Phonetic Distance Based Cross-lingual Search

Sriram S.*, Partha Pratim Talukdar, Sameer Badaskar, Kalika Bali, A. G. Ramakrishnan†

Hewlett-Packard Labs India
24 Salarpuria Arena, Hosur Road, Bangalore, India
Email: {*partha.talukdar, sameer.badaskar, kalika*}*@hp.com*

† Indian Institute of Science
Bangalore, India
Email: *ramkiag@ee.iisc.ernet.in*

*Birla Institute of Technology & Science
Pilani, Rajasthan, India
Email: *ssriram_rajan@yahoo.co.in*

Abstract

This paper proposes a novel approach for multilingual query processing, wherein we propose a phonetic distance based measure, for searching proper name data in Indian language scripts. The system allows query in a language of user's choice. A cross-lingual search is conducted with the query being in one language and the documents being searched for, in another. Grapheme-to-Phoneme converters are used to convert the user's query into an intermediate language-independent Common Ground (CG) representation. A dynamic time warping algorithm, wherein the substitution cost is based on a weighted phonetic distance measure, is used to match and rank the query results. In turn, a Phoneme-to-Grapheme converter is used to convert the search results in CG representation to the user's query language. We also discuss in detail the various issues particular to cross-lingual search on proper name data and address the same using the proposed approach.

1 Introduction

India as a multilingual society presents certain challenges for technology. One such issue is the way proper nouns coming from the same origin are written and pronounced in different Indian languages. This creates a difficulty when someone's name needs to be searched, for example, in a list of railway passengers. The problem is compounded due to the presence of certain characters/sounds specific to a particular language which do not have their counterparts in other languages. This is apart from the constraints imposed by various language scripts. All these combined, calls for a real need to perform an approximate match of proper names from a database in which names originate from different language sources. Other applications for cross-lingual search of proper names occur in looking for names in the database of Indian National IDs, in trying to trace a criminals record across different states of India. People who do not know the script of a particular language can still have access to information in that script by merely transliterating it into a script familiar to them. Ideally, such a system provides for the query data entry in any of the supported languages.

Considerable amount of work related to phonetic data storage and retrieval has been done in the recent past. Codes like *Soundex*, *Phonix*, *Editex*, *Metaphone* and *Caverphone* have been used to store phonetic information in the data. But these algorithms work only with English and rely on a strict character to code assignment that does not allow approximate matching between the query and the

stored data. The possibility of cross-lingual information retrieval using phonetic similarity has been explored in [Wei-Hao *et al.*, 2002]. It converts the query and the documents into IPA (International Phonetic Alphabet). But again, it looks for an exact match between query and stored data.

2 Phonetic Distance Approach

Unlike the aforesaid schemes, our scheme performs an approximate matching in phonetic domain between words across languages. This grants flexibility to the system as regards variations in languages and also allows for categorization and ranking of results based on similarity score. We propose a novel approach for multilingual query processing which allows proper name query in one Indian language from documents in another. The process involves three steps - (i) conversion of query into a language independent Common Ground (CG) representation (ii) query matching at CG level in the phonetic space and (iii) conversion of search results to the query language. By measuring the 'distance' between phonemes (CG symbols in this case) in terms of their constituent features, we generate a distance matrix that can be used by the DTW algorithm for matching the query to the database records. We use a phoneme set which serves as a universal set across the languages under consideration as a basis for the common ground (CG) representation.

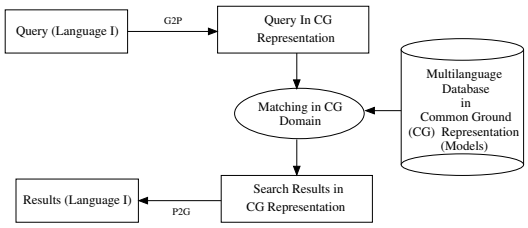

Figure 1: Block diagram of Cross-lingual Search

2.1 Common Ground (CG) Representation

Fig. 1 illustrates our approach to cross-lingual search. The CG representation consists of a set of symbols, each of which represents a particular sound in Indian languages. These symbols, correspond to different graphemes in different Indian languages. For instance, all the characters that sound similar to the phoneme /w/ are grouped under the symbol 'W' in the common ground. Fig. 2 shows the CG representative symbol 'W'. The two characters shown, from the languages Hindi and Tamil, are both

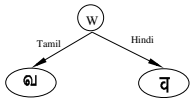

Figure 2: CG symbol 'W'

represented by the same CG symbol. The design of the set of common ground symbols is guided by the following considerations:

- *Character Superset*: Each character in any Indian language should belong to one of the CG categories/classes.
- *Extensibility*: The representation should be flexible enough to allow addition of a new language at a later point of time with minimal effort.
- *Removing ambiguity*: The grouping of characters in a language into a single class should not make the reverse mapping (i. e., from the CG symbols to characters) ambiguous.
- *Phonetic data storage*: The phonetic information needs to be preserved while allotting language specific characters to each of the common ground representations. This is important because the transliteration scheme is based on the phonetic nature of the languages.

