4

Tamil Spell Checker

T Dhanabalan, Ranjani Parthasarathi, T V Geetha

Resource Centre for Indian Language Technology Solutions – Tamil,
School of Computer Science and Engineering, Anna University, Chennai, India.
*E Mail {**dhanabalan@cs.annauniv.edu**, **rp@annauniv.edu**, **tvgeedir@cs.annauniv.edu** }*

Abstract

Design and development of Spell Checker for Tamil language and details of the implementation have been discussed in this paper. Lexicons with morphological and syntactic information are needed for the development of spell checker that can be integrated in word processors, as well as for the development of morphological and syntactic analyzers that can be exploited by more complex natural language processing applications.

1. Introduction

Lexicons with morphological and syntactic information are needed for the development of spell checker, morphological and syntactic analyzers. These can be incorporated in the complex natural language processing applications.

Tamil is a morphologically rich language in which most of the morphemes coordinate with the root words in the form of suffixes. Person, gender and number markings combine with root words and normally match with person, gender and number markings of the subject of the sentence. In addition auxiliaries, which convey modal, aspect, etc. also combine with the main verb and form a cohesive unit. Unlike most other languages, in Tamil, case markers occur along with the nouns along with number markings. In fact, in Tamil, case markings serve as an indication that the word is a noun.

Spell checking applications present valid suggestions to the user based on each mistake they encounter in the user's document. The user then either makes a selection from a list of suggestions or chooses to ignore the suggestions and accepts the current word as valid. Regardless of how often this is done, the spell checking application will perform its task independent of the types of mistakes most commonly made by that particular user [3].

A spell checker program is often integrated with word processing software for checking the correct spelling of words in a document. Each word is compared against a dictionary of correctly spelt words. The user can usually add words to the spell checker’s dictionary in order to customize it to his or her needs. In order to catch the misspelt words, the checker would need to incorporate syntactic and semantic knowledge.

The tool provides a facility to customize the spell checker's dictionary so that the words like Company's name, technical words related to a particular domain and proper nouns etc can be appended as per the user's need.

2. Difficulty in Tamil Spell Checking

In Tamil Spell checker the following issues have to be tackled.

1. Handling the case endings (Suffixes).

In English language the prepositions come as a separate word before the noun. But in Tamil language these prepositions come as case endings or postpositions.

Ex.

- Ram went to Delhi

  Ram Delhiikku cenRaan.

Here the preposition 'to' comes as a separate word in English. But in Tamil that comes as case ending ('kku') and is attached to the noun.

- Raman tells about Krishnan.

  Raman krishnanaip patRi solkiRaan.

Here the preposition 'about' comes as above. But in Tamil it comes as a postposition ('paRRi').

2. Classification of the root word as noun or verb by the case endings.
3. Handling the short vowel and long vowel errors.

   E.g. 'kiNaRu' can be misspelled as 'keeNaRu'

4. Handling the Sandhi letter between two words.

   E.g. 'avanukkup piRaku' Here the sandhi letter 'p' joins the two words.

3. The modules of the Tamil Spell Checker

General flow for the Tamil Spell checker is given in figure 1. Initially the Spell Checker reads extracted words from the document, one at a time. The dictionary then examines the extracted words is obtained. If the word is present in the dictionary, it is interpreted as a valid word otherwise the next word. If a word is not present in the dictionary, it is forwarded to the Error correcting process.

The spell checker consist of the following phases namely text parsing, spelling verification and correction, and generation of suggestion list. To aid in these phases, the spell checker makes use of the following.

- Morphological analyzer for analyzing the given word
- Morphological generator for generating the suggestions.

Text parsing isolates a word suitable for spell checking from its surrounding text in the input document. This word is taken for spelling verification. If the word found is incorrect, it is further passed onto the spelling correction and suggestion generation phases. After parsing the document into a list of words, each word is passed to the morphological analyzer. The morphological analyzer first tries to split the suffix. It is designed in such a way that it can analyze only the correct words. When it unable to split the suffix due to spelling mistake, it

passes the word to spelling verification and correction phase to correct the mistake. After splitting the word into the root word and a set of suffixes, the root word is checked for its validity by comparing against the dictionary. If the word is not present in the dictionary, then the nearest matching words are chosen as the suggestions.

