SIGNIFICANCE OF CHARACTER 'H' IN SOUNDEX PATTERNS ON INDIAN NAMES

G. Christopher Jaisunder[a], Israr Ahmad[b], Dhavamani Christober[c]
a Research Scholar, Mewar University, Gangrar, Chittorgarh, Rajasthan, India, Email: christopher@nic.in
b Department Of Computer Science, Jamia Millia Islamia, Delhi, India, Email: israr_ahmad@rediffmail.com
c Principal & Secretary, The American College, Madurai, India, Email: christober.md@gmail.com

Abstract:
In this digitization age, particularly under the umbrella of Digital India scheme, each and every establishment in India is in the process of digital transformation one way or the other. The establishments vary from small private business to larger enterprise and in the government sector in various levels of central/ state/ district administration. In government, the enrollment, verification, identification, usage and maintenance of demographic data particularly on the personal names in the critical government records play a very important role. A very small spelling mistake in the name at the time of enrollment process leads to complications and the citizen runs from pillar to post for the rectification process. Though the rectification process in every system of enrollment is well defined, the process to get the spelling mistake corrected is very painful. The applicant needs to defend himself with 'n' number of documents available with him to prove his correct spelling in the required document. In recent years, the spelling mistakes happening at the enrollment stage are being resolved at the initial stage itself as the data entry is in the form of self service. One of the best practices is to get the consent of the applicant directly prior to processing his/ her application. The complication starts when the enrollment process happens in the absence of the applicant as the consent of the concerned applicant can't be taken across the counter. In government, most of the citizenship services are outsourced and the data correction process is being carried out as a separate entity as part of the application acceptance. During the application acceptance process, every agency does the data verification in their own style as per the guidelines from the concerned government. Irrespective of the methodology being followed in name checking, the earlier occurrence of the same applicant within their own system is being carried out as part of the verification process. To implement this activity, government agencies need to follow some sort of soundex mechanism to find out the alternatives over the names in the data base and cross verify the same with other related parameters. In this scenario, it is understood that the spelling of the same name in a different part of India differs phonetically as some characters play a significant role. In this paper, we have tried to find and analyze the significance of the character 'H' in soundex patterns in Indian names.

KEYWORDS - *soundex pattern, false positive, false negative, verification, N-Factor, H-Factor*

I. INTRODUCTION

Information retrieval in the context of digital transformation is a broad concept where the requirement is to be defined very specifically depending upon the nature of the problem. The digital information retrieval needs a special attention to the digital representation of the basic data with specific methodology. Data de-duplication methods like soundex patterns help in fulfilling the requirements of search demographic data, particularly the search over the personal names. Some sort of optimization techniques in respect of storage are to be used in defining the soundex pattern that represents the theoretical base. Many more researchers are already working on this information retrieval under data mining concepts. Phonetic matching is used to evaluate similarity of the names without looking into to the actual spelling or comparing the name by character to character. Some more matching algorithms being used are Edit Distance algorithm, K-String and Q gram algorithm, Guth algorithm, Daitch Mokotoff algorithm, Metaphone coding algorithm and Soundex algorithm. The popular technique of information retrieval is the phonetic matching by using soundex patterns to compare the personal names based on the pronunciation. On this assumption, we have tried to do a small change in the standard soundex pattern by inclusion of the alphabet character 'H' and studying the significance of the same in the search process in the database.

This paper is organized as follows. Section I gives the introduction of the subject matter of this paper. Section II gives the standard soundex pattern methodology. Section III gives the understanding of the 'H' factor. Section IV gives the experiment and the results of the 'H' factor. Section V concludes the paper followed by references.

