Hindi Spell Checker

Amit Sharma, Pulkit Jain
Dr. Amitabha Mukherjee

Motivation

- ❑ Spell checking tools are important for editors, search engines etc.
- ❑ A lot of text is typed in Hindi
  - Books
  - Novels
  - Newspapers
  - Magazines
- ❑ Many spell checking tools exist for English, but not many for Hindi

Introduction

- ❑ Error Detection
  - Non Word Errors

    बस्तु for वस्तु
    दांत for दाँत
  - Real Words Errors
    - दुकान उस और है for दुकान उस ओर है
- ❑ Error Correction
  - Generate Candidate corrections
    - P(c|w) denotes the probability that c is correction for w
    - Find a correction c for word w such that

      P(c|w) is maximized

      P(c|w) = P(w|c) P(c) / P(w)
  - Rank candidates
    - Damerau Levenshtein distance
    - Word Frequency
    - Similarity Measures

Previous Work

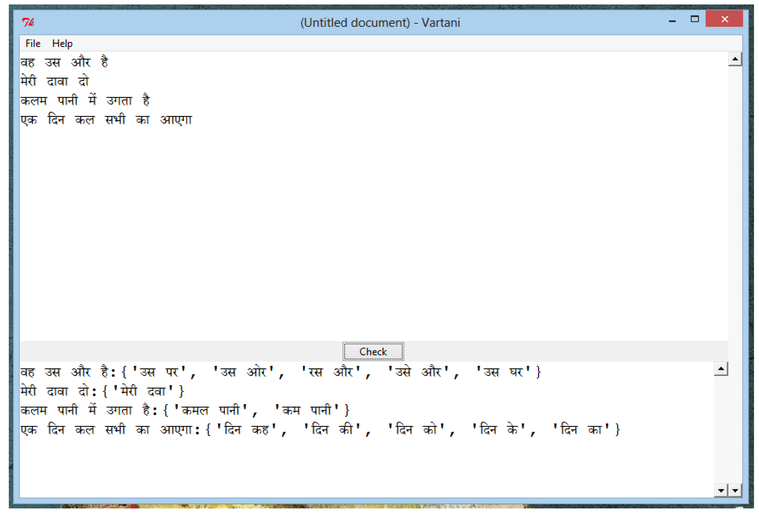
- ❑ Lexicon based lookup methods for error identification
- ❑ N - Gram analysis
- ❑ Context Aware Spell Checking
  - Finite State Automaton for spell checking [1]
  - Co Occurrence Graphs for spell checking [2]
  - Trigram Probabilities for building context [3]

Our Approach

- ❑ We aim to build a context sensitive spell checker
- ❑ Use of dictionary lookup for non word error identification
- ❑ Combine edit distance and word frequency for ranking.
- ❑ Use of N- Gram probabilities for context based checking
  - Use 2-grams, 3-grams and 4-grams
  - Identify erroneous grams – grams with low probabilities
  - Generate candidate grams
  - Rank candidate grams

Results

- ❑ For evaluation of non-word error detection and ranking, a set of 291 commonly misspelled hindi words, along with intended words were collected. A lexicon was trained from hindi corpus. Words that appeared less than 20 times were removed.
- ❑ 69.1 % of the intended words were found in top 10 ranked candidates
- ❑ 12.02 % were misclassified as correct and 6.18% were ranked below the top 10 and of 12.7% words were actually not present in the database



Future Work

We limited ourselves to use of 3-Grams and 2-Grams for context based spell checking, due to the noisy nature of corpus we had. Availability of a larger and richer corpus in terms of number of correct words will be helpful in improving the results.

A more sophisticated GUI which does real time spell checking while typing can be implemented for better user experience.

Contact

Pulkit Jain
IIT Kanpur, Kanpur. 208016
Email: pulkitj@iitk.ac.in

Amit Sharma
IIT Kanpur, Kanpur. 208016
Email: amits@iitk.ac.in

CS365 : Artificial Intelligence

References

[1] Tommi Pirinen and Krister Linden. ***Finite-state spell-checking with weighted language and error models.*** Proceedings of LREC 2010 Workshop on Creation and use of basic lexical resources for less-resourced languages [2010]

[2] Francesco Bonchi, Ophir Frieder, Franco Maria Nardini, Fabrizio Silvestri and Hossein Vahabi. ***Interactive and Context-Aware Tag Spell Check and Correction*** [2012]

[3] Suzan Verberne. ***Context-sensitive spell checking based on word trigram probabilities*** [2002]

[4] Neha Gupta, Pratistha Mathur. ***Spell Checking Techniques In NLP: A Survey*** [2012]

[5] Peter Norvig. How to write a spelling corrector. http://norvig.com/spell-correct.html