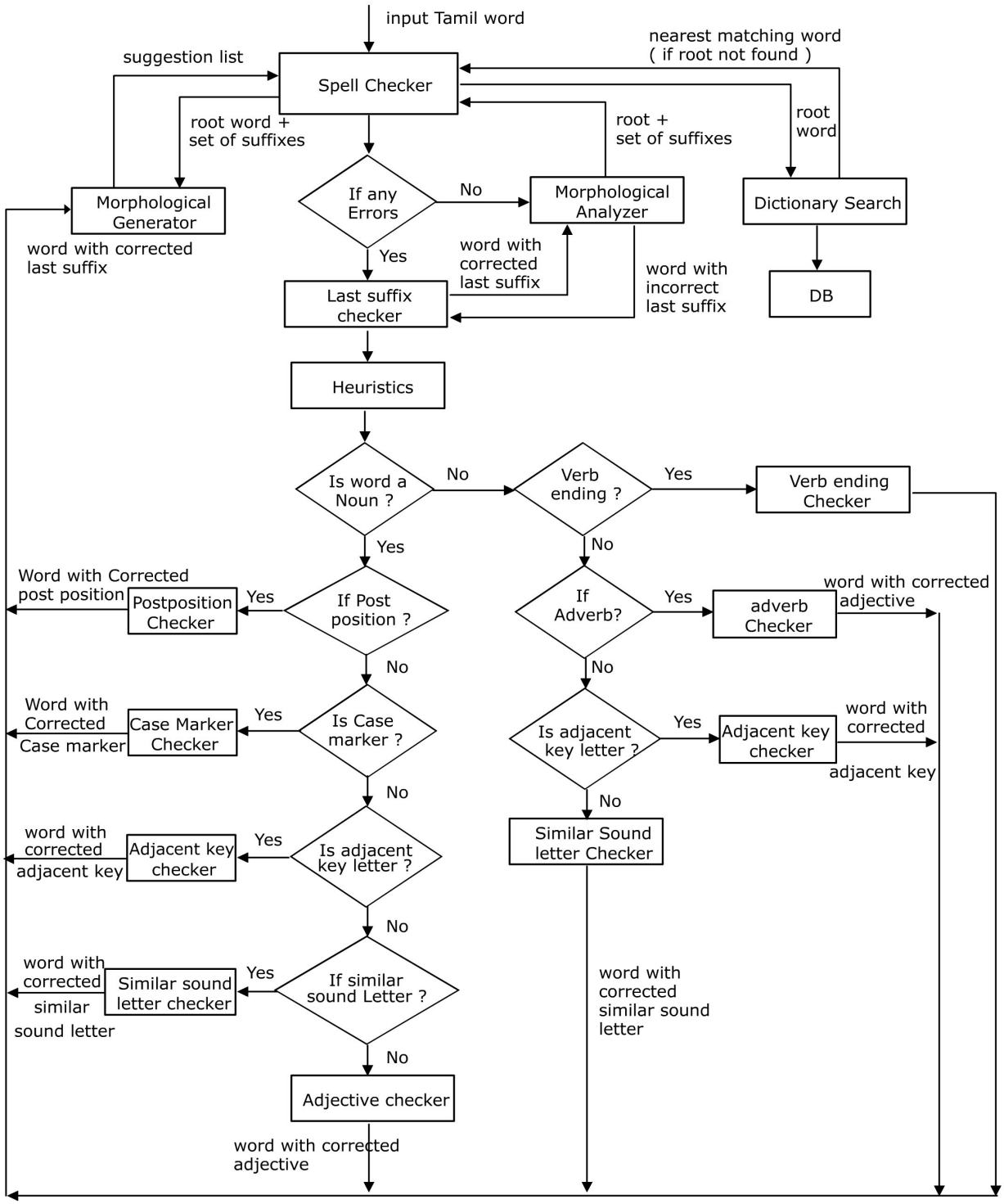

Figure 1. General flow of Tamil Spell checker

3.1 Checking nouns

In Tamil noun is a word, which is capable of taking a case suffix and/or plural suffix and/or a postposition suffix and/or clitic suffixes. A Noun stem can be divided in to two parts as monomorphemic (e.g. '*kal*', '*malar*') and polymorphemic (e.g. '*kalvi*','*kaatchi*'). Monomorphemic words are ones, which cannot be split into meaningful constituents while polymorphemic words are derived or inflected from other root words [4]. Case markers express the syntactic and semantic relation between a noun phrase and a verbal predicate. Most cases in Tamil are realized by case suffixes although some cases are indicated by case suffixes along with postpositions. We have considered ten cases and the list of cases along with their suffixes is given in the table 1.