II. SOUNDEX PATTERN

Soundex pattern is the process of defining a set of similar sounding character into a defined value for the given character string. A soundex pattern is defined as a set of algorithm by means of the pronunciation of the string. They are necessarily complex algorithms with many rules and exceptions because of the English spelling and complicated pronunciation caused by historical changes in pronunciation and words borrowed from

many languages [4]. The spelling of the strings may be differently represented but phonetically same [8]. Phonetic matching is used to identify strings that may be of similar pronunciation, regardless of their actual spelling [5]. To understand the working of matching operation we will discuss the example of large database that consists of the names Stefan, Steph, Stephen, Steve, Steven, Stove, and Stuffin [1]. Suppose that we want to search for the name Stephen [1]. The matches that the search finds are called the positives, and those names that it rejects are called the negatives [1]. Those positives that are relevant are called true positives, and the others are false positives [1]. There is no single best technique available. Objective of selecting a suitable technique is to reduce the false positive and false negative cases [6].

As an example, let us assume that the matches found when searching for Stephen in the above database are Stefan, Stephen, Steven, and Stuffin [1]. The first three are probably relevant, and are names that we would have wanted to see. So these are the true positives [1]. Stuffin, however, is probably not relevant – it is a false positive [1]. The names that were rejected are Steph, Steve, and Stove [1]. Of those, Stove is probably not one that we would have wanted. So it is a true negative [1]. But Steph and Steve are ones that we would probably be interested in [1]. They are false negatives.

Searching names in large database have always been a problem. In the practical scenario, globally these individual names shall be heterogeneous in nature having wide range of varieties; but, locally homogeneous in nature in respect of the common names, spellings and pronouncing method [4]. The solution to the problem was given by Robert Russell in 1912 as he proposed the soundex algorithm [2]. The names might be misspelled in a large database or might not be spelled as one expected. In this case rather than looking for exact matching, searching for approximate matching will be significant [3]. One solution is to say that two names are approximate matches if they sound the same. Here, the question is, whether we could build the right algorithm with the sound principles that can be extended to reduce the error rate [7]. Soundex is the best-known phonetic matching scheme. Developed by Odell and Russell, and patented in 1918, soundex uses codes based on the sound of each letter to translate a string into a canonical form of at most four characters, preserving the first letter [5].

Soundex pattern is a system whereby values are assigned to names in such a manner that similar-sounding names get the same value. These values are known as soundex encodings. A search application based on soundex will not search for a name directly but rather will search for the soundex encoding. Based on the soundex encoding the similar sounding names would be retrieved from the large database.

Outline of Soundex Algorithm [5]

- Convert the string to English upper case characters only to fix the work domain.
- Retain the first letter of the string.
- Change all occurrences of the following letters to zero: A, E, H, I, O, U, W and Y.
- Assign numbers to the remaining letters (after the first) as follows: B, F, P, V = 1; C, G, J, K, Q, S, X, Z, = 2; D, T = 3; L = 4; M, N = 5; R = 6.
- Remove all pairs of digits which occur beside each other from the string that resulted after the previous step.
- Remove all the zeros from string that results from the previous step.
- Return the at most four characters, right-padding with zeroes if there are fewer than four.
- Taking an example we will see how soundex algorithm works. Example-"SMITH" will code to"S5030" which will then reduce to "S530"by computing the steps of soundex algorithm.
-

III. THE ‘H’ FACTOR

In India, the spelling of the personal names is being written differently from region to region with the country speaking many languages. The personal names like ‘Sunitha’ and ‘Sunita’ are being pronounced the same way, but with different spelling. For example the spelling ‘Sunitha’ is being used in the southern part of India whereas the spelling ‘Sunita’ is being used in the northern part of India. In the similar lines, the surnames like ‘Mishra’ and ‘Misra’ are being pronounced the same way, but with different spelling. For example the spelling ‘Mishra’ is being used in the northern part of India whereas the spelling of ‘Misra’ is being used in the southern part of India. Researchers use their own customized search algorithms in the search engine to search personal names in the large databases. We need to have our own customized soundex pattern algorithm that sounds similar names irrespective of their spelling. On this assumption, let us have a small change in the standard soundex pattern with the inclusion of the alphabet character ‘H’. The effect of this ‘H’ factor in the definition level shall be the same as the standard soundex pattern except the new introduction of the assignment for the alphabet character ‘H’.

