Natural Language Processing

Paper Code: CS-821/6

UG 8th Sem

Dr. Samit Biswas, *Assistant Professor*

Department of Computer Sc. and Technology

Indian Institute of Engineering Science and Technology, Shibpur

Email: samit@cs.iiests.ac.in

Plan for Today

- Probabilistic Language Models

Probabilistic Language Modeling

- **Goal:** compute the probability of a sentence or sequence of words. **P(words)** is the joint probability that a sequence of ***words***$=w_1w_2...w_n$ is likely for a specified natural language.

- **Related task:** probability of an upcoming word

  $P(w_5|w_1, w_2, w_3, w_4)$

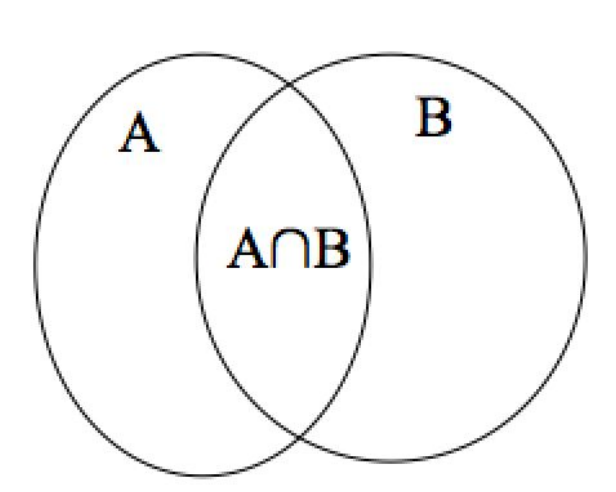
- A model that computes either of these:

  $$\mathbf{P(W)} \text{ or } \mathbf{P(w_n|w_1, w_2...w_{n-1})}$$

  is called a language model.

- **Probability Function:**
    - P(A) means that how likely the event A happens.
    - P(A) is a number between 0 and 1.
    - P(A) = 1 is a certain event.
    - P(A) = 0 is an impossible event.
- **Unconditional Probability** or Prior Probability:
    - P(A): the probability of the event A does not depend on other events.
- **Conditional Probability** – Posterior Probability – Likelihood:
    - P(A|B): this is read as the probability of A given that we know B.
- **Example:**
    - P(put) is the probability of to see the word put in a text.
    - P(on|put) is the probability of to see the word on after seeing the word put.

Unconditional and Conditional Probability Probability



- $P(A|B) = P(A \cap B)/P(B)$
- $P(B|A) = P(A \cap B)/P(A)$

Bays' Theorem

- Bayes' theorem is used to calculate P(A|B) from given P(B|A).
- We know that

$$P(A|B) = P(A \cap B)/P(B)$$
$$P(B|A) = P(A \cap B)/P(A)$$

- So, we will have

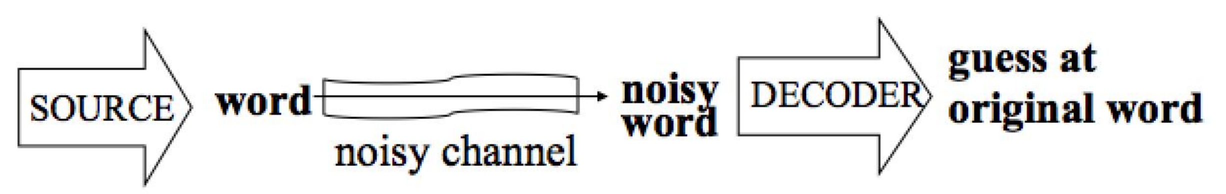
$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$
$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

Language Models

Language Models:

- The Noisy Channel Model
- N-GRAM models are the language models which are widely used in NLP domain.

The Noisy Channel Model



Many problems in Natural Language Processing can be viewed as noisy channel model.

- optical character recognition.
- spelling correction.
- speech recognition.

The Noisy Channel Model

Chain Rule

- The probability of a word sequence $\boldsymbol{w_1, w_2, \ldots w_n}$ is:
  $$\boldsymbol{P(w_1, w_2, \ldots w_n)}$$
- We can use the chain rule of the probability to decompose this probability:
  $$P(w_1^n) = P(w_1)P(w_2|w_1)P(w_3|w_1^2)\ldots P(w_n|w_1^{n-1})$$
  $$= \Pi_{k=1}^{n} P(w_k|w_1^{k-1})$$
- **Example**
  P(the man from jupiter) = P(the)P(man | the)P(from | the man)P(jupiter | the man from)

N Grams

N Grams

- To collect statistics to compute the functions in the following forms is difficult (sometimes impossible):

$$P(w_n|w_1^{n-1})$$

- Here we are trying to compute the probability of w_n after seeing w_{n-1}.
- We may approximate this computation just looking N previous words:

$$P(w_n|w_1^{n-1}) \approx P(w_n|w_{n-N+1}^{n-1})$$

- So, a N-GRAM model:

$$P(w_1^n) \approx \Pi_{k=1}^{n} P(w_k|w_{k-N+1}^{k-1})$$

N Grams ...