S.No.	Case	Suffix	S.No.	Case	Suffix
1.	Nominative	ϕ	6.	Sociative	*ooTu, uTan*
2.	Accusative	*Ai*	7.	Locative	*ilirunthu, iTamirunthu*
3.	Dative	*Ku*	8.	Ablative	*In*
4.	Benefactive	*ukkaaka*	9.	Genetive	*athu*
5.	Instrumental	*Aal*			

Table 1. Tamil Case Markers and its corresponding suffixes

Tasks in noun correction are further classified as Case marker, plural checking, post-position checking and adjective checking. The different case endings are extracted from the morphological analyzer like ill, all, itam, udan, aaga, kal, ukku, odu, irunthu, udiadu and adu. After this phase a check is made to ascertain the validity of the last suffix. If the case endings are misspelt the corrections are done to rectify the mistake. The followings are some of the errors in the case endings in nouns.

E.g.: 'marattal' instead of 'marattil'

'avaneedam' instead of 'avanidam'

'avanukkaka' instead of 'avanukkaaka'

'marangkaaL' instead of 'marangkaL'

'marattilaruntu' instead of 'marattiliruntu'

'marattakku' instead of 'marattukku'

Consider the word 'marattilaruntu' for spell checking. This word is sent to the morpho-logical analyzer as input for analysis. Since, the last suffix of the word 'elarnutu' is misspelt, the morphological analyzer is unable to split it. The spelling verification and correction phase corrects the last suffix ('elaruntu') of the word to 'elirunthu' and sends the word with corrected last suffix to morphological analysis for splitting the words. Finally it returns the root word and suffixes to suggestion generation phase.

3.2 Checking verbs

The word that is capable of taking a tense marker is a verb. A verb may be qualified by an adverb or adverbial, it may be conjugated for PNG and it may be changed into adjectival or

adverbial participle. The verbs are divided into two sets. They are finite forms and non-finite forms.

3.2.1 The Finite Form

The finite form of a verb may be Simple or Complex. In finite forms the simple form is divided into three groups according to their structural classes.

1. Verbs that doesn't take any suffixes e.g. sari, tani.

2. Tenseless verbs that takes suffixes but doesn't have PNG markers e.g. illai, untu

3. Verbs that takes suffixes including PNG markers. e.g. cey, sappitu.

The verb stems have to be classified as follows on the basis of the tense markers they take.

(i) Simple Finite Verbs

The verb, which takes tense marker and person number gender marker, is said to be simple finite verb. The tense maker is directly added to the root verb followed by the person number and gender marker. The tense marker for human and non-human is not always same. Also the tense marker for non-human singular and non-human plural is not always same. The person number gender maker has an attachment with the subject. The negative marker 'aa' is also can be attached with the simple verb to indicate the negative form of the verb.

e.g.	cey – kir – aan	(present)
	cey – th – aan	(past)
	cey – v – aan	(future)

(ii) Complex Finite Verbs

The complex verb may consist of verb stem, aspect, voice, modals, tense, person number gender and clitics. Among these, the items that follow the stem may be optional. The complex verb consists of one or more auxiliaries. Normally the one type of auxiliary does not follow the same type of auxiliary.

