#### Lecture 21

Advanced smoothing techniques (N-gram models)

# N-gram probability calculation

eg. Of next word prediction (bigram model): this is ...?...

Probability of the Nth word given the previous N-1 words

(bigram: N=2) 
$$P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)}{C(w_{n-1})}$$

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

max probability→winner among all candidate words=next word

 Probability of the whole sentence containing n words (eg. for bigrams)

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

Dr. Seba Susan Delhi Technological University

### OOV (Out of Vocabulary) words and why do you require smoothing?

Formula 1 (prob of next word)

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

Formula 2 (prob of whole sentence)

- 0/0 (avoid divide by zero)
- Count of denom in non zero
- Count of num=0 (prob in Formula
- Even if 1 prob in chain is 0 the who sent. prob=0

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

- wk-2wk-1wk (trigram)
- This is...wk...

## Laplace Smoothing (recap)

- Laplace smoothing (of last slide probability formula)
- Also called add-one smoothing
- To prevent 0/0 situation
- Add 1 to numerator and V to the denominator

$$p_i^* = \frac{c_i + 1}{N + V}$$

Let c\* be the modified count in the numerator

V=size of the vocabulary = number of unique unigram (denominator) i the training corpus

### Class Assignment (deadline 5 pm on 21/10/20)

input≯prediction of the next word≯prob(sentence) [LM: Languag Training corpus→build a probabilistic model (N-gram)→Test modelling

Training corpus:

N=2 (bigram) V=13

<s>the start of the day was good</s><s> if the start is good the who day is good</s><s> the goodness of the day matters</s>

Text prediction (LM):

# Advanced smoothing techniques

## 1. Good-Turing discounting

- Proposed by Good (1953)
- Ques: How to handle OOV (Out of Vocabulary) words whose count the training corpus is zero?
- Let Nc be the number of N-grams that occur c times (in the training corpus)
- (EXCEPT NO)

- $c^*=\overline{(c+1)}\frac{N_{c+1}}{N_c}$
- N2=6 N1=14 N0=0 N=20
- For N0 case (OOV), c\*=N<sub>1</sub>/N=14/20

### 2. Knesser-Ney smoothing

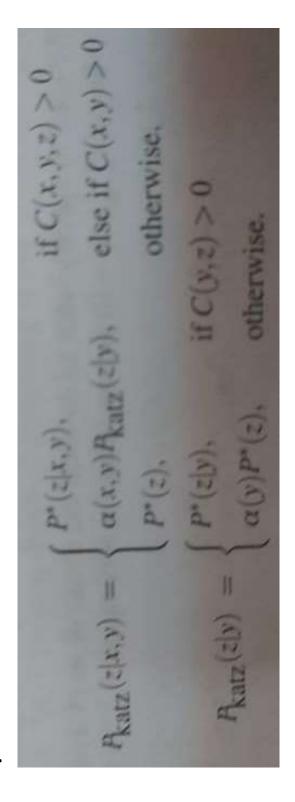
$$c_{N}(w_{i}|w_{i-1}) = \begin{cases} \frac{C(w_{i-1}w_{i}) - \mathbf{D}}{C(w_{i-1})}, & \text{if } C(w_{i-1}w_{i}) > 0\\ \frac{C(w_{i-1}w_{i}) - \mathbf{D}}{C(w_{i-1})}, & \text{otherwise.} \end{cases}$$

#### 3. Interpolation

$$P_{\text{base}}(w_i|w_{i-n+1}^{i-1}) = \lambda_n P_{\text{ML}}(w_i|w_{i-n+1}^{i-1}) + (1-\lambda_n) P_{\text{base}}(w_i|w_{i-n+2}^{i-1})$$

### 4. Katz backoff

- Similar to interpolation
- Backoff to lower N-grams in case higher N-grams not present in training corpus



Dr. Seba Susan Delhi Technological University