The new algorithm looks like this:

- Convert the string to English upper case characters only to fix the work domain.
- Retain the first letter of the string.
- Change all occurrences of the following letters to zero: A, E, I, O, U, W and Y.

- Assign numbers to the remaining letters (after the first) as follows: B, F, P, V = 1; C, G, J, K, Q, S, X, Z, = 2; D, T = 3; L = 4; M, N = 5; R = 6; H=7.
- Remove all pairs of digits which occur beside each other from the string that resulted after the previous step.
- Remove all the zeros from string that results from the previous step.
- Return the at most four characters, right-padding with zeroes if there are fewer than four.
- Taking an example we will see how soundex algorithm works. Example-"SMITH" will code to "S5037" which will then reduce to "S537"by computing the steps of Soundex algorithm.

This redefinition is named as the introduction of 'H' factor to test and analyze the impact over the search process. We have created the soundex encodings for both the algorithms for a set of Indian personal names in the database. A set of search names were selected and the soundex encoded search key is codified in both the algorithms. A search engine is designed to find matches with a set of codified soundex encoded search keys over a database to analyze the quality and quantity of the matches due to the introduction of 'H' factor. The experiment result may include false positive and false negative cases in both the algorithms that have been analyzed.

IV. EXPERIMENTS WITH 'H' FACTOR

For the experimental purpose, we have collected short Indian personal names with only one component and tried to search similar names by using the soundex patterns. The maximum length of the name in the data bank is "CHANNABASAVESHWAR" with 17 characters in length. The maximum length of the name in the test name is "VENKATRAMAN" with 11 characters in length. It is believed that the short names shall give more accurate results as the soundex pattern for the experiment is limited to 4 characters only. It is ensured that the collected names represent all alphabets as starting letters and have the English alphabets only. The search over the name is not case sensitive. Prior to getting into the experiment let us understand the two key words used in explaining the experiment. One is the "soundex character" and the other one is the "non-soundex character" being considered while designing the soundex patterns. The English alphabets are grouped according to the expected phonetic sound and equated with a common representative character respectively. Such alphabets are termed as soundex characters and the other characters are termed as non-soundex characters.
Experimenting the significance of any factor involved in recording of observations both in the normal and new change factor of soundex pattern without disturbing the operating environment. Oracle pl/ sql programming language is used to design, develop and test the experiment. The normal soundex pattern is designed similar to the soundex native function defined in the oracle data base to have the base factor. To be specific to this context, the character 'H' is defined as the soundex character where the phonetic sound is equated to the other soundex characters like A, E, I, O, U, W and Y. These characters are further dropped and do not have the participation in the soundex pattern. This methodology is named as "N-Factor" method. The new soundex pattern is designed with a one character assignment change from the normal soundex pattern N-Factor. To be specific to this context, the character 'H' is defined as the non-soundex character and the phonetic sound is not equated to any other character. This methodology is named as "H-Factor" method. Both the N-Factor and H-Factor methodology is applicable to the names starting from the 2nd character only leaving the first character as it is.

A total of 2965 Indian names were collected from various regions of the country to have a stratified sample of names in the data bank. Two set of new functions were created for the implementation of soundex patterns with and without soundex characters separately. It is decided to have approximately 1% of the data bank as the test cases to experiment the 'H' factor significance in the search operation in a data bank.
As part of the experiment, both the soundex patterns were updated in the data bank with the respective functions. Each and every test case is taken one by one and searched over the data bank on both the soundex patterns N-Factor and H-Factor separately. The search findings are recorded as data observations for both the patterns separately for comparative analysis. The significance is analyzed in quantitative as well as qualitative dimensions over the number of matches and the hit list against each and every match.

It is observed that out of 31 test names only 24 names had a matching either in N-Factor or in H-Factor methods. That means, 7 names have no matches, neither in N-Factor nor in H-Factor methods. There are 24 test names had matches in N-Factor method and 15 test names had matches in H-Factor method. And, there are 15 test names that had matches both in 'N' factor and H-Factor methods. There are 9 test names had matches in N-Factor method and do not have any match in H-Factor method. This has been represented in Figure-1.

