A MODEL FOR OVERLAPPING TRIGRAM TECHNIQUE FOR TELUGU SCRIPT

[1]B.Vishnu Vardhan, [2]L.Pratap Reddy [3]A.VinayBabu

[1]Department of computer science, Indur Institute of Engg &Tech (+ 91- 9848435073)
[2]Department of Electronics, J.N.T.U. Hyderabad (+ 91-9440629036)
[3]Director,SCDE, J.N.T.U. Hyderabad (+ 91-9396438343)

Email: [1]mailvishnu@yahoo.com, [2]prataplr@rediffmail.com, [3]dravinaybabu@yahoo.com

ABSTRACT

N-grams are consecutive overlapping N-character sequences formed from an input stream. N-grams are used as alternatives to word-based retrieval in a number of systems. In this paper we propose a model applicable to categorization of Telugu document. Telugu is an official language derived from ancient Brahmi script and also the official language of the state of Andhra Pradesh. Brahmi based script is noted for complex conjunct formations. The canonical structure is described as ((C) C) CV. The structure evolves any character from a set of basic syllables known as vowels and consonants where consonant, vowel (CV) core is the basic unit optionally preceded by one or two consonants. A huge set of characters that resemble the phonetic nature with an equivalent character shape are derived from the canonical structure. Words formed from this set evolved into a large corpus. Stringent grammar rules in word formation are part of this corpus. Certain word combinations result in the formation of single word is to be addressed where the last character of the first word and first character of the successive word are combined. Keeping in view of these complexities we propose a trigram based system that provides a reasonable alternative to a word based system in achieving document categorization for the language Telugu.

Key words: *canonical structure, Text categorization, trigram, bigram, conjuncts*

1. INTRODUCTION

Izumi Suzuki defined [1] a character as a discrete element that appears in the written or printed material of a language. It includes not only alphabetic and numeric elements but also punctuation marks, diacritical marks, mathematical and logical symbols, and other elements. A character code is a byte string that has a value indicating one of a set of characters, and an encoding scheme (or character set) is a particular mapping between characters and byte strings. ASCII (or US-ASCII) is one-byte encoding scheme commonly used for English, and

ISO 8859-1 is a superset of ASCII that covers the alphabet for Western European languages.

A script is a system of characters used for writing or printing a natural language. Latin script represents Western European languages and Devanagari for Hindi, Telugu, Kannada and Tamil scripts for Dravidian languages. Machine processing of document categorization demands for establishing a relation between coded sequence of characters and human perception of the language.

Text categorization is the problem of choosing a category from the catalogue or database that has common characteristics with selected text. However, creating such a database consumes more amount of time, because each text (or its part) is to be assigned to a correct category. Usually text is represented as a sequence of words and to find a correct category for it needs complex relations and computation methods of information about the text (number of words, relations between them, etc.). J.Trenkle and W. Cavnar introduced [2] N-gram Based Text Categorization as a simple method based on statistical information about the usage of sequences of characters. Prior to that whole words or sequences of words are used in text categorization. Initially

this method was designed for categorization by a language. N gram tests are carried out on categorization by subject with boundary specific restrictions

N-grams are consecutive overlapping N-character sequences formed from an input stream. An N-gram is just a string consisting of n symbols, usually taken from a text. Often this N-gram is made of symbols originating from the same word where length two is represented as bigram and n with three as trigram. N-grams are used as alternatives to word-based retrieval in a number of systems. DeHeer proposed [3] syntactic traces to demonstrate an efficient strategy for retrieval when thesaurus-based and multi-attribute search techniques are inadequate. Adams and Meltzer proposed [4] trigrams and inverted files for exact matches with query terms. They reported [5] 100% recall with high precision for their experiments and recommended trigram based search as an acceptable alternative to word-based search and a superior method for retrieval of word fragments. N-grams are used as an affective metric for TREC-2's retrieval and routing tasks with promising results. Since N-grams are [6, 7] language-independent, the strategies used for retrieval can be used for document collections in languages other than English. N-grams are used [8] along with word-based systems for effectively retrieving compound nouns in Korean documents. N-grams can be used [2, 9] to distinguish between documents of different languages in multi-lingual collections and to gauge topical similarity between documents in the same language. Retrieval based on N-grams is found [6, 10, 13] to be robust to spelling errors or differences and garbling of text. In the present work we propose a model for Telugu text retrieval using trigrams keeping in view of the complexities of the Telugu script.