N Grams ...

- **Unigrams:** $P(w_1^n) \approx \Pi_{k=1}^{n} P(w_k)$
- **Bigrams:** $P(w_1^n) \approx \Pi_{k=1}^{n} P(w_k | w_{k-1})$
- **Trigrams:** $P(w_1^n) \approx \Pi_{k=1}^{n} P(w_k | w_{k-1} w_{k-2})$
- **Quadgrams:** $P(w_1^n) \approx \Pi_{k=1}^{n} P(w_k | w_{k-1} w_{k-2} w_{k-3})$

N Grams Example

- **Unigrams:**

$$P(the\ man\ from\ jupiter)$$
$$\approx P(the)P(man)P(from)P(jupiter)$$

- **Bigrams:**

$$P(the\ man\ from\ jupiter)$$
$$\approx P(the| < s >)P(man|the)P(from|man)P(jupiter|from)$$

- **Trigrams:**

$$P(the\ man\ from\ jupiter)$$
$$\approx P(the| < s >< s >)P(man| < s > the)P(from|the\ man)P(jupiter|man\ from)$$

Simple Markov Models

- The assumption that the probability of a word depends only on the previous word is called **Markov assumption.**
- **Markov models** are the class of probabilistic models that assume that we can predict the probability of some future unit without looking too far into the past.
- A **bigram** is called a first-order **Markov model** (because it looks one token into the past);
- A **trigram** is called a second-order **Markov model**;
- In general a **N-Gram** is called a **N-1 order Markov model**.

Estimating N Gram Probabilities

- Estimating **Bi-Gram Probabilities**

$$P(w_n|w_{n-1})$$

$$= \frac{C(w_{n-1}w_n)}{\sum_w C(w_{n-1}w)}$$

$$= \frac{C(w_{n-1}w_n)}{C(w_{n-1})}$$

- Here C is the count of that pattern in the corpus.
- Estimating **N-Gram Probabilities**

$$P(w_n|w_{n-N+1}^{n-1})$$

$$= \frac{C(w_{n-N+1}^{n-1}w_n)}{C(w_{n-1}^{n-N+1})}$$

Example: Estimating N gram Probabilities

- Consider a mini corpus of three sentences:
  <s>I am Sam</s>
  <s>Sam I am</s>
  <s>i do not like green eggs and ham</s>

- A few **bigram probabilities** from this corpus :

$$P(I|<s>) = \tfrac{2}{3} = 0.67 \qquad P(Sam|<s>) = \tfrac{1}{3} = 0.33$$
$$P(am|I) = \tfrac{2}{3} = 0.67 \qquad P(<s>|Sam) = \tfrac{1}{2} = 0.5$$
$$P(Sam|am) = \tfrac{1}{2} = 0.5 \qquad P(do|I) = \tfrac{1}{3} = 0.33$$

Which N Gram?

- Which N gram should be used a language Model ?
    - Unigram, Bigram,Trigram, . . .
- Bigger N, the model will be more accurate.
    - But we may not get good estimates for N-Gram probabilities.
- The N-Gram tables will be more sparse.
- Smaller N, the model will be less accurate.
    - But we may get better estimates for N-Gram probabilities.
        - The N-Gram table will be less sparse.
- In reality, we do not use higher than **Trigram (not more than Bigram).**
- **How big are N-Gram tables with 10,000 words?**
    - Unigram – 10,000
    - Bigram – 10000*10000 = 100,000,000
    - Trigram – 10000*10000*10000 = 1,000,000,000,000
-

References

References

- Daniel Jurafsky & James H. Martin, "*Speech and Language Processing*", Prentice Hall
- Diana Maynard, Kalina Bontcheva, Isabelle Augenstein, "*Natural Language Processing for the Semantic Web*", A Publication in the Morgan & Claypool Publishers series.
- Steven Bird, Ewan Klein and Edward Loper, "*Natural Language Processing with Python*", By O'Reilly.
- Christopher Manning and Hinrich Schtze, "*Foundations of Statistical Natural Language Processing*", The MIT Press.

Thank You

Contacts:

Dr. Samit Biswas
Department of CST,
IIEST, Shibpur

samit@cs.iiests.ac.in
https://www.iiests.ac.in/IIEST/Faculty/cs-samit