proceedings

MDPI

Proceedings

Design and Implementation of NLP-Based Spell Checker for the Tamil Language †

Pawan Kumar *, Abishek Kannan and Nikita Goel

Virtuous Transactional Analytics Pvt. Ltd., Noida 201309, India; abishek.kannan@vitrana.com (A.K.); nikita.goel@vitrana.com (N.G.)

* Correspondence: pawan.gupta@vitrana.com

† Presented at 1st International Electronic Conference on Applied Sciences, 10–30 November 2020; Available online: https://asec2020.sciforum.net/.

Published: 10 November 2020

Abstract: A spell checker is a tool used for analyzing and validating spelling mistakes in the text. Recently, the role of a spell checker has diversified, and it is also used to suggest possible corrections to the detected spelling mistakes. Tamil is one of the oldest surviving and international spoken languages of the world, and it is grammatically very rich. Grammar is vital for effective communication and information transmission. However, learning the language rules and the old teaching methodology becomes a challenge for the researchers. The amalgamation of computer and language using natural language processing (NLP) provides a solution to this problem. In this paper, an advanced NLP technique is used to detect wrongly spelled words in the Tamil language text, and to provide possible correct word suggestions and the probability of occurrence of each word in the corpus. The proposed model recommends correct suggestions for the misspelled words using the minimum edit distance (MED) algorithm, which is customized for the Tamil vocabulary. A distance matrix is created between the misspelled word and all possible permutations of the word. Dynamic programming is used for calculating the least possible changes needed to correct the misspelled words, and suggesting the most appropriate words as the corrections.

Keywords: automatic spelling suggestion; dynamic programming; minimum edit distance (MED); natural language processing (NLP); Tamil spell checker

1. Introduction

Tamil (தமிழ்) is a Dravidian language largely-spoken by the residents of South India and parts of North India. It is one of the longest surviving traditional languages and is also widely spoken in Sri Lanka, Malaysia, and Singapore. Tamil has 247 letters comprised of 12 vowels, 18 consonants, 216 composite letters, and one special letter, 'ஃ' known as "ayutha eluththu" [1]. In Tamil, the nouns are categorized as "rational" and "irrational". The humans and demiurges are grouped as rational while the rest are grouped as irrational.

In the current internet era, high-quality content is an important asset. The content quality is mainly decided by the typos, misspelled words, and grammatical mistakes. Fabricating error-free content adds a professional touch to the work. Therefore, spell checkers in the text processors comes to aid. Error detection and correction are basic needs for any text processing software or tool. Misspelled words are classified into two groups, namely non-word errors and real-word errors. The non-word errors are either not valid or not present in the lexicon.

A considerable amount of work has already been done on the foreign and Indian languages, but quite a few on the Tamil language. In [2], a sequence clustering algorithm was reported to check a word in the dictionary. If the word was not found, then possible suggestions for the misspelled word

were generated through the n-gram technique. In [3], a spell checker was proposed to validate the text and minimum edit distance (MED) to generate possible suggestions. But, this approach showed limited functionality, and if the word was not found in the lexicon then its validity cannot be predicted. The authors in [4] had used unison and MED to find valid or invalid words, and n-gram and bigram models to provide suitable suggestions. In [5], the forward and reverse finite automata were used to identify the text errors, and MED and n-gram technique for possible suggestions. In [6], the authors had presented an optical character recognition (OCR) and morphological analyzers for error identification, and used MED and the bigram language model for potential suggestions. In [7], the bigram probabilistic model was reported for suggesting words in the subject of the sentence. The model was trained using a 3 GB volume of Tamil text. An approach that splits the Tamil words morphologically and checks for error using the Tamil grammar rules was reported in [8]. A system with n-gram, MED, and frequency of words was reported in [9], where appropriate recommendations were proposed for the wrongly spelled words. Hashing techniques were used to refine the processing speed for spell checking and word recommendations. The approach was trained with a dictionary of 4 million Tamil words. In [10], a Tamil spell checker web application was proposed, which was used for finding spelling mistakes and recommending appropriate alternatives. But, this system can process limited words at a time. The system presented in [11] performed real-time spell checking and provided relevant suggestions for the misspelled words. This system takes a sentence as an input, tokenizes it, locates misspelled words, recommends suggestions, and use the n-gram technique to rank and return the best corrections. However, the morphologically rich essence of the Tamil makes it challenging for the spell checkers to validate the text.

In this paper, an approach that morphologically identifies spelling mistakes in the Tamil sentences, and recommends correct word suggestions for the misspelled words, is proposed. The model detects misspelled words by checking their presence in the corpus, and uses MED to form a distance matrix, which helps in identifying the most likely suggestions. The proposed spell checker could be useful for various applications such as machine translation systems, information extraction, filtering systems, and search engines.

