Language Model

N-Gram Language Model and Spell Error Correction

What is Model?

Model Example

- Regression Model
- Simulation Model
- Business Model
- ...

Why we need model?

- A system that can ...
- **Understand**, **define**, **quantify**, **visualize**, or **simulate** the world by referencing to **existing** and usually **commonly accepted knowledge**.

Language Model

- Understand human language

- Much of Natural Language Processing can be structured as (conditional) language modelling:

N-grams

- Unigram
  - "An", "apple", "a", "day", "keeps", "doctor", "away", "."
- Bigram
  - "An apple", "apple a", "a day", ...
- Trigram
  - "An apple a", "apple a day", "a day keeps", ...
- Fourgram, Fivegram, ...

N-gram Language Model

- Assign a probability to any word sequences
  - $p(w_1, w_2, ...)$
- Can be used for....
  - OCR
  - Speech recognition
  - Predicting sequence
  - ...

N-gram LM application

Thus we can compare different orderings of words (e.g. Translation):

p(he likes apples) > p(apples likes he)

or choice of words (e.g. Speech Recognition):

p(he likes apples) > p(he licks apples)

Example – Filling in Articles and Prepositions

- In Alan Meryers (2005) Gateways to Academic Writing pp. 277, learners are asked to fill articles and prepositions in the blanks.

The model T Ford was a fragile-looking the automobile, but it became **__the__** most popular car in history. Henry Ford sold 16 million Model Ts _____ the years 1908 and 1928. The Model T is _____ immediate best-seller, not only because of its low price, but because it was _____ powerful car.

Example – using articles and prepositions correctly

- In Gateways to Academic Writing (Meyers 2005, pp. 277), learners are asked to fill articles and prepositions in the blanks.

The model T Ford was a fragile-looking the automobile, but it became **the** most popular car in history. Henry Ford sold 16 million Model Ts **in** the years 1908 and 1928. The Model T is **an** immediate best-seller, not only because of its low price, but because it was **a** powerful car.

Assign Probability Intuitively

- Word count in Lab 1
- Unigram
  - $p(w) = count(w) / N$
- Bigram
  - $p(w_2|w_1) \sim= count(w_1, w_2) / count(w_1)$
- ....

But...

- Pure word count is not enough
  - Small decimal number multiplication (may be out of computer precision)
  - Computing probability of unseen word ($p(w_1|w_0) = count(w_1) / 0$, divided by zero)
- Many other problem worth exploring

Smoothing

- Laplace Smoothing
- Add 1 to every ngrams and renormalize

  P = (count + 1) / N + V

  V: Vocabulary Size

- More smoothing methods

  http://cpmarkchang.logdown.com/posts/190999-equations-for-nlp-ngram-smoothing

Extended Reading

- Oxford Deep NLP course 2017

- https://github.com/oxford-cs-deepnlp-2017/lectures

- Lecture 3 & 4

Extended Reading

- NAIST NLP Programming Tutorial
- http://www.phontron.com/slides/nlp-programming-en-01-unigramlm.pdf
- http://www.phontron.com/slides/nlp-programming-en-02-bigramlm.pdf