - Measure.

3.2.1 Precision

Precision gives the total number of true positives obtained over the total number of suggestions generated for a misspelled word.

$$P = \frac{\sum p}{\sum Number\ of\ suggested\ words\ for\ each\ corrected\ word}$$

, where $p = \begin{cases} 1, if\ the\ word\ is\ corrected \\ 0, if\ the\ word\ is\ not\ corrected \end{cases}$

P = cumulative precision for an algorithm.

3.2.2 Recall

Recall provides the total number of relevant words over the total number of suggestions (Kelkar and Manwade, 2012).

$$R = \frac{Number\ of\ corrected\ words}{Total\ number\ of\ misspelled\ words}$$

where R = recall or efficiency of an algorithm.

The efficiency of an algorithm is obtained by calculating these metrics for different input records.

3.2.3 F-Measure

The F – Measure is calculated based on precision and recall and is defined as the harmonic mean of precision and recall. It is given by

$$F = \frac{2 \times P \times R}{P + R}$$

For comparison, maximum F - Measure for different datasets are considered, which vary in size and features.

3.3 Results and Evaluation

The experiment illustrates the performance of different algorithms on datasets of particular size having various types of errors. From the results, it can be stated that the variations in performance is highly dependent on the type of error. And also, for this experiment the size of the database of English dictionary is comparatively larger than the database of North Carolina addresses. As a result, it is observed that there is a significant difference in the results of calculating precision and recall of misspelled words from English dataset and North Carolina Address dataset.

3.3.1 Analysis on Synthetic Data from English Dictionary

Overall, the experimental results show that Metaphone excels its performance compared to other techniques for all types of errors, which is followed by Caverphone and NYSIIS. The test results are obtained from four different datasets for various sizes of data ranging from 200 to 800. But, as per the observation, the results are not highly distinguishable for different size of datasets. Hence, in the rest of the paper the average values of the test results are coming from the datasets containing 800 records.

Figure 1 represents the F-measure for different techniques implemented on the datasets having synthesized data of English dictionary words of data size 800. From the figure, it can be clearly stated that Metaphone provides best performance compared to other techniques on all datasets having different types of errors such as single error in a word, double errors in a word, mixed errors, and when there is an accidental additional character in the word. Caverphone gives second best performance while the lowest performance is given by Soundex. DMetaphone has similar performance to Soundex whereas NYSIIS has an average performance. The figure also illustrates that Metaphone has its highest performance on the datasets having words with single error. Also, we observed the same behavior in other size of datasets as well.

From the obtained values, it can be inferred that all algorithms produce average competence when the errors are mixed irrespective of the size of the dataset for English words.

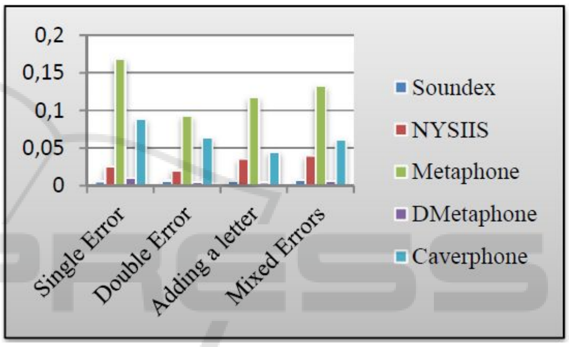

Figure.1: F-measure for different techniques on English dictionary datasets of size 800.

Figure 2 interprets the recall value of different algorithms for various types of errors on English dictionary datasets of size 800. In spite of having the lowest F-measure, the recall value of Soundex is exceptional for any of the dataset. The analysis also reflects that the accuracy rate is high for mixed errors than other type of errors, whereas it is very low for words having double errors.

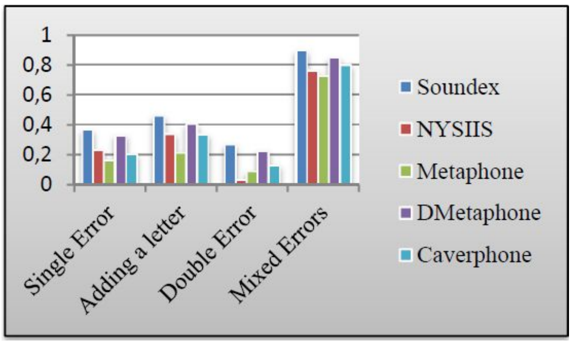

Figure 2: Recall of different algorithms for various types of errors on English dictionary dataset of size 800.

3.3.2 Analysis on Synthetic Data from NC Master Address Dataset

Figure 3 details the F-measure obtained by applying phonetic techniques on the datasets of NC Address. From the figure, it can be seen that NYSIIS provides better performance compared to other techniques on different datasets of various types of errors. Metaphone has the next best performance, while Soundex shows worst performance for all type of errors and size of datasets. The analysis also shows that all the algorithms have low performances for double errors. Of all types of errors, NYSIIS and Metaphone obtained better performance for datasets having words with single errors and additional character in them.