Table 1: Phoneme features

Feature Name	Possible Values	Weight
Rounding (r)	rounded(+), unrounded(-), no specific value(0)	5
Frontness (f)	front(1), mid(2), back(3), no specific value(0)	5
Height (h)	high(1), mid-high(2), mid(3), mid-low(4), low(5), no specific value(0)	5
Length (l)	short(s), long(l), diphthong(d), schwa(a), no specific value(0)	0.5
Place of Articulation (p)	labial(l), dental(d), alveolar(a), retroflex(r), palatal(p), velar(v), uvular(u), no specific value(0)	4
Manner of Articulation (m)	stop(s), continuant(c), fricative(f), affricate(a), nasal(n), glide(g), taps(t), trill(r), flaps(p), laterals(l), no specific value(0)	4
voicing (v)	voiced(+), unvoiced(-), no specific value(0)	1
aspiration (a)	aspirated(+), unaspirated(-), no specific value(0)	0.5

2.2 Phonetic Distance based Matching

Phonetic distance is a means of representing difference between phonemes. In our method, the search query is first converted to the CG representation using the G2P converter. We have used HP Labs G2P converter [A.G Ramakrishnan *et al.*, 2003] for converting Indian language text to the CG representation. Since the CG symbols are analogous to phonemes, we can create a phonetic distance matrix, which is a measure of phonetic similarity between any two CG strings.

Thus, using phonetic matching, we can find the strings which are most likely to be similar in pronunciation to the given query string. The resulting CG strings can be converted to the query language using a P2G converter. These can be ranked based on their similarity or difference score. We now explain the concept of a phoneme feature vector.

2.3 Phoneme Feature Vector

Speech sounds occurring in a language can be described in terms of a finite set of attributes called phonetic features. For example, the sound [p] can be further analyzed as a consonant [+consonant], it is produced with a full stop or closure [+stop], the closure is at the lips [+labial] and the vocal cords are delayed in vibrating after the release of the stop closure [-voice].

The CG symbols can be represented by 8 dimensional feature vectors, where the initial four are vowel specific and the subsequent four features, consonant specific. This is given in Table 1. The feature vector for a CG symbol/phoneme is of the form $\{r, f, h, l, p, m, v, a\}$ The individual features of a phoneme vector can be binary or multivalued ($\geq$ 3 values). The value(s) that each of the phoneme features can take are listed in Table. 1. For instance, the feature vector for the CG symbol k(0) = {?,?,?,?,v,s,-,-} represents a velar stop consonant which is unvoiced and unaspirated. The similarity/dissimilarity measure between the phonemes can be given as a function of their feature vector specifications. In all its generality, difference between phonemes P_1 and P_2 = d(feature vector for P_1, feature vector for P_2) where d(.) is the distance metric as defined below.

2.3.1 Weighted Distance

The distance metric has to be derived keeping in mind the fact that all features are not equally important in determining the difference. For example, the distance between the phonemes /p/ and /b/ is greater than that between the allophones /p/ and /p^h/. The first pair differs in voicing whereas the second differs only in aspiration. Since the former is perceptually much different from the latter case even though the difference as such lies in only one feature, it is sensible to give more weightage to the voicing feature than the one for aspiration. With this reasoning, we assign relative 'importance' to features in a phoneme vector when calculating the distance. The feature weights are given in Table.

1. The discriminative capabilities of the various phonetic features are discussed in detail in [Kondrak G., 2000]. It must also be noted that the method of deriving the feature weights in a principled manner still remains an open problem. We precompute the distances for all the possible values of the particular feature and store these in Attribute Difference Matrices (ADMs). An ADM for a feature is a symmetric matrix of order n x n with principal diagonal elements as zeros and n being the number of distinct values the feature can take.

2.3.2 Difference Function

Given $\mathbf{x}$ and $\mathbf{y}$ are the feature vectors for phonemes P_1 and P_2, the difference function given by

$$d(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{N} w_i d(x_i, y_i) \tag{1}$$

where $d(x_i, y_i)$ is obtained from ADM for i^{th} feature, w_i is the weight of feature i and N the dimensionality of the phonetic feature vector. This function is applied to each pair of CG symbols and the pairwise distances are stored in a distance matrix. This matrix forms the inputs to the Dynamic Time Warping (DTW) algorithm.