2. BIGRAMS AND TRIGRAMS OF TELUGU WORDS

A general word string consisting of m letters leads to m+1 bi-grams, m+2 trigrams and m+n-1, n-grams. The theoretical number of possible n-grams is very high. For an alphabet of 26 letters there are $26^2 = 676$ bi-grams and $26^3 = 17576$ tri-grams possible. However, in English only 64% of these bi-grams and 16% of all tri-grams actually exist. It is this fact that allows for the detection of spelling errors. The n-grams that are possible from the Telugu word దశరథరాముడు are explained as the first case. N-grams are then constructed by sliding a window of length n over this string, progressing one symbol at a time. Redundant coding is applied in this case. The bi-grams formed in this way from the Telugu word దశరథరాముడు are:

ద దశ శర రథ థరా రాము ముడు డు .The word దశరథరాముడు with 7 letters results in 8 bi-grams.

From the same word 9 tri-grams are extracted and listed below.

ద *దశ దశర శరథ రథరా థరాము రాముడు ముడు* డు

Non-redundant coding uses word fragments with no overlaps. Then the word దశరథరాముడు yields:

*ద శర థరా ముడు with bi grams and *దశ రథరా ముడు* with trigrams. Wrongly spelled words usually have a large similarity with their correct version. There are some variations while attempting with bigrams, the last space missed in the above example. We consider a trigram index providing us better result while searching with a Telugu string.

Telugu script is syllabic, in the sense that vowels are represented differently in different contexts; the syllabic (primary) context and the intra-syllabic (secondary) context. That is, vowels have one form when they appear in a stand-alone form and in a different form when they appear in conjunction with consonants.

Any text contains many variant word forms, such as: "అక్కడ" "అక్కడే" "అక్కడా" "అక్కడకూడ" "అక్కడక్కడ" and so on. A conflation algorithm is a program that brings all these variants together into one word class. Clearly, words belonging to the same class have a very large bi-gram similarity. Similarity between two text strings can be measured indifferent ways. Using bi-grams we see that the similarity is observed between "అక్కడ" and "అక్కడకూడ" and presented as

అ అక్క క్కడ డ *అ అక్క క్కడ డకూ కూడ డ*

The entire index is to keep a duplicate copy of the index in which each of the index terms is spelled backwards. When a trigram based index is used, word suffix retrieval is simply a word fragment retrieval in which the last character of the trigram is a blank. In general a word is treated as a sequence of characters delimited by blanks. Word prefix retrieval is simply a word fragment

retrieval in which the first character of the trigram is a blank.

These issues are to be addressed while searching for a proper query in Telugu. Shannon proposed
[12] Dice coefficient (calculated as twice the number of bi-grams they have in common divided by the sum of the number of bi-grams in each word) is:

$$\frac{2*4}{4+6} = 0.8$$

While using the Jaccard index (another well-known similarity measure, equal to the number of bi-grams in common (a mathematical intersection) divided by the total number of bi-rams occurring in at least one of the two words (a mathematical union), this becomes:

$$\frac{4}{8} = 0.5$$

Exact values of similarity measures are usually not important. Their importance lies in the ranking they create. Term matching algorithms use these rankings. When searching for a word in a text or database the algorithm provides the user with a ranked list of words that have the largest similarity to the word used in the query.

While performing search in databases or the document with same word is not only of our interest but also the texts with similar words provides us extended boundaries for categorization. Robertson and Willett applied [13] this method successfully on databases containing old English texts. They queried the texts using modern English, but are able to recover many old-English spelling variants.

Meltzer and Kowalski suggested [14] using trigrams as the index entries in an inverted file retrieval system. Yochum used [15] them in hi document dissemination system. Trigram encoding is used [16] in text compression and to manipulate the length of index terms. D'Amore and Mah used [17] trigrams, as well as bigrams and other n-grams, as index elements for the inverted file system they implemented on a variety of machines. Traditional document retrieval systems have word based indexes. Meltzer and Kowalski specified [14] six distinct types of searches that a document retrieval system should be able to perform: word, word prefix, word suffix, word fragment, contiguous word phrase, and word proximity. While addressing Telugu words for categorization, extraction of N-grams are to be considered with similarity measure to achieve effective classification.

3. WORD STEMMING OF TELUGU TEXT

Some words possess the tendency of changing its meaning differently with the addition of a vowel at the end. This phenomenon is common in many languages. The specification lies within the language. Here we present a case of Telugu word.

సంస్థ

సంస్థ + ల+అ =సంస్థలా

సంస్థ + ల + ఉ = సంస్థలు

సంస్థ +ల + ఓ=సంస్థలో

సంస్థ +న +ఇ =సంస్థని

Each word which has been formed has different meaning and will be used in different context. To extract the base word stemming is to be imbibed with stringent grammar rules.

