Volume 2, Issue 12, December 2012 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Spell Checking Techniques in NLP: A Survey

Neha Gupta
Department of Computer Science,
Banasthali Vidyapith, Jaipur,
Rajasthan, India

Pratistha Mathur
Department of Computer Science,
Banasthali Vidyapith, Jaipur,
Rajasthan, India

Abstract: Spell checkers in Indian languages are the basic tools that need to be developed. A spell checker is a software tool that identifies and corrects any spelling mistakes in a text. Spell checkers can be combined with other applications or they can be distributed individually. In this paper the authors are discussing both the approaches and their roles in various applications.

Keywords: Type of Error, Error Detection, Error Correction, Spell Checker.

I. INTRODUCTION

When a modern spell checker for hindi would be used to spell check errors. All modern commercial spelling error detection and correction tools work on word level and use a dictionary. Every word from the text is looked up in the speller lexicon. When a word is not in the dictionary, it is detected as an error. In order to correct the error, a spell checker searches the dictionary for words that resemble the erroneous word most. These words are then suggested to the user who chooses the word that was intended. Spelling checking in used in various application like machine translation, search, information retrieval and etc .There are two main issue related to spell checker .These are error detection and error correction .In developing upon the type of error non word error and real word error .there are many techniques available for detection and correction .In this survey paper we are describing types of error in text, techniques for detection and correction.

II. TYPE OF ERROR

Techniques were designed on the basis of spelling errors trends these are also called error patterns. Therefore many studies were performed to analyze the types and the trends of spelling errors. The most notable among these are the studies performed by Damerau [2]. According to these studies spelling errors are generally divided into two types Typographic errors and Cognitive errors.

A.Typographic errors

These errors are occurring when the correct spelling of the word is known but the word is mistyped by mistake. These errors are mostly related to the keyboard and therefore do not follow any linguistic criteria. A study by Damerau [2] shows that 80% of the typographic errors fall into one of the following four categories.

1. Single letter insertion; e.g. typing acress for cress.
2. Single letter deletion, e.g. typing acress for ctress.
3. Single letter substitution, e.g. typing acress for across.
4. Transposition of two adjacent letters, e.g. typing acress for caress.

The errors produced by any one of the above editing operations are also called single-errors.

B.Cognitive errors

These are errors occurring when the correct spellings of the word are not known. In the case of cognitive errors, the pronunciation of misspelled word is the same or similar to the pronunciation of the intended correct word.
(e.g. receive -> recieve, abyss -> abiss).

III. ERROR DETECTION

A string of character in separated by space bar or puncation marks may be called a candidate words. A candidate word is a valid if it has a meaning else it is a non word. There are two techniques for error detection is N-garm analysis and dictionary lookup. The error detection process usually consists of checking to see if an input string is a valid index or dictionary word. Efficient techniques have been devised for detecting such types of errors. The two most known techniques are *n*-gram

analysis and dictionary lookup. Spellcheckers rely mostly on dictionary lookup and text recognition systems rely on *n*-gram techniques.

A. N-gram Analysis [3]

N-gram analysis is described as a method to find incorrectly spelled words in a mass of text. Instead of comparing each entire word in a text to a dictionary, just n-grams are controlled. A check is done by using an n-dimensional matrix where real n-gram frequencies are stored. If a non-existent or rare n-gram is found the word is flagged as a misspelling, otherwise not. An n-gram is a set of consecutive characters taken from a string with a length of whatever n is set to. If n is set to one then the term used is a unigram, if n is two then the term is a Bigram, if n is three then the term is trigram. The n-gram algorithm was developed as one of the benefits is that it allows strings that have differing prefixes to match and the algorithm is also tolerant of misspellings. Each string that is involved in the comparison process is split up into sets of adjacent n-grams. The n-grams algorithms have the major advantage that they require no knowledge of the language that it is used with and so it is often called language independent or a neutral string matching algorithm. Using n-grams to calculate for example the similarity between two strings is achieved by discovering the number of unique n-grams that they share and then calculating a similarity coefficient, which is the number of the n-grams in common (intersection), divided by the total number of n-grams in the two words (union) for example:

