Issues in the Design of a Spell Checker for Morphologically rich Languages

K. Narayana Murthy
Department of Computer and Information Sciences,
University of Hyderabad, Hyderabad, 500 046,
email: knmcs@uohyd.ernet.in

Abstract

In this paper we discuss the various issues relating to the design of spell checkers for language which exhibit a very rich system of inflectional morphology. Kannada, a Dravidian language spoken mainly in the southern state of Karnataka in India, is taken as an example. We start by briefly sketching the design issues for spell checkers in general. Next we take up the issues that need to be considered in designing a good spell checker for languages such as Kannada and Telugu where each root can participate in the formation of hundreds of inflected and/or derived words. Spelling error detection as well as correction issues are discussed and the importance of morphological analysis and generation are highlighted. Finally, a computational technique for improving the performance and efficiency of spell checkers for such languages is described.

1 Introduction:

A spell checker is a computer program that deals with the detection and correction of spelling errors in texts. An ideal detector should pin point all and only those words in the text that are wrongly spelled. An ideal corrector should automatically correct all such errors. In practice, neither detection nor correction can be perfect. Most practical spell checkers do not even attempt to automatically correct spelling errors - instead they only offer list of suggested alternatives for the user to choose from. The user has several options: he may select one of the suggestions provided by the system, he may direct the program to accept the specific instance or even all instances of current spelling in the text, he may direct the system to add this "new" word to the custom dictionary, or he may choose to edit the word manually.

A good spell checker must detect all or most of the spelling errors and at the same time minimize false alarms. It must offer either the single *right* suggestion or a small set of suggestions for correction in which the *right* suggestion is included. The suggestions offered must be ranked and the *right* suggestion must occur in the top of the list in a

majority of cases.

Most practical spell checkers work one word at a time and hence cannot even detect *real word errors* - mistakes that result in some other valid word in the language, as against *non-word errors* - mistakes that result in an invalid word. For example, most spell checkers will not find anything wrong in the sentence "One plus one is tow". Catching real word errors requires context based processing. If not full syntactic parsing, at least a statistical or linguistic processor that considers the textual context in which a word is used, must be employed.

Detection and correction can both be done using dictionaries and morphology and other linguistic tools, using statistical techniques, or using a combination of both. In a dictionary based approach, a word not found in the dictionary is considered to indicate a spelling error and other words from the dictionary which are *close to the input word in terms of spelling* are given out as suggestions for correction. In a simple statistical approach, the probability of a particular alphabet sequence in the language is used instead as the basis. The survey paper by Karen Kuckich [1] describes a variety of techniques for detection and correction of spelling errors.

In this paper we discuss the design of a dictionary based non-word spelling error detection and correction system for Kannada. We start by looking at the issues that crop up in languages such as Kannada that exhibit a very rich system of morphology.

2 What is a spelling error?

All through the previous section, we used the term *word* to mean simply a sequence of alphabets separated by spaces. Thus a word is seen here from the perspective of its spelling rather than from the perspective of pronunciation or meaning. The mapping from spelling to meaning is usually arbitrary.

In languages like English, the mapping from spellings to pronunciation is also quite ad-hoc. The same alphabet gives rise to different sounds in different contexts and the same sounds can be realized using a variety of spelling combinations. Thus spellings have to be learnt and carefully remembered and people naturally tend to make mistakes. The origin of spelling errors can thus be cognitive. Of course phonetic and typographical errors are also possible.

Indian languages, on the other hand, are primarily phonetic in nature - the orthography reflects the phonetics to a large extent. Thus there is really no such thing as *spelling*. Nevertheless, there is scope for mistakes of a variety of kinds and techniques of spell checking can be applied just the same.

One common mistake in many Indian languages is the use of an aspirate for the cor-

responding non-aspirate or vice versa - is it *sambaMdha* or sambhanda?. Modern Indian languages have many words borrowed and/or assimilated from Sanskrit. While Sanskrit uses aspirates and non-aspirates distinctively, the distinction may be inherently less prominent or absent in the native language and hence the confusion. These are cognitive errors.

Phonetic errors are also not inconceivable. *aDugemane* (kitchen) is quite often pronounced and written as *aDigemane*. *huDugi* often becomes *huDigi*. Non-initial vowels appear to be less important. Frequent and regular usages like *huDgi* especially in the spoken variety only substantiate this view. Even *huDagi* may be acceptable. This results in variations in spelling. In Kannada, Word initial *ha* is often pronounced as *a* - *aasana* (seat or chair) for *haasana* - the name of a place. One may become conscious of this error and may even resort to hyper-correction. These effects in pronunciation can be reflected in orthography leading to spelling errors.

Typos are also possible for typed text although it is possible to design editors that disallow right away certain invalid combinations of symbols. For example, an editor may prevent an accidental second maatra for a consonant. Nonetheless, it will not be possible to prevent typos in all cases.