2. Materials and Methods

The features used in the proposed model are highlighted below, which will give a detailed understanding of its working and architectural flow.

2.1. Dataset

There is no specific dedicated dataset for evaluating Tamil spell checkers. Researchers and scholars working on the Tamil language usually use data from various sources like Wikipedia, Tamil articles, short stories, newspapers, and online websites. The dataset used in this work is prepared from the commonly used Tamil words (source Wikipedia) and the corpus of the Tamil article [12]. The data was also pre-processed and corrected grammatically.

2.2. Methodology

2.2.1. Data Pre-Processing

Real-world data are often incomplete, inconsistent, inaccurate, and lack specifically required trend. Therefore, data pre-processing is a primary and most significant step in natural language processing (NLP). It is a crucial process as it directly affects the success rate of the model. The steps involved in data pre-processing are tokenization, stop word removal, stemming, and lemmatization.

In the proposed model, the text is tokenized into different words, and string manipulation operations are performed. Further, each word is checked in the vocabulary corpus of the Tamil dictionary. In this process, a sentence is split into chunks of words, and string manipulation operations are performed on them for the formation of all possible word combinations. The words not found in the vocabulary corpus are categorized as misspelled, and further processing is performed on them.

2.2.2. Minimum Edit Distance (MED)

MED is applied to each word of the Tamil vocabulary corpus to detect misspelled words. The MED is calculated word wise, where a matrix is formed to calculate the number of operations required to correct the misspelled word present in the corpus, iteratively. The operations, used for calculating the MED, are divided into three categories. The first type is insert operation, which is given weight as 1, where an alphabet is added at a certain position in the string. The second type is the delete operation, which is also given weight as 1, where an alphabet is removed from a certain position. This step changes the misspelled word to the same as the word present in the corpus. The third type is the combination of delete and insert operations, and it is called replace operation. Since it involves both operations, hence its weight is given as 2. Here, if an alphabet is removed from a certain position of the word, a new alphabet is added to the same position to replace the old alphabet. The MED technique is used to estimate the equivalence of two words–the lesser the computed cost, the higher the equivalence. For example, the distance between "வாள" and "வா" is 1 as it requires one deletion operation "ள". Likewise, the edit distance between the incorrectly spelled word "வணக்கம்ம்" and the correct word "வணக்கம்" is 2, where the '◌்' is replaced by the 'ம'.

2.2.3. Matrix Formation Algorithm

Distance matrices are used to envision predictive analytics, like the accuracy and precision of the model. A distance matrix is formed concerning the misspelled word (source) and the possible word suggestions (target word). The matrix is used to calculate the cost of operations needed to be performed to achieve the target word from the source word. In the proposed model, the misspelled words, after data pre-processing, are compared with the most likely words as per the lexical analysis. The matrix illustrates the alphabetical segmentation of both the words and shows the weight required for each edit as depicted in Table 1.

Table 1. Alphabetical segmentation of the words வணக்கம் and வணக்கம்ம்.

	#	வ	ண	க	◌்	க	ம	◌்
#	0	1	2	3	4	5	6	7
வ	1	0	1	2	3	4	5	6
ண	2	1	0	1	2	3	4	5
க	3	2	1	0	1	2	3	4
◌்	4	3	2	1	0	1	2	3
க	5	4	3	2	1	0	1	2
ம	6	5	4	3	2	1	0	1
◌்	7	6	5	4	3	2	1	0
ம	8	7	6	5	4	3	2	1
◌்	9	8	7	6	5	4	3	2

2.2.4. Spelling Suggestions

After detecting the misspelled words, the proposed model recommends a list of appropriate suggestions. Implementing the cost calculated from the distance matrix, the word suggestions are ranked. Suggested word with the least cost is given the highest ranking. The words with the smallest edit distance, with the words of the corpus, are the most promising suggestions. Spelling suggestions are given for the words, which match with the highest probabilistic words displayed on the top of the list of the Tamil vocabulary.

Figure 1 shows an architecture of the proposed model, where an input sentence with a misspelled word is passed through it. The sentence is tokenized during the data pre-processing step, and a matrix is formed to find the most probabilistic correct word. A probabilistic approximation is calculated, after performing MED operation of the potential misspelled word with each word in the Tamil word corpus, using the formula (1). The top probabilities represent the best matching word for the misspelled word.