String 1 – "statistics"
String 2 – "statistical"
If n is set to 2 (Bigrams are being extracted), then the similarity of the two strings is calculated as follows. Initially,the two strings are split into n-grams:
Statistics - st ta at ti is st ti ic cs 9 Bigrams
Statistical - st ta at ti is st ti ic ca al 10 Bigrams
Then any n-grams that are replicated within the term are removed so that only unique Bigrams remain. Repeated digrams are removed to prevent strings that are different but contain repeated digrams from matching
Statistics - st ta at is ti ic cs 7 unique digrams
Statistical - st ta at ti is ic ca al 8 unique digrams
The n-grams that are left over are then arranged alphabetically.
Statistics - at cs ic is st ta ti
Statistical - al at ca ic is st ta ti
The number of unique n-grams that the two terms share is then calculated.The shared unique digrams is: at ic is st ta ti. In total 6 shared unique digrams.
A similarity measure is then calculated using the above-mentioned
Similarity coefficient using the following formula:
A = the sum of unique n-grams in term 1.
B = the sum of unique n-grams in term 2.
C = the sum of unique n-grams appearing in term 1 and term 2.
The above example would produce the result 0.80.
(2*6) / (7+8) = 12/15 = 0.80.

B. Dictionary Lookup

A dictionary is a list of words that are assumed to be correct. Dictionaries are represented in many ways, each with their own characteristics like speed and storage requirements. Large dictionary might be a dictionary with most common word combined with a set of additional dictionaries for specific topics such as computer science or economy .Big dictionary also uses more space and may take longer time to search. The non-word errors can be detected as mentioned above by checking each word against a dictionary. The drawbacks of this method are difficulties in keeping such a dictionary up to date, and sufficiently extensive to cover all the words in a text. At the same time one should keep down system response time. Dictionary lookup and construction techniques must be tailored according to the purpose of the dictionary. Too small a dictionary can give the user too many false rejections of valid words, too large it can accept a high number of valid low-frequency words. Hash tables are the most common used technique to gain fast access to a dictionary. In order to lookup a string, one has to compute its hash address and retrieve the word stored at that address in the pre constructed hash table. If the word stored at the hash address is different from the Input string, a misspelling is flagged. Hash tables main advantage is their random-access nature that eliminated the large number of comparisons needed to search the dictionary. The main disadvantage is the need to devise a clever hash function that avoids collisions. To store a word in the dictionary we calculate each hash function for the word and set the vector entries corresponding to the calculated values to true. To find out if a word belongs to the dictionary, you calculate the hash values for that word and look in the vector. If all entries corresponding to the values are true, then the word belongs to the dictionary, otherwise it does not.

IV. ERROR CORRECTION

Correction of spelling errors is an old problem. Much research has been done in this area over the years. Most existing spelling correction techniques focus on isolated words, without taking into account the textual context in which the string appears. Error correction consists of two steps: the generation of candidate corrections and the ranking of candidate corrections. The candidate generation process usually makes use of a precompiled table of legal n-grams to locate one or more potential correction terms. The ranking process usually invokes some lexical similarity measure between the misspelled string and the candidates or a probabilistic estimate of the likelihood of the correction to rank order the candidates. These two steps are most of the time treated as a separate process and executed in sequence. Some techniques can omit the second process though, leaving the ranking and final selection to the user. The isolated-word methods that will be described here are the most studied spelling correction algorithms, they are: *edit distance, similarity keys, rule-based techniques, n-gram-based techniques, probabilistic techniques and neural networks*. All of these methods can be thought of as calculating a distance between the misspelled word and each word in the dictionary or index. The shorter the distance the higher the dictionary word is ranked.

A. Edit Distance

Edit distance is a simple technique. Simplest method is based on the assumption that the person usually makes few errors if ones, therefore for each dictionary word .the minimal number of the basic editing operations (insertion, deletions, substitutions) necessary to covert a dictionary word in to the non word .the lower, the number ,the higher the probability that the user has made such errors. Edit distance is useful for correcting errors resulting from keyboard input, since these are often of the same kind as the allowed edit operations. It is not quite as good for correcting phonetic spelling errors, especially if the difference between spelling and pronunciation is big as in English or French.

B. Similarity Keys

A key is assigned to each dictionary word an only dictionary keys are compared with the key computed for the non word .The word for which the keys computed for the non word .The word for which the keys are most similar are selected as suggestion .such an approach is speed effective as only the words with similar keys have to be processed With a good transformation algorithm this method can handle keyboard errors.

C. Rule-based Techniques

Rule-based methods are interesting. They work by having a set of rules that capture common spelling and typographic errors and applying these rules to the misspelled word. Intuitively these rules are the "inverses" of common errors. Each correct word generated by this process is taken as a correction suggestion. The rules also have probabilities, making it possible to rank the suggestions by accumulating the probabilities for the applied rules. Edit distance can be viewed as a special case of a rule-based method with limitation on the possible rules.