2.4 DTW Alignment and Query Matching

DTW is used to compute the best *alignment warp* between the search query (called the data) and the stored CG representations (called models) in the database. Suppose, in CG domain, the query string Q and the model, C are given by $Q = q_1, q_2, ...q_n$ and $C = c_1, c_2, ...c_m$, the objective is to obtain a warping path $W = w_1, w_2, ...w_k \;\; k < m+n-1$ subject to certain boundary conditions, continuity and monotonicity requirements which minimize the warping cost / alignment score [Keogh *et al.*, 2000]. We estimate the warping path through the following recurrence relation.

$$c(i, j) = d(q_i, c_j) + min\{c(i, j-1), c(i-1, j-1), c(i-2, j-1)\} \tag{2}$$

where $d(q_i, c_j)$ is the phonetic distance given by (1). The cost $c(m, n)$ (termination point) is taken as the alignment score between the query Q and the model C. From this cell, the warping path is obtained by tracing back along the parent cells.

DTW is applied to match the query with each word in each model (under the assumption that the search is performed on names and other such strings). The minimum of the alignment scores of the query with the words in a model is taken as the alignment score between the query and the model.

3 Issues

3.1 Special characters or sounds

There are special characters (sounds) in some languages (Eg. the character /zh/ in Tamil) that do not have counterparts in other languages. In this case, the method maps the special character to the CG representation which best describes its phonetic value. For instance, there are three /l/ like sounds in Tamil. We store all these symbols under the CG representation say, 'L'. We distinguish between these three as L(0), L(1) and L(2). A query string in Hindi for the word *Tamil* can still match its counterpart in a Tamil document, as the phonetic distance based matching results in a close match. This is depicted in Fig. 3.

3.2 Character-set constraints

The correspondence of a single grapheme to multiple phonemes in certain languages. E.g. the grapheme 'க' in Tamil corresponds to the sounds /k/, /kh/, /g/ and /gh/. Consider the case, where a user wishes to search for *Gamini*, the Tamil query string can only be *Kamini* because of character-set constraints in Tamil script. Using the proposed phonetic distance based approximate matching, the desired result can be ensured to be in one of the top matches. Fig. 4 illustrates this. Thus, one to many mapping between grapheme and phonemes in certain languages can be addressed to a certain extent by phonetic distance based matching.

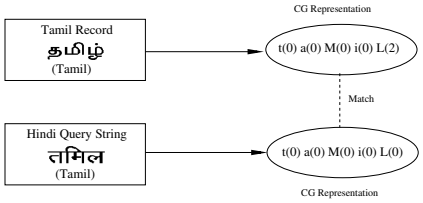

Figure 3: Example of match at CG level

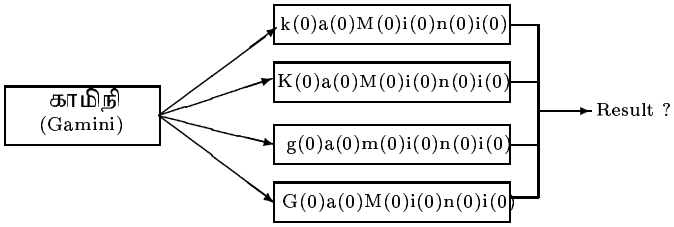

Figure 4: Example of match at CG level showing one to many mapping

3.3 Ranking of results

The search results, which are displayed to the user, must be phonetically unambiguous. This implies that the user must know how far the results that are displayed, conform to his search query. A solution to this is a phonetic distance based search which gives a quantitative measure of the accuracy thus enabling categorization of search results.

4 Conclusions and Future Directions

We have presented a generic framework for cross-lingual search, which is based on a phonetic distance measure, employing an 8 dimensional feature vector and a weighted distance metric as a basis for computing the DTW alignment score. Some issues pertaining to cross-lingual search and the manner in which they are handled by the algorithm have also been described. Future work will focus on improving in the efficiency of DTW matching and incorporating more languages into the framework. We have identified the need for cross-lingual search of proper name data and the work presented in this paper is intended to serve as a step in that direction.

References

[A.G Ramakrishnan *et al.*, 2003] Kalika Bali, Partha Pratim Talukdar, N. Sridhar Krishna, A.G. Ramakrishnan. Tools for the Development of a Hindi Speech Synthesis System. *5th ISCA Speech Synthesis Workshop*, pp. 109/302/255114, Pittsburgh, 2004.

[Wei-Hao *et al.*, 2002] Wei-Hao Lin, Hsin-Hsi Chen. Backward Machine Transliteration by Learning Phonetic Similarity. *Proceedings of Sixth Conference on Natural Language Learning (CoNLL)*, Taipei, Taiwan, August 31 - September 1, 2002.

[Kondrak G., 2000] Kondrak G. A New Algorithm for the Alignment of Phonetic Sequences. *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics (ANLP-NAACL 2000)*, pp. 288 - 295, Seattle, April, 2000.

[Keogh *et al.*, 2000] Keogh, Eamonn J. Exact Indexing of Dynamic Time Warping. *Very Large Databases (VLDB) Conference*, pp. 406 - 417, 2000.