Result of test names (31):

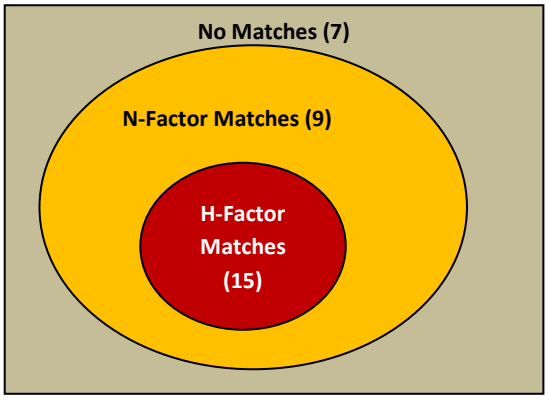

Figure-1

Let us analyze the quantitative and qualitative part of comparison of the number of matches observed in both the methods and their corresponding hit lists against each matches. It is observed that the test names that are matched in H-Factor method also matched in N-Factor and is a perfect subset of the N-Factor Matches. The same is illustrated in Table-1

Sl	Name	N-Factor Matches	H-Factor Matches	Total
1.	AMIT	6	6	12
2.	BHIM	9	0	9
3.	CHINMAYA	4	0	4
4.	DEEPAK	10	10	20
5.	DHAVAMANI	0	0	0
6.	ESRAR	0	0	0
7.	FARHA	0	0	0
8.	GIRI	2	2	4
9.	HARI	10	10	20
10.	IRANI	1	1	2
11.	JAISUNDER	4	4	8
12.	KALYAN	1	1	2
13.	LENIN	0	0	0
14.	MANISH	19	0	19
15.	NAMBI	0	0	0
16.	NANDU	2	2	4
17.	OM	7	7	14
18.	PANDIAN	0	0	0
19.	PERUMAL	2	2	4
20.	QUADEER	0	0	0
21.	RAVI	10	10	20
22.	SELVAM	2	2	4
23.	SHANKAR	13	0	13
24.	SRIDHAR	1	0	1
25.	SURESH	15	0	15
26.	THAMBI	1	0	1
27.	UMESH	4	0	4
28.	VENKATRAMAN	3	3	6
29.	VIJAY	21	21	42
30.	WASIM	1	1	2
31.	YOGESH	1	0	1

	Total	149	82	231

Table-1

It is observed that out of 231 match names, approximately 65 % of match names are from the N-Factor method and 35 % of match names are from the H-Factor method. The hit list ratio is given in Figure-2.

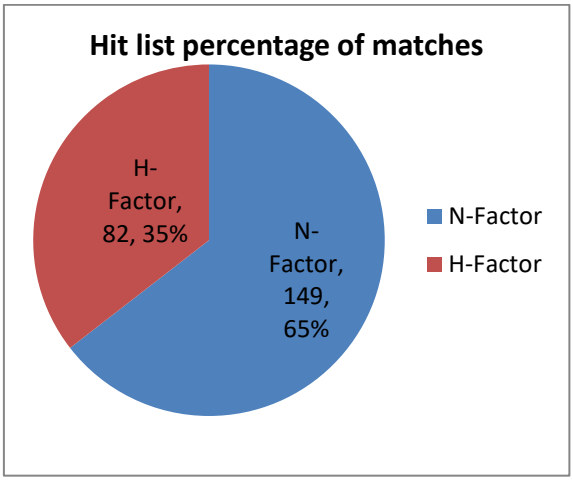

Figure-2

The effect of designing 'H' as non-soundex character shall reduce the number of matches and ultimately the related hit list against each match. The H-Factor is significant in defining and differentiating the names 'RAM' and 'RAHIM' as different names as the same is the universal truth. But, the N-Factor method just ignores this and proposes that the names are one and the same by giving matches with higher volume of hit lists of false positives for both 'RAM' and 'RAHIM'. The H-Factor may be very useful in filtering the false positives, but shall ignore the true positives as in the case of the name 'SUNITHA' and surname 'MISHRA' pointed out earlier. If the test name does not have the 'H' character alphabet, then it is observed that the number of matches and the hit list of each match remain same. This is because the soundex pattern for both the N-Factor method and H-Factor method are same if the test name do not have the 'H' character alphabet within the soundex pattern. This is represented in Figure-3.