3.2.2 Non – Finite forms of verb

The non-finite forms of verb can also be classified into two groups namely adjectival participle and adverbial participle. The adjectival participles can be got by adding the tense marker and 'a' with the root verb.

e.g.	pati – tt – a	→	patitta
	pati – kinr – a	→	patikkinra
	pati – um	→	patikkum

The adverbial participle suffixes can be get by adding a 'a' or 'thu' or 'aal' like suffixes.

e.g.	poo – ka	→	pooka	
	pati – kka	→	patikka	(infinitive)

cey – thu	→	ceythu	
paar - ththu	→	paarththu	(conjuntive)
cey – th – aal	→	ceythaal	
pati – thth – aal	→	patiththaal	(conditional)

Verb checking tasks is further classified into subtask as Person, Number & Tense marker checking. The different endings in the verb are extracted from the morphological analyzer result. The tense marker endings include ‘kiru’, ‘kindru’, ‘aaninRu’, ‘p’, ‘v’, ‘th’, ‘t’,’ and ‘R’. All these define the particular verb as present tense or past tense or future tense. The gender markers include ‘an’, ‘aan’, ‘aL’, ‘aaL’, ‘ar’, ‘aar’, ‘thu’, ‘a’, and ‘kaL’. They define the gender as masculine or feminine. The number markers are ‘thu’, ‘vai’, ‘a’, ‘kaL’. They define the word as singular or plural [1]. If the word has been misspelled in any of the above endings of the verb, mistakes have to be corrected and the possible suggestions should be generated.

E.g.:

‘ceikiRarkaL’ instead of ‘ceikiRaarkaL’

‘ceithatha’ instead of ‘ceithathu’

‘ceivan’ instead of ‘ceivaan’

Consider the word ‘thoongkukindRan’. This word is send to the morphological analyzer for analysis. As the word is miss-spelt , the analyzer is unable to split it. The spelling verification and correction phase corrects the last suffix ‘an’ of the word ‘thoongkukindRan’ to ‘aan’ and sends the word with corrected last suffix to the morphological analyzer. Analyzer returns the root word and the suffixes to suggestion generation phase. The suggestion generation phase sends this root word and suffixes returned by the morphological analyzer to the morphological generator to generate suggestion.

3.3 Checking postpositions

A word that does the function of a case suffix is called a postposition. Normally the postposition comes after the case suffix. In some places the case suffix is optional. The postpositions are classified into five, according to the case suffix they take:

- Accusative postpositions
  - *vita, pola, kontu, nokki,* pattri, kuriththu, suttri, vittu, thavira, munnittu, vendi, otti, poruththu, poruththavarai, *etc.*
- Genitive postpositions
  - miitu, meel, vazhiyaaka, mUlamaaka, vaayilaaga, keezh , etc.
- Locative postpositions
  - Irunthu, ulla
- Dative postpositions
  - *aaka, enRu, mun, pin,* endru, uriya, idaiye, maththiyil, ethiril, arukil, pathil, thaguntha, maaraaga, naeraaga *etc.*

If the input word is a postposition, it verifies whether it is correct or not. If the word is incorrect, the correction is done and suggestions are generated.

3.4 Correcting the similar sounding letter

Similar sounding letter can cause a mistake in spelling of word. For example consider the word ‘Thaalam’. The letter ‘La’ misspelled as ‘la’. In Tamil language, three letters have similar sounding letters. Suggestions are generated by examine the possible similar sounding letter for the erroneous word. The suggestions that exist in the dictionary are displayed to the user. Possible Similar sounding letters in Tamil are

la, La, zha

ra, Ra

Na, na, nn

3.5 Correcting the adjacent key errors

While typing the word the user can type a wrong letter instead of the correct one. Hence consideration of adjacent keys of a mistyped letter plays a key role. If any adjacent key of the mistyped letter matches with the original letter replace that letter instead of mistyped one. After that check, the corrected word is checked against dictionary or the case ending rules to generate the possible suggestions. Adjacent key error corrections are based on Tamil ’99 standard keyboard.

Fig 2. Tamil 99 keyboard

For example the word ‘avanutaiya’ can be mistyped as ‘avanutaeya’ or ‘avanutuya’ or ‘avanutooya’. These are all the possible adjacent key errors for the letter ‘tai’. All the possible adjacent keys are replaced and checked against dictionary and error correcting functions.

E.g.: a aa, E, e, and aov.

Ka La, Ra, pa, la, and ng

e a, aa, E, u, aov, and O

pa Ra, na, ka, ma, la, ra

When the correction of errors in root word is completed, the corrected along with associated suffixes are sent to the suggestion generation phase.