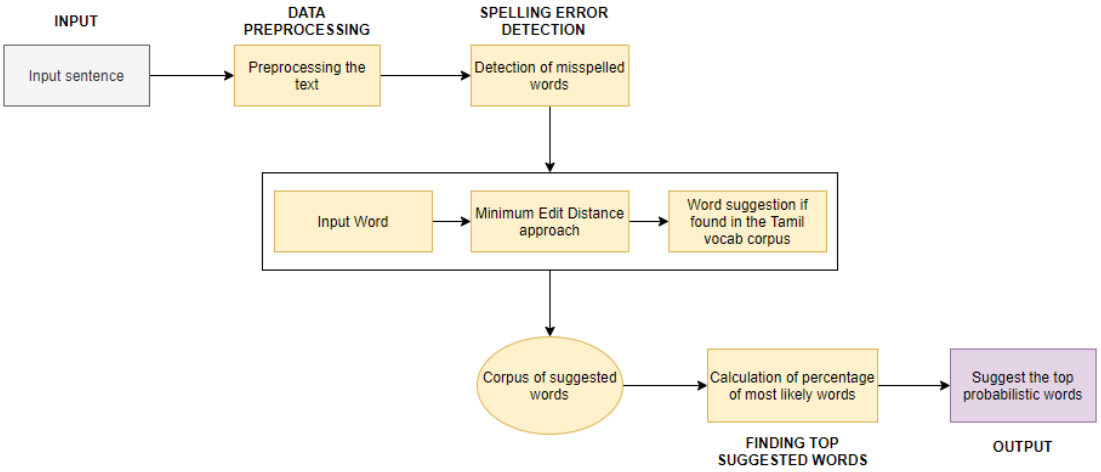

Figure 1. Workflow of the spell checker.

3. Results

The code of the proposed NLP-based spell checker is implemented using Python 3.6 programming language, and it is executed on the Linux environment with Tesla P100-PCIE-16GB GPU. The proposed model detects the incorrect word, and predicts the correct alternatives of the word with their probability of occurrence. The correct words predicted by the model are found to be lexically similar to the misspelled word. The probability of the correct alternatives is calculated using the following expression

$$P(w) = \frac{C(w)}{T(w)} \quad (1)$$

where $P(w)$ is the probability of the recommended word, $C(w)$ is the count of the word in the corpus, and $T(w)$ is the total count of words in the corpus. The spell checker model is configured to predict a maximum of four suggestions with their highest probability values. The following examples show the output of the spell checker model.

Example 1: Input word: வாள *Results of the spell checker model*: word 1: வா, probability 0.000310 word 2: வாள், probability 0.000219 word 3: வாளி, probability 0.000123	*Example 2*: Input word: பல *Results of the spell checker model*: word 1: பல, probability 0.001888 word 2: புலி, probability 0.001056 word 3: புலம், probability 0.000529 word 4: புளி, probability 0.000321
Example 3: Input word: அர *Results of the spell checker model*: word 1: அற, probability 0.000760 word 2: அரை, probability 0.000678 word 3: அறை, probability 0.000098	*Example 4*: Input word: தட *Results of the spell checker model*: word 1: தடை, probability 0.001260 word 2: வடை, probability 0.000576 word 3: படை, probability 0.000487

The minimum edits required to convert the incorrect word string to the correct word string can be seen from the distance matrix constructed using dynamic programming. The following mathematical expression is used for generating a distance matrix (*Mat*) between the misspelled word and the correct word

$$Mat\ [0, 0] = 0 \quad (2)$$

$$Mat\ [i, 0] = Mat\ [i - 1, 0] + deletion\_cost\ (\text{source}[i]) \quad (3)$$

$$Mat\ [0, j] = Mat\ [0, j-1] + insertion\_cost\ (\text{target}[j]) \quad (4)$$

where "*i*" is the row (source) number, i.e., the index of the misspelled word, and "*j*" is the column (target) number, i.e., the index of the correct word. The matrix (5) is formed as per each cell operations performed on the expressions (2)–(4).

$$Mat[i,j] = min\begin{cases} Mat[i-1,j] + deletion\_cost \\ Mat[i,j-1] + insertion\_cost \\ Mat[i-1,j-1] + \begin{cases} replacement\_cost \text{ if source}[i] \neq \text{target}[j] \\ 0 \text{ if source}[i] = \text{target}[j] \end{cases} \end{cases} \quad (5)$$

Here, the *deletion_cost* and *insertion_cost* are 1, and *replacement_cost* is 2. For the input misspelled word "வாள", one edit is required (deletion of "ள" with the deletion cost of 1) to form the correct word suggestion "வா" as can be seen from the distance matrix, shown in Table 2.

Table 2. Distance matrix.

	#	வ	◌ா
#	0	1	2
வ	1	0	1
◌ா	2	1	0
ள	3	2	1

Similarly, for the misspelled word "அர", one edit is required (replacement of "ர" with "ற" with the replacement cost of 2) to give the correct suggested word "அற", shown in Table 3. However, in a few cases, the spell checker model does not provide the correct word suggestion, and it can be further improved by using a larger training dataset.

