Maximum Matching Word Segmentation Algorithm

MaxMatch Segmentation is a word segmentation algorithm that breaks down a sentence into a list of words by selecting the longest possible word in the dictionary at each step.

Given a wordlist and a string.

1) Start a pointer at the beginning of the string
2) Find the longest word in dictionary that matches the string starting at pointer
3) Move the pointer over the word in string
4) Go to 2

Max-match segmentation illustration

Thecatinthehat	the cat in the hat
Thetabledownthere	the table down there
	theta bled own there

Doesn’t generally work in English!

But works astonishingly well in Chinese

莎拉波娃现在居住在美国东南部的佛罗里达。

莎拉波娃 现在 居住 在 美国 东南部 的 佛罗里达

Modern probabilistic segmentation algorithms even better

- **Greedy Approach**: The algorithm takes a greedy approach by always choosing the longest match first. This approach might not always yield the optimal segmentation, but it is efficient.
- **Dictionary Dependency**: The success of the MaxMatch algorithm heavily depends on the quality and comprehensiveness of the dictionary being used. If the dictionary does not contain all possible valid words, the algorithm might fail to segment correctly.
- **No Backtracking or Hierarchical Search**: MaxMatch does **not** create a hierarchy of all possible words. It does not consider all possible segmentations or word combinations. Instead, it greedily selects the longest possible word match at each step and moves on. This means that once it selects a word, it does not backtrack to try other possibilities.

Basic Text Processing

Sentence Segmentation

Sentence Segmentation

!, ? are relatively unambiguous

Period "." is quite ambiguous

- Sentence boundary
- Abbreviations like Inc. or Dr.
- Numbers like .02% or 4.3

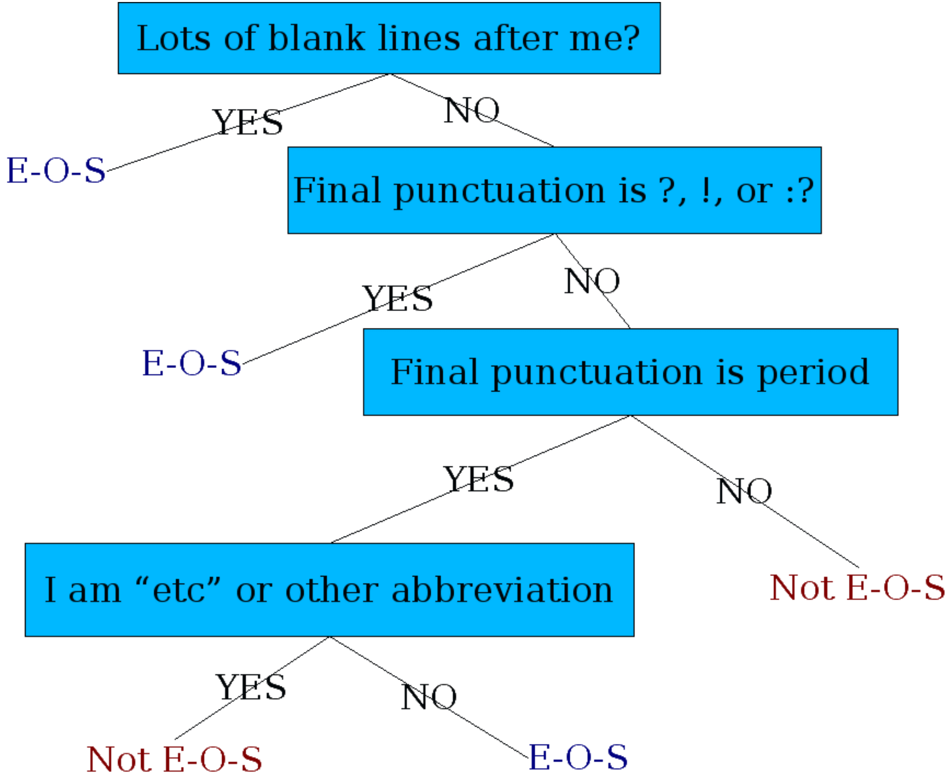
Build a binary classifier

- Looks at a "."
- Decides EndOfSentence/NotEndOfSentence
- Classifiers: hand-written rules, regular expressions, or machine-learning

Sentence Segmentation

- **Punctuation-based**: This method uses the presence of punctuation marks, such as periods, exclamation marks, and question marks, to identify sentence boundaries.
- **Statistical:** are based on machine learning, and consider a variety of features, such as punctuation marks, capitalization, and word frequency, to make predictions about sentence boundaries.
  - Trained on a large annotated corpus where the sentence boundaries are already marked. The model learns patterns from this training data.
- **Hybrid:** This method combines the punctuation-based and statistical methods to provide improved performance and accuracy.

Determining if a word is end-of-sentence: a Decision Tree


Lots of blank lines after me?
YES
NO
E-O-S
Final punctuation is ?, !, or :?
YES
NO
E-O-S
Final punctuation is period
YES
NO
I am “etc” or other abbreviation
Not E-O-S
YES
NO
Not E-O-S
E-O-S