3.6 Character positioning rules

A Tamil word should start with any one of the 12 Vowels or 10 Vowel Consonants (‘ka’, ‘sa’, ‘tha’, ‘N’, ‘pa’, ‘ma’, ‘va’, ya’, ‘ng’). Similarly a Tamil word should end with any one of the 12 Vowels or 11 Consonants (‘ng’, ‘Na’, ‘ma’, ‘na’, ‘ya’, ‘ra’, ‘la’, ‘va’, ‘zha’, ‘La’) [2].

If a word does not start with the set, the spell checker will do the possible correction and also get the nearest matching root words from the dictionary. Similarly, if a word does not end with the above-mentioned letter, the spell checker will correct it with the case ending rules.

3.7 Dictionary Look Up

Another important issue is the computerized dictionary which is concerned with the size of the dictionary, the problem of inflection and creative morphology, the dictionary file structure, dictionary partitioning, word access techniques and so on [5].

The Tamil dictionary used here contains all the root words like noun, verb, adjective, adverb and participles. If there is any spelling mistake in root word, the correction of the word has been carried out and all the nearest matching dictionary entries are collected. This informa-tion will be sent to the suggestion generation phase.

4. Suggestion Generation

Spell checker makes use of the morphological generator to generate all possible suggestions. A morphological generator designed for Tamil needs to tackle the different syntactic categories such as nouns, verbs, postpositions, adjectives, adverbs etc. separately, since the addition of morphological constituents to each of these syntactic categories depends on different types of information [7].

While generating the noun derivatives from the roots, linguistic rules determining the form of a plural suffix has to be considered. The attachment of case suffixes to nouns is an important part of the morphological generator for nouns. In addition, this has to take into consideration the fact that certain nouns can take case suffixes only in oblique form. The euphonic suffix sometimes comes along with oblique suffixes or with plural suffixes. During the combination of the root noun with the above-mentioned suffixes, "sandhi rules" have to be taken into account.

The Morphological structure of Tamil verb is quite complex since it caters to person, gender, and number markings and also combines with auxiliaries that indicate aspect, mood, causation, attitude etc. While morphologically generating the verb, the gender, number and person of the subject is necessary in order to select the appropriate suffix catering to the selected tense.

Efficiency of the spell checking process depends on the right suggestion presented as a default suggestion. In such a case, the user needs only to confirm the default suggestion and proceed with the next error. Otherwise, the user needs to scroll through a list of suggestions and pick one as the right one. All the suggestions have been compared against the words in the dictionary. The suggestions that exist in the dictionary are displayed to the user.

5. Conclusion

The Spell checker for Tamil helps the user to identify most of errors, which may occur while typing. The task implemented in Tamil Spell checker are Case marker, postposition check-ing for nouns, Adjective checking for nouns, Case ending and PNG marker checking for verbs, Adverb checking, and Adjacent key errors checking. The applications of the Tamil Spell checker are Word processors, search engines, information filtering and extraction systems, and machine translation systems.

References:

1. Thomas Lehmann (1993) “A Grammar of modern Tamil”, Second Edition, Pondicherry Institute of Linguistics and Culture.
2. Dr. K. Balasubramanian (2001), “Studies in Tholkappiyam”, Annamalai University.
3. A Smart Spell checker system http://www.coe.neu.edu/
4. Anandan. P, Ranjani Parthasarathy, Geetha T.V. (2001), Morphological Analyzer for Tamil*,* ICON 2002, RCILTS-Tamil, Anna University, India.
5. Bidyut Baran Chaudhuri (2000) “Reversed word dictionary and phonetically similar word grouping based spell-checker to Bangla text”, Indian statistical Institute, Kolkata, India.
6. Sengupta and B.B. Chaudhuri *(*1996*), “*Morphological processing of Indian languages for lexical interaction with application to spelling error correction” Sadhana, Vol. 21, Part. 3, pp. 363-380.
7. P. Anandan, T.V. Geetha, and Ranjani Parthasarathi (2001), “Morphological Generaor for Tamil”, Tamil Inaiyam 2001, Malaysia.