Overlapping trigrams are extracted from the words in the text by moving a window of three characters along a word, one character at a time, picking up each string of three characters as it appears in the window. When a word is a string of characters delimited by blanks, there will be as many trigrams extracted from each word as there are non-blank characters in the word. Using this method of extraction, no extracted trigram will contain more than one blank and no extracted trigram has a blank in its middle position since trigrams are not carried across words. Overlapping trigrams allow retrieval based on any sequence of three or more characters anywhere in a word since every trigram is indexed regardless of its position in a word.

Words with a common stem with suffixes and prefixes individually or combined is another complexity that has to be addressed. An illustration with the potential usefulness of being able to perform a retrieval based on a common trigram substring is presented below.

రాముడు

శ్రీరాముడు

సీతారాముడు

భార్గవ రాముడు

దశరథ రాముడు

The noun రాముడు is occurring in different positions in different contexts. At the same time this word is to be indexed regardless of its position. It is inevitable that using bigrams and trigrams only the distinction can be made between a word and a stem.

4. GRAMMAR RULES IN TELUGU

Telugu is a language that possesses a large amount of corpus where in stringent grammar rules provides a way out in the formation of a new word when two words are combined with specific rule without changing the basic meaning. This nature is distinct for Brahmi based scripts. There are around twenty rules that are common for all the languages derived from Sanskrit. At the same time an addition of 50 rules are found with Telugu script. These rules are called sandhi.

The basic nature of these rules defines a single word formation from two base words where the last character of the first word and the first character of the second word are combined using the canonical structure. Few examples are presented in figure1.

1.	ముని + ఇంద్ర = మునీంద్ర (సవర్ణ దీర్ఘ సంది)
2.	ప్రళయ + అగ్ని = ప్రళయాగ్ని (సవర్ణ దీర్ఘ సంది)
3.	మేన + అల్లుడు = మేనల్లుడు (అకార సంది)
4.	రాముడు + అతిడు = రాముడతిడు (ఉకారసంది)
5.	ఉప + ఇంద్రుడు= ఉపేంద్రుడు (గుణ సంది)

Figure 1 Sandhi based Examples

In the above example 1, 2 and 3 follows the canonical structure but 4, 5 doffers with the canonical structure yielding a new entity altogether.

Extraction of bigrams, trigrams from a word id found with a different complexity due to the nature of the grammar rules. It is necessary to analyze the grammar rule that is applied with in the word in the first stage and extraction of bigrams and trigrams is to be carried out in the second stage for these words. Keeping these complexities a model is proposed for document categorization using trigrams.

Proposed Model

Document categorization using words as a basis is a complex task due to the nature of the corpus that exists in the language. To reduce the computational complexity it is necessary to adopt a corpus database with base words. Trigram based categorization of words is one of the solution to achieve the above goal. The proposed algorithm is as follows. A detailed flow chart is can be seen in figure 2.

1. Read the query string
2. Split in to words
3. For each word
4. Find the length (n)
5. If (n>=3)
6. Check for validity
    If (valid)
        Add to corpus
            If (already exist)
                Discard
            Else
            Check for
            grammar rules and
            split
            Add to corpus
            Endif
    Else
        Split into trigrams
        Check for validity and
        add to corpus
        For remaining trigrams
        goto step 8
    End if
7. Else
8. Split them as bi grams and uni grams
9. Check for validity
    If valid add to corpus
    Else discard

End if

10. Endif
11. Repeat
12. End

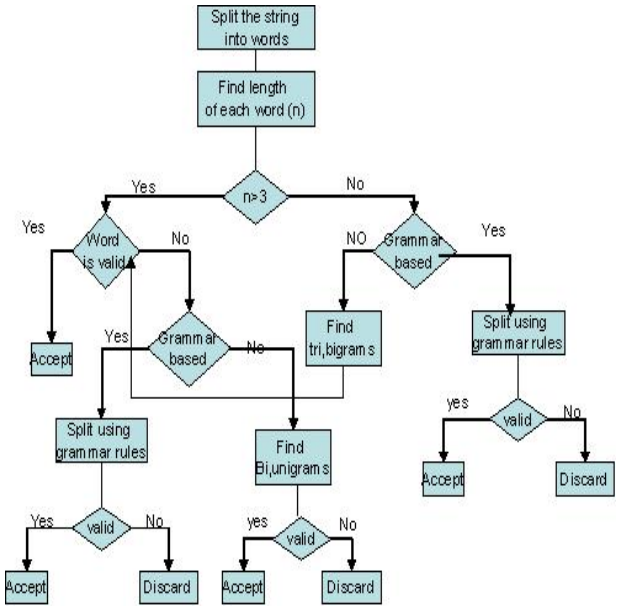

Figure 2. Text categorization flow chart.