D. N-gram-Based Techniques

N-grams can be used in two ways, either without a dictionary or together with a dictionary. Used without a dictionary, n-grams are employed to find in which position in the misspelled word the error occurs. If there is a unique way to change the misspelled word so that it contains only valid n-grams, this is taken as the correction. The performance of this method is limited. Its main virtue is that it is simple and does not require any dictionary. Together with a dictionary, n-grams are used to define the distance between words, but the words are always checked against the dictionary. This can b done in several ways, for example check how many n-grams the misspelled word and a dictionary word have in common, weighted by the length of the words.

E. Probabilistic techniques

They are, simply put, based on some statistical features of the language. Two common methods are transition probabilities and confusion probabilities. Transition probabilities are similar to n-grams. They give us the probability that a given letter or sequence of letters is followed by an16 other given letter. Transition probabilities are not very useful when we have access to a dictionary or index. Given a sentence to be corrected, the system decomposes Each string in the sentence into letter n-grams and retrieves word candidates from the lexicon by comparing string n-grams with lexicon-entry n-grams. The retrieved candidates are ranked by the conditional probability of matches with the string, given character confusion probabilities. Finally, a word-bigram model and a certain algorithm are used to determine the best scoring word sequence for the sentence. They claim that the system can correct non-word errors as well as real word errors and achieves a 60.2 % error reduction rate for real OCR text.

F. Neural Networks

Neural networks are also an interesting and promising technique, but it seems like it has to mature a bit more before it can be used generally. The current methods are based on back-propagation networks, using one output node for each word in the dictionary and an input node for every possible n-gram in every position of the word, where n usually is one or two.

Normally only one of the outputs should be active, indicating which dictionary words the network suggests as a correction. This method works for small (< 1000 words) dictionaries, but it does not scale well. The time requirements are to big on traditional hardware, especially in the learning phase.

V. AVAIABLE SPELL CHECKERS

There are many spell checkers for Indian languages are developed by using above techniques. This section provides brief discussion of some available spell checkers.

A. Bangla Spell Checker [5]

Bangla Spell Checker can act in both offline and online notes. It has a graphics user interface via an editor. Whenever the user types bangle text, it checks for wrong spelling and gives suitable suggestion. On the other hand, one can run this software on previously typed material such as – An OCR Document.For a single error word, word is found within top 4 words in the suggestion list. Words having more than 1 error are so captured and for most of them, words are in the upper half in the suggestion list. However, Suggestions cannot be given on some inflected words. It has facility to add new words in the dictionary against which spellings are checked.

B. Oriya Spell Checker [6]

Error detection and automatic or manual correction for miss spelled words, is taken care successfully by Oriya Spell Checker. There has been developed some algorithm to perform OSC in order to find out more accurate suggestion for a miss spelled word. The words are indexed according to their word length in word's data base in order for effective searching. On the basis of the miss spelled word, it takes into account the number of:

i) Matching Characters,

ii) Matching Characters in the forward direction,

iii) Corresponding Matching Characters in backward direction to give more accurate, suggestive words for the miss spelled word.

The OSC is running successfully in word processor using this technique, Hindi spell checker and English spell checker has also been developed for word processor. The word files are stored in ASCII Format and supports Oriya, Hindi and any font in English. This software is designed using Java and Java Swing for both the Windows 98 / 2000 / NT and the Linux – OS.

C. *Marathi Spell Checker [3]*

Under this ongoing project, a standalone spellchecker is being built for Marathi. The spellchecker will be available to spell Check document in a given Encoding. From the CIIL (Central Institute of Indian Language) Corpus, 12886 distinct words have been listed. A morphological analysis is being carried out on the collection of words. For e.g., An automatic grouping algorithm identified 3975 groups out of 12886 distinct word. 1st word is usually the route word. Thus there are approximately 4000 route words from Marathi Corpus. A manual proof reading will be done on these results. The morphology will be enriched.

D. Annam (Tamil Spell Checker) [7]

Tamil spell checker is used as a tool to check the spelling of Tamil words. It provides possible suggestion for erroneous words. User has the provision to select the suggestion among the list, ignore the suggestion or add the particular word to the dictionary. This module extract the route word from the given word (Noun / Verb) with the help of morphological analyzers and the route word is checked in dictionary and is found, the word is termed as correct word. Otherwise, the correction process is activated. The correction process includes error handling and suggestion generation module. After finding the types of error, the right form of suffix Noun or Verbs are given as input to the suggestion generation module. With the help of morphological generator the correct word is generated, And this module also handles the operation like – Select, change or Ignore the suggested word and adding the word to the dictionary.