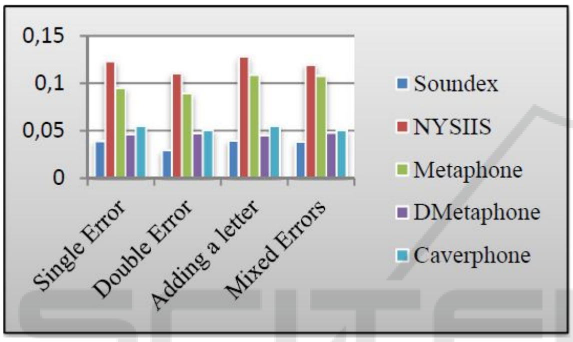

Figure 3: F-Measure for different techniques on NC Address datasets of size 800.

From the experiment, it is also observed Soundex has the highest recall for a given dataset. It clearly shows that the correction rate is high for Soundex compared to other algorithms. Figure 4 reflects the accuracy of different algorithms on NC Address datasets of size 800. The poor F-measure of Soundex is due to the retrieval of high false positives from the database.

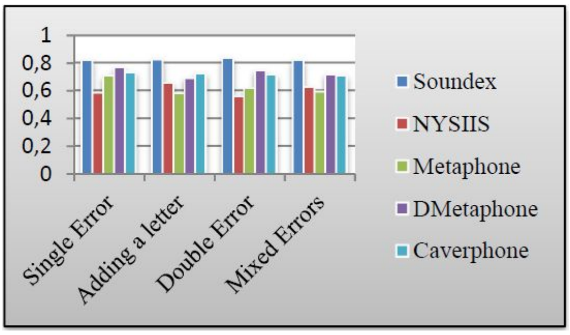

Figure 4: Recall of different algorithms for various types of errors on NC Address datasets of size 800.

3.3.3 Evaluation of Processing Time

Figure 5 and Figure 6 represents the processing time for each algorithm based on the type of errors. Metaphone consumes least processing time when synthetic dataset of English dictionary words is used as shown in Figure 5, whereas NYSIIS has the least processing time compared to other algorithms when synthetic dataset of street names is used as portrayed in Figure 6. The maximum processing time is consumed by Soundex, in both the scenarios, independent of type of errors in the dataset, and its size. Caverphone has an average processing time, better than Soundex and DMetaphone. The experimental results also show that the processing time is highly dependent on the size of dataset. The processing time of Metaphone is 48 seconds for a data size of 200 for English dictionary dataset, whereas the value is as high as 240 seconds (nearly) for dataset of size 800 on a single processor Windows 7 OS desktop computer with 4GB of memory. Similarly, NYSIIS and Metaphone have processing time of 10 seconds for a dataset of size 200 which is way less than 40 seconds (nearly) for NC Address dataset of data size of 800.

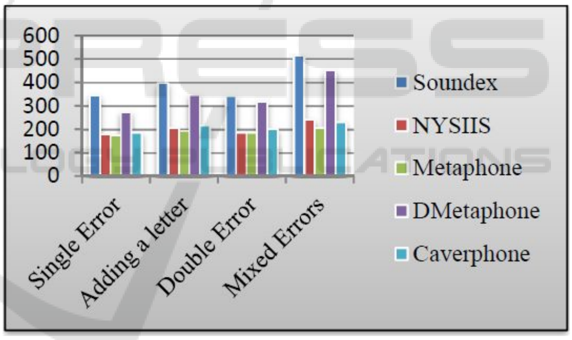

Figure 5: Processing time (in seconds) for different techniques on English dataset of size 800.

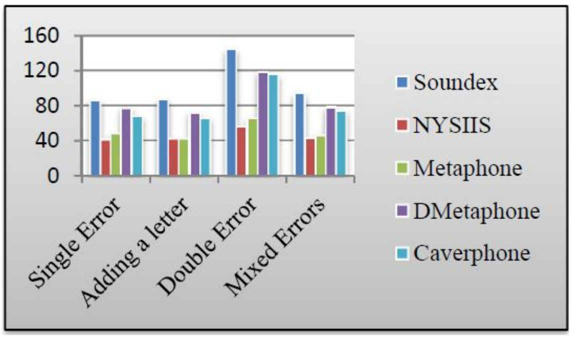

Figure 6: Processing time (in seconds) for different techniques on NC Address dataset of size 800.