If the test name has the 'H' character alphabet, then it is observed that there is no soundex pattern match found in H-Factor method while getting soundex pattern matches with sizable hit list from N-Factor method. This is represented in Figure–3 and Figure-4.

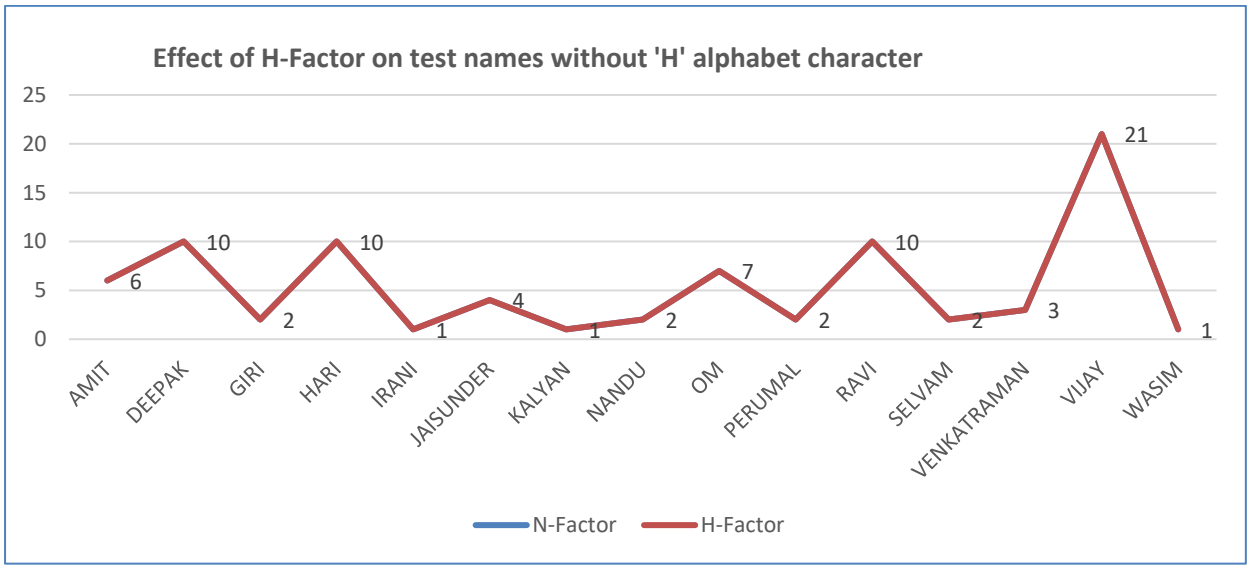

Figure-3

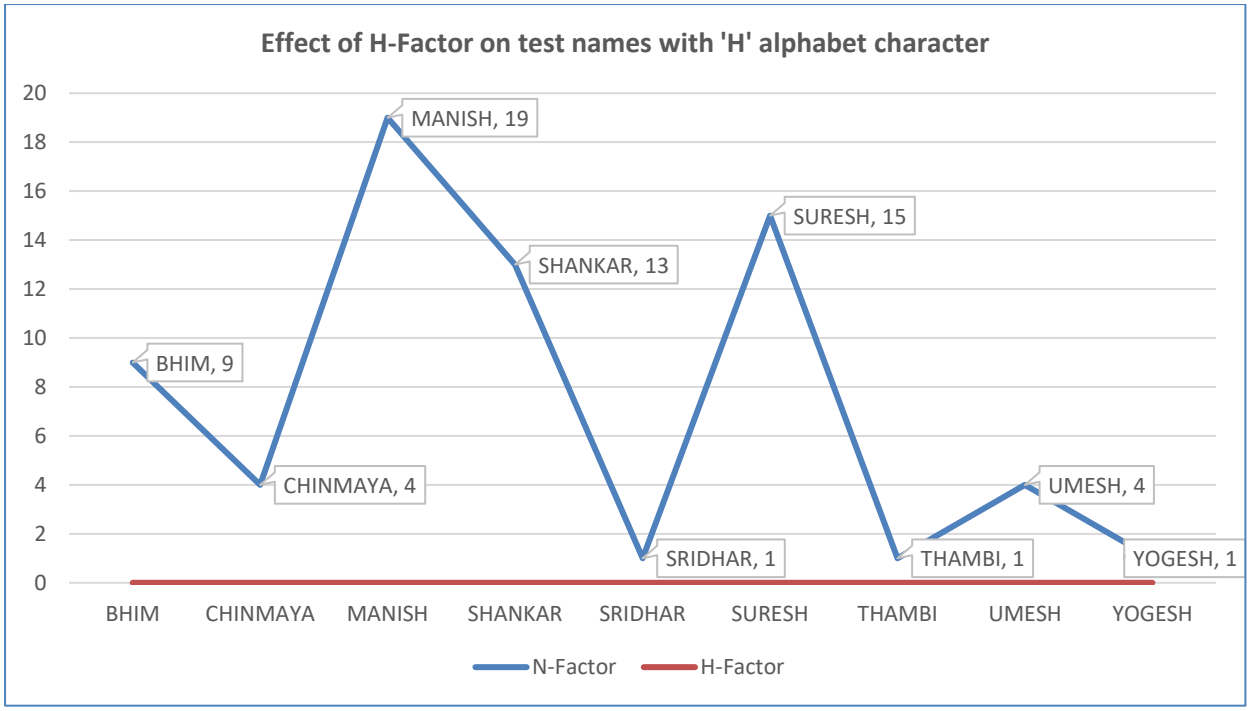

Figure-4

V. CONCLUSION

In handling soundex pattern technique, the main game play is to handle the alphabet character definition. Depends upon the requirement of the problem the whole English alphabets are to be identified in two basic forms called soundex and non-soundex alphabet characters. If required, some of the alphabet characters are to be dropped as in the case of standard soundex algorithm. From this experiment, we understand that, every alphabet character is to be taken at most care while defining whether the alphabet character is to be included into the soundex pattern or not. In case of inclusion, the same is to be decided whether to include the alphabet character in the form of soundex equivalent or to keep as non-soundex alphabet character while designing the soundex pattern. There may be some more alphabet characters in the English alphabets apart from the alphabet character 'H', for the consideration of experiment. Reasonable efforts have been made to analyze the significance of the alphabet character 'H' that plays a role in the