In the proposed algorithm grammar rules based word segmentation and trigram based categorization are addressed. The separation of the word whose length is greater than or equal to three is checked with grammar rules then a trigram based separation is applied. Generally unigram, bigram words in Telugu never depend on grammar rules. Splitting the words and their classification involves the grammar rules and there by we can update and improve the existing corpus.

5. CONCLUSION

N-gram Techniques are language independent and they are well suited for different languages. For a complex language like Telugu, where any part of base corpus or derived from stringent grammar rules, text categorization found to be efficient using N-gram techniques. In this paper we proposed a trigram based system that can handle overlapping trigrams and also the grammar rules. Combining grammar rules derived from Sanskrit in to the algorithm is in progress.

6. REFERENCES:

[1]. Ariho ohsato, Izumi Suzuki, Yoshi mikami., *A language and character set determination method based on N-gram statistics,* ACM Transactions on Asian language information processing, Sep 2002

[2]. Cavnar, W. B., Trenkle, J. M., *N-gram-Based Text Categorization*, Symposium on Document Analysis and Information Retrieval, April 1994.

[3]. De Heer, T., *Experiments with Syntactic Traces in Information Retrieval*, Information Storage Retrieval, Volume 10, January 1974.

[4]. Adams, E. S., Meltzer, A. C., *Trigrams as Index Elements in Full Text Retrieval Observations and Experimental Results*, ACM Computer Science Conference, February 1993

[5]. Cavnar, W. B., *N-gram-Based Text Filtering for TREC-2*, The Second Text Retrieval Conference (TREC-2), February 1994.

[6]. Cavnar, W. B., *Using an N-gram-Based Document Representation with a Vector Processing Retrieval Model*, The Fourth Text Retrieval Conference (TREC-3), April 1995.

[7]. Cohen, J. D., *Highlights: Language- and Domain-Independent Automatic Indexing Terms for Abstracting*, Journal of the American Society for Information Science, 46(3), 1995.

[8]. Lee, J. H., Ahn, J. S., *Using n-grams for Korean Text Retrieval*, 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1996

[9]. Damashek, M., *Gauging Similarity with n-grams: Language-Independent Categorization of Text*, Science, Volume 267, February 1995.

[10]. Huffman, S., *Acquaintance: Language-Independent Document Categorization*

by N-grams, The Fourth Text Retrieval Conference (TREC-4), October 1996

[11]. Kukich, K., *Techniques for Automatically Correcting Words in Text*, Computing Surveys, 24(4):377-440, December 1992.

[12]. Shannon, C.E., *Prediction and entropy of printed English*, in: Bell System Technical Journal, 30 (1951), p. 50-64.

[13]. Robertson, A. M., Willett, P., *Searching for Historical Word-Forms in a Database of 17th-Century English Text using Spelling-Correction Methods*, 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1992.

[14]. Meltzer, Arnold C., and Kowalski, Gerald, "Text &arching Using an Inversion Database Consisting of Trigram", IEEE Proceedings of Second International; Conference on Computers and Applications. Pp.65-69, 1987

[15]. Yochum, Julian A., "A High-Speed Text Scanning Algorithm Utilizing Least Frequent Trigrams", IEEE Proceedings New Directions fn Computing @symposium, Trondheim, Norway, pp. 114121,1985.

[16]. Wisnieweki, Januxz L., "Effective Text Compression with Simultaneous Digram and Trigram Encoding", Journal of Information Science: Principles & Practice, Vol. 13, No. 3, pp. 159-164, 1987.

[17]. DeAmore, Raymond J., and Mab, Clinton, P., 'One-Time Complete indexing of Tex: Theory and Practice", Research and development in information retrieval: Eighth Annual International ACM SIGlR Conference, pp. 155164,Montmal, Quebec, Canada, 1985.

[18]. Huffman, S., Damashek, M., *Acquaintance: A Novel Vector-Space N-gram Technique for Document Categorization*, The Third Text Retrieval Conference (TREC-3), April 1995

[19]. Salton, G., Wong, A., Yang, C. S., *A Vector Space Model for Automatic Indexing*, Communications of the ACM, Volume 18, Number 11, November 1975.