We also found that the precision is high for Metaphone and NYSIIS. In detail, for the NC Address dataset of size 800, the cumulative precision of Metaphone and NYSIIS is nearly 0.06 respectively, whereas for Soundex and DMetaphone the value is nearly 0.02. However, the value of cumulative precision is dependent on type of errors for the English dictionary dataset. But, it is observed that Metaphone has highest precision varying from 0.2 to 0.07, while Soundex and DMetaphone has the lowest value varying from 0.008 to 0.002. This clearly shows that Soundex and DMetaphone have high noise of all the five algorithms, which decrease their performance simultaneously increasing their processing time.

Overall, the experiments can be concluded that Metaphone is a better algorithm comparatively for English dictionary words while NYSIIS is better algorithm for street names.

4 CONCLUSION AND FUTURE WORK

This paper elicits the efficient algorithm by calculating the precision and recall on different inputs for the street names from NC Address dataset and English dictionary in the database. In spite of errors being typographical, the phonetic matching algorithms are still able to address them in acceptable level.

The algorithms are fruitful in terms of accuracy, but they are not very productive as the precision is very low due to number of false positives. Metaphone and NYSIIS are more efficient of the five analyzed algorithms, comparatively, for different inputs having different types of errors. Caverphone has relatively more efficiency than DMetaphone and Soundex.

As per the observations, Soundex has high recall compared to other algorithms but because of its low precision the algorithm is not very efficient. Due to its high accuracy, the algorithm is still used in various applications having high tolerance to false negatives. From the above experimental results, it is evidential that there is no unique algorithm which is effective for all types of databases.

Though the experiment gives near suggestions from the five algorithms, it would not detect all the close matches, as the matched word from the database is an extraction with exact replica of complete phonetic code generated. A more transparent analysis can be performed to obtain the efficient algorithm by considering a threshold in obtaining the near matches. The threshold can be fixed based on employment of string matching algorithms like Levenshtein Edit Distance (LED) algorithm or Boyer-Moore string matching algorithm on the phonetic codes to improve the accuracy and F-measure. Moreover, efficiency on Street names can be improved if other languages' phonetic structures are introduced to the system.

REFERENCES

Balabantaray, RC, Sahoo, B, Lenka, SK, Sahoo, DK & Swain, M May 2012. An Automatic Approximate Matching Technique Based on Phonetic Encoding for Odia Query. *IJCSI International Journal of Computer Science Issues*, Vol. 9, Issue 3, No 3.

Beider, A & Morse, SP March, 2010. Phonetic Matching: A Better Soundex. [Online] Available from: http://stevemorse.org/phonetics/bmpm2.htm

Bhattacharjee, AK, Mallick, A, Dey, A & Bandypoadhay, S September 2013. Enhanced Technique for Data cleaning in text files. *International Journal of Computer Science Issues*, Vol. 10, Issue 5, No 2.

Carstensen, A September 2005. An Introduction to Double Metaphone and the Principles behind Soundex. [Online] Available from: http://www.b-eye-network.com/view/1596

Chan, K, Vasardani, M & Winter, S August 2015. Getting lost in Cities: Spatial Patterns of Phonetically Confusing Street Names. *Transactions in GIS*, Vol. 19, Issue 4, August 2015.

Christen, P December 2006. A Comparison of Personal Name Matching: Techniques and Practical Issues. *Sixth IEEE International Conference on Data Mining - Workshops (ICDMW'06)*, pp. 290-294, December 2006.

Hood, D December, 2004. Caversham Project Occasional Technical Paper.

Kelkar, BA & Manwade, KB June 2012. Identifying Nearly Duplicate Records in Relational Database. *IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS)*, Vol. 2, No.3

Kukich, K December 1992. Techniques for automatically correcting words in text. *ACM Computing Surveys*, Vol. 24, No.4

Lawler, J March 1999, An English Words List, [Online] Available from: http://www-personal.umich.edu/

Nikita, March 2013. Phonetic Algorithms. [Online] Available from: http:// ntz-develop.blogspot.com/2011/03/phonetic-algorithms.html

Pande, BP & Dhami, HS August 2011. Application of Natural Language Processing Tools in Stemming. *International Journal of Computer Applications* (0975 – 8887) Volume 27– No.6

Philips, L June 2000. The Double Metaphone Search Algorithm. [Online] Available from: http://www.drdobbs.com/the-double-metaphone-search-algorithm

Shah, R, & Singh, DK February, 2014. Analysis and Comparative Study on Phonetic Matching Techniques.

International Journal of Computer Applications, Volume 87 – No.9.

Varol, C & Talburt, JR 2011. Pattern and Phonetic Based Street Name Misspelling Correction. *Eighth International Conference on Information Technology: New Generations*.

Zobel, J & Dart, P 1996. Phonetic String Matching: Lessons from Information Retrieval. *Nineteenth Annual International ACM SIGIR conference on Research and development in Information Retrieval*.

2014 NC Master Address Dataset Project, *Center for Geographic Information and Analysis*, [Online] Available from: http://www.cgia.state.nc.us/Services/NCMasterAddress.aspx