E. Malayalam Spell Checker [8]

It is a software sub system that can be executed with Microsoft word as a macro or the Malayalam editor, developed by CDAC, Tiruvananthpuram, to check the spelling of words in a Malayalam text file. While running as a macro in a word, it functions as an offline spell checker in the sense that one can use this software with the previously typed text file only. Both offline and online checking are possible when it is integrated with the text editor. It generates suggestion for wrongly spelled words. The system based on dictionary look up approach. This module split the input words into route word, suffixes, post positions etc. Checks the validity of each using the rule database finally it will check the dictionary to find the route word is present in the dictionary. If anything goes wrong in the checking. It is detected as an error and the error word is reprocessed to get three 3 – 4 valid words which are displayed as suggestion. The user can add new words into a personalized database file, which can be added to the dictionary, if required. is integrated with the text editor. It generates suggestion for wrongly spelled words. The system based on dictionary look up approach. This module split the input words into route word, suffixes, post positions etc. Checks the validity of each using the rule database finally it will check the dictionary to find the route word is present in the dictionary. If anything goes wrong in the checking. It is detected as an error and the error word is reprocessed to get three 3 – 4 valid words which are displayed as suggestion. The user can add new words into a personalized database file, which can be added to the dictionary, if required.

F. Akhar (Punjabi Spell Checker) [9]

A language sensitive Punjabi / English spell checker has been provided in Akhar. Akhar can automatically detect the language and invokes the respective spell checker. The Unicode complaint Punjabi Spell Checker is font independent and can work on any types of the popular Punjabi fonts such as, Anantpur Sahib, Amritlipi, Jasmine, Punjabi, Satluj etc. This removes the contrast on the user to type the text in pre defined font only.

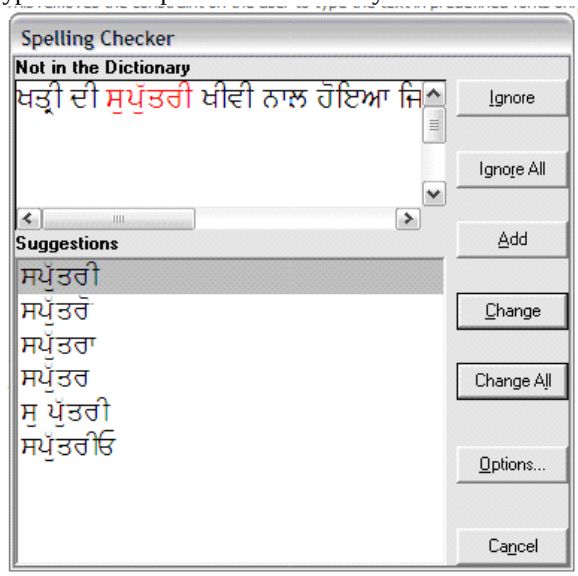

Fig1:snapshot of akhar(punjabi spell checker)

V. CONCLUSION

In this paper we have surveyed the area of Spell checking techniques. We have discussed various detection and correction techniques that are useful in finding the text with error. In future we will implement algorithm that is based on dictionary lookup techniques for detection and minimum edit distance techniques for correction of result in the area of Indian language spell checking.

REFERENCE

[1] Daniel Jurafsky, James H. Martin, *Speech and Language Processing*, PEARSON, 2nd ed.

[2] Dr.T.V Geeta ,Dr.Rajani Parthearath,"Tamil spellchecker",Resource centre for Indain language Technology Solution ,TDIL newsletter

[3] Dr. R.K Sharma ,"The Bilingual Punjabi English spell checker ," Resource centre for Indain language Technology Solution ,TDIL newsletter.

[4] "F.J Damerau(1964)"A technique for computer detection and correction of spelling error",Communicaation ACM.

[5] Http:// www..Baraha.com.

[6] Manisha Das,S.Borgohain,Juli Gogai,S.B Nair (2002),"Desingn and implemention of a spell checkers for Assamese",Language Engineering Conference.

[7] Mukand Roy ,Gaur Mohan,Karunesh K arora,"Compartive study of spell checker algorithm for building a generic spell checkers For Indian language C-DAC NODIA ,India.

[8] P.Kundra and B.B Charudhari (1999),"Error pattern in Bangla text ","international Jounral of Dravidian Linguistics.

[9] Prof.Puspak Bhattcharya and Prof. Rushikersh Josh, "Design and implantation of morophology based spellcheckers for Marathi",TDIL Newsletter.

[10] R.Ravindra Kumar ,K.G S ulochana,"Malayalam spell checker ",Resource centre for Indain language Technology Solution ,TDIL newsletter.

[11] Sebatian Deoprowicz,Marcing Ciura,"Correctingspelling errors by Modelling their causes,"Institute J.Appli.Math Computer Science ,2005,Vol15,no.2,275-285.

[12] Tanveer Siddiqui,U.S Tiwary,(2008),"Natural language processing and information Retrieval ",Oxford University.

.

.