Further, there are often variations in spellings, some of which may be more acceptable than others for specific purposes and thus spelling error correction can be viewed as more of a *standardization* effort. In Kannada, the spoken and written varieties of language are quite distinct (example - barutteene vs. baruttiini). Dialectal variations also exist. There are loan words whose native sounds cannot be represented directly in orthography. There is no phonetically accurate way to write *bank* in Kannada and naturally new conventions have come to use. A common rendering is *byaank*. There is representation for the 'fa' sound and a convention that has come to widespread use is to write two dots (a special kind of nukta) below the 'pa' letter - as in *kaapi* (coffee). However, these conventions are not followed uniformally in all cases thus leading to variations in spelling. The half consonant 'ra' in consonant clusters as in *muurti* is written in Kannada as a special symbol called *arkaavattu* and is written by convention after the latter consonant in the cluster. However, in recent writings, the default method of writing consonant clusters is used quite often instead of the 'arkaavattu'. Standardization is highly desirable where the texts are intended to be subjected to automatic language processing tools. Otherwise there would be difficulties in searching, sorting and many other such processes.

3 Spell Checking in Morphologically Rich Languages:

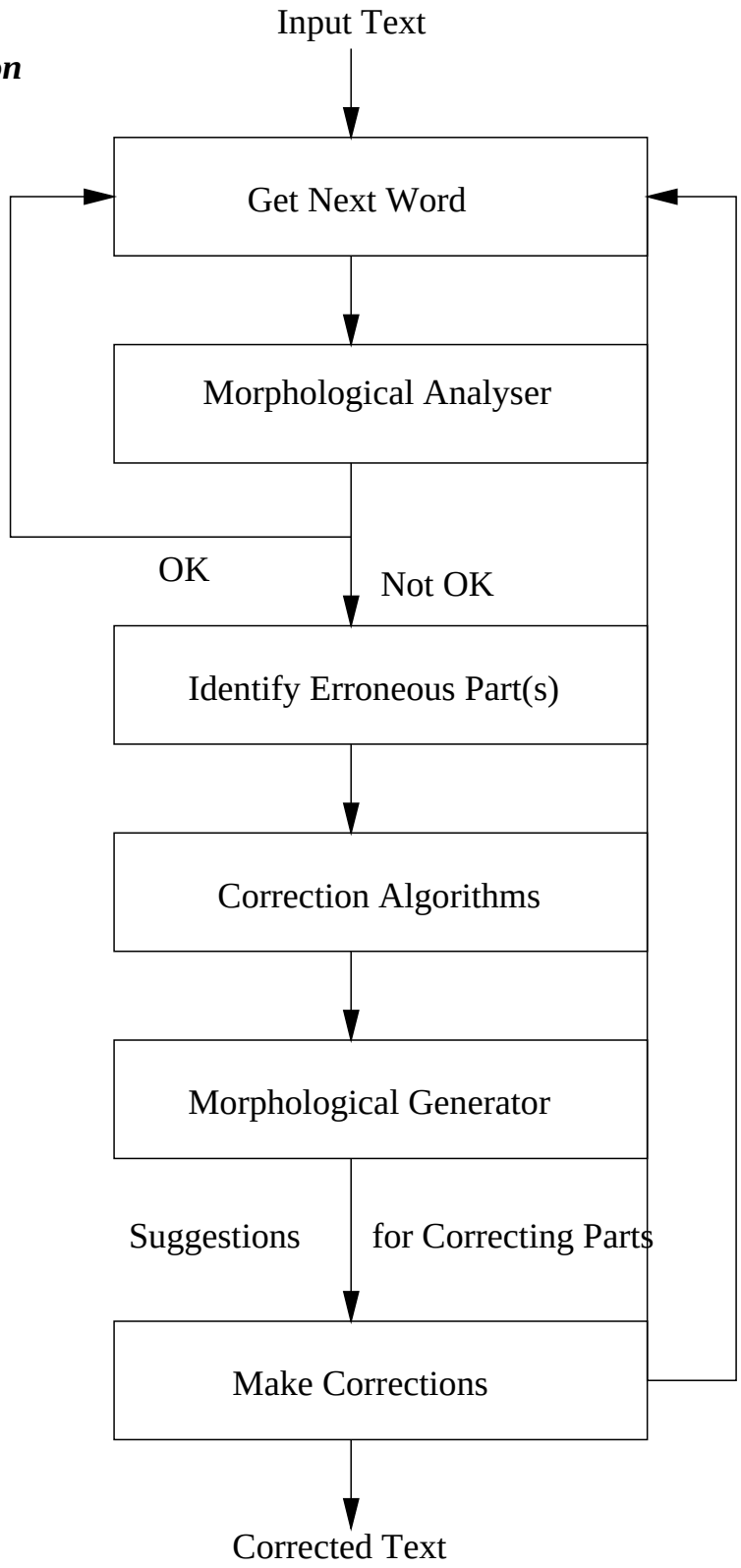
Although English has a rich and complex system of derivational morphology, inflectional morphology is quite simple and straight forward. Most spell checkers for English therefore store the derived forms directly in the lexicon and apply rules of morphology only for a few cases where the rules are simple and highly productive. This approach is practically feasible and reasonably efficient for languages such as English.