soundex patterns of Indian names. The result may not be similar if any other alphabet character is defined as soundex or non-soundex characters and the same is to be experimented according to the need. This work shall further be improved to meet the requirement of the search over the digital libraries with the required level of customization.

REFERENCES

[1] Beider. A, Stephen P. Morse, Phonetic Matching: A Better Soundex, March, 2010.
[2] Beider. A, Stephen P. Morse, Phonetic Matching: An Alternative to Soundex With Fewer False Hits, 2008.
[3] Hall, P. A. V., and Dowling, G. R., Approximate String Comparison, Computing Surveys, 12, 381-402, 1980.
[4] Jaisunder G. C., Ahmed I., and Mishra R. K., "Need for Customized Soundex based Algorithm on Indian Names for Phonetic Matching", Global Journal of Enterprise Information System, 8(2), pp. 30-35, 2016.
[5] Justin Zobel, Philip Dart, Phonetic String Matching: Lessons from Information Retrieval, 1996.
[6] Mishra R K, "Information Technology as Management Tool for Process Re-Engineering and Preventing Forgery of Indian Documents", Jamia Millia Islamia, Central University, March 2010.
[7] Peter Christian, Soundex - can it be improved? March 1998.
[8] Sandeep Chaware, Srikantha Rao, "Analysis of Phonetic Matching Approaches for Indic Languages", International Journal of Advanced Research in Computer and Communication Engineering Vol. 1, Issue 2, April 2012.