Table 3. Distance matrix.

	#	அ	ற
#	0	1	2
அ	1	0	1
ர	2	1	2

4. Conclusions

In this paper, an NLP-based spell checker is proposed for detecting spelling mistakes in the words of the Tamil language. The proposed model not only detects the wrongly spelled words but also predicts the possible suggestions of the correct words that the user might want to write. The proposed spell checker finds its application in various fields such as detecting typo errors, machine translation systems, information extraction, filtering systems, and search engines. A lot of work can be performed in this direction as per the availability of more content in the Tamil language. As the size of the corpus will increase, more suggestions can be given for a particular word. However, there is no benchmark dataset for the Tamil language as it is for other languages. This is one of the reasons for inefficient spell checkers in Tamil as there is no proper dataset to test and validate the accuracy of the system. The proposed model is tested on the data collected from Tamil articles, short stories, and newspaper. This helped in incorporating commonly used words in the vocabulary corpus to make more accurate detection and correction of the misspelled words.

Author Contributions: Conceptualization, P.K., A.K. and N.G.; methodology, P.K. and A.K.; software, P.K.; validation, A.K. and N.G.; formal analysis, P.K.; investigation, A.K. and N.G.; resources, P.K.; data curation, P.K., A.K. and N.G.; writing—original draft preparation, P.K., A.K. and N.G.; writing—review and editing, P.K.; visualization, A.K. and N.G.; supervision, P.K.; project administration, P.K., A.K. and N.G.; funding acquisition, P.K, A.K. and N.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sakuntharaj, R.; Mahesan, S. A novel hybrid approach to detect and correct spelling in Tamil text. In Proceedings of the 2016 IEEE International Conference on Information and Automation for Sustainability: Interoperable Sustainable Smart Systems for Next Generation, ICIAfS 2016, Galle, Sri Lanka, 16–19 December 2016; pp. 1–6.
2. Indumathi, J.; Anish, A. Sequence clustering algorithm for spell checking and spell suggestion in Tamil language. In Proceedings of the 2014 Tamil Internet Conference, Pondicherry, India, 19–21 September 2014; pp. 1–6. Available online: http://uttamam.org/papers/14_41.pdf (accessed on 6 November 2020).
3. Pirinen, T.A.; Lindén, K. Finite-state spell-checking with weighted language and error models—Building and evaluating spell-checkers with Wikipedia as corpus. In Proceedings of the Seventh SaLTMiL Workshop on Creation and Use of Basic Lexical Resources for Less-Resourced Languages, Valletta, Malta, 23 May 2010; pp. 13–18.
4. Segar, J.; Sarveswaran, K. Contextual spell checking for Tamil Language. In Proceedings of the 2015 Tamil Internet Conference, Singapore, 30 May 2015; pp. 379–383.
5. Ram, R.V.S.; Mouli, N.C.; Bhuvaneswari, P.; Priya, J.A.; Shanmugam, B.K. Hybrid approach for developing a Tamil spell checker. In Proceedings of the International Conference on Natural Language Processing, ICON 2005, Kanpur, India, 18–20 December 2005; pp. 111–115.
6. Sridhar, R.; Rathi, L.R.; Rithya, P.; Nivrutha, P. Use of Tamil grammar rules for correcting errors in optical character recognised document. In Proceedings of the 2013 Tamil Internet Conference, Kuala Lumpur, Malaysia, 24 March, 2013; pp. 165–173.
7. Sakuntharaj, R.; Mahesan, S. Detecting and correcting real-word errors in Tamil sentences. *Ruhuna J. Sci.* **2018**, *9*, 150–159.
8. Thendral, S.; Subhashini, R.; Karky, V.M. Tamil spell checker app for iPhone. *Indian J. Sci. Technol.* **2019**, *12*, 1–4.
9. Uthayamoorthy, K.; Kanthasamy, K.; Senthaalan, T.; Sarveswaran, K.; Dias, G. DDSpell—A data driven spell checker and suggestion generator for the Tamil language. In Proceedings of the 19th International Conference on Advances in ICT for Emerging Regions, ICTer 2019, Colombo, Sri Lanka, 2–5 September 2019; pp. 1–6.
10. Tamil spell Checker—Vaani (வாணி). Available online: http://vaani.neechalkaran.com (accessed on 27 May 2019).
11. Gupta, P. A context-sensitive real-time spell checker with language adaptability. In Proceedings of the 14th International Conference on Semantic Computing, ICSC 2020, San Diego, California, USA, 3–5 February 2020; pp. 116–122.
12. Kaggle.com. Available online: https://www.kaggle.com/praveengovi/tamil-language-corpus-for-nlp (accessed on 6 November 2020).