In languages such as Kannada, a verb root may give rise to several hundred complete words. A verb form may include several aspectual auxiliaries, clitics, particles and vocatives, apart from tense, gender, number and person suffixes. A word in Kannada often corresponds to a whole phrase in English. Thus 'although (I/we/you/he/she/it/they) had certainly wanted to come' would be just one word in Kannada - roughly 'baraleebeekeMdukoMDiddaruu'. Inter-word saMdhi and compounds add to the problem. See [2] and [3] for more on Kannada morphology. It is practically not feasible to store all forms of all words in the dictionary. A detailed morphological component is essential for developing a good spell checker for languages such as Kannada.

In order to detect spelling errors, every word in the text has to be morphologically analyzed and checked with a dictionary. Only then can we accept all and only the valid words and flag the others as erroneous. Spelling errors may occur in the roots, in the various affixes or in the internal saMdhi that glues these parts into the whole word. Once the source of the error is found, appropriate suggestions for correcting that part may be generated using a variety of techniques [1]. Finally, the morphological generator would be called to put together the correct parts to re-build the complete word form. Thus 'hugaLannu' will be analyzed as 'hu + gaLu + annu' (noun + plural + accusative), 'hu' corrected to 'huu' (flower)', and then the correct form 'huugaLannu' generated. Similarly, 'huugalannu' can be corrected as 'huugaLannu'. Complexity increases if there are multiple errors in a word. However, preliminary studies have showed that cases of multiple errors are very rare.

The block diagram below gives the overall structure of the spell checker that is being developed for Kannada. The techniques used here can also be applied fruitfully for other languages. The morphological analyzer and generator are implemented in the *Network and Process model* [3]. The current implementation is in Prolog. Web based tools and services for spell checking are also proposed to be developed at a later stage.

Isolated Word Error Detection and Correction System



A small prototype system has been implemented and tested. A full-blown version is under development. No attempt will be made to *standardize* the language. Instead, the user will be given a number of options. He can, for example, decide whether words from

a specific dialect should be or should not be accepted. Further, he may decide whether an attempt should be made to correct the deviations or simply indicate or warn the user of such usage.

4 Techniques for improving the spell checker:

In this section we describe a simple computational technique that can substantially simplify the work load of the spell checker and thereby give better performance in terms of speed.

It has been observed above that a verb root in Kannada can give rise to hundreds of word forms. Each such word form may include several levels of affixation and corresponding internal saMdhi processes. Thus *tiMduhaakiddanu* ((he) had eaten) can be analyzed as

tinnu	i	haaku	i	iru	id	anu
Root: (eat)	past-part.	asp. aux.	past-part.	Perfective	Past	m,sl,p3

Morphological analysis and generation is quite an involved and complex task. Often there are six or seven levels of affixation, there being several possible affixes at each level. A language may have a large number of verb roots, each of which may take many of these combinations of affixes, thus leading to combinatorial explosion.

However, morphological processes are usually fairly uniform and so combinatorial explosion can be avoided. An excellent technique for reducing the complexity of any problem is to replace multiplication with addition. This can be done by reducing the number of levels of combination with the attendant increase in the number of possibilities at each level. When reduced to two levels, this means that the dictionary will include the verb roots as also the valid combinations of affixes. We would then only need to pick up the right verb and the right suffix cluster and put them together using the appropriate rules of internal saMdhi. There will be only one level combination, only one application of a saMdhi rule. This technique is especially suited for interactive spell checking, where speed is important. This technique can be used as a purely internal mechanism, invisible to the user. The system should be capable of doing complete and detailed analysis and give appropriate explanatory responses for purposes other than spell checking.

5 Conclusion:

In this paper we have briefly sketched the issues concerned with the development of good spell checkers, especially for languages with a rich system of morphology. The need for a robust system of morphological analysis and generation has been highlighted. A simple computational technique of reducing the computational complexity of the system without externally violating the linguistic requirements has also been described. Although Kannada has been taken as an example, the basic techniques outlined here can be applied for other languages too.

6 References:

[1] Karen Kukich, "Techniques for Automatically Correcting Words in Text", ACM Computing Surveys, Vol 24, No. 4, December 1992, pp 377-435

[2] S. N. Sridhar, "Kannada", Routledge, 1990

[3] K. Narayana Murthy, "A Network and Process Model for Morphological Analysis and Generation", Second International Conference on South-Asian Languages ICOSAL-II, January , Patiala, India.