Evaluating Language Models

Natalie Parde
UIC CS 421

Evaluating Language Models

- Two types of evaluation paradigms:
  - Extrinsic
  - Intrinsic
- **Extrinsic evaluation:** Embed the language model in an application, and compute changes in task performance
- **Intrinsic evaluation:** Measure the quality of the model, independent of any application

Perplexity

- Intrinsic evaluation metric for language models
- Perplexity (PP) of a language model on a test set is the **inverse probability of the test set**, normalized by the number of words in the test set

More formally….

- $PP(W) = \sqrt[n]{\frac{1}{P(w_1 w_2 \ldots w_n)}} = \sqrt[n]{\prod_{i=1}^{n} \frac{1}{P(w_i|w_1 \ldots w_{i-1})}}$
  - Where *W* is a test set containing words w_1, w_2, …, w_n
  - History size depends on n-gram size
    - $P(w_i|w_{i-1})$ vs $P(w_i|w_{i-2}w_{i-1})$, etc.
- Higher conditional probability of a word sequence → lower perplexity
  - Minimizing perplexity = maximizing test set probability according to the language model

Example: Perplexity

Training Set

Word	Frequency
CS	10
421	10
Statistical	10
Natural	10
Language	10
Processing	10
University	10
of	10
Illinois	10
Chicago	10

Example: Perplexity

Training Set

Word	Frequency
CS	10
421	10
Statistical	10
Natural	10
Language	10
Processing	10
University	10
of	10
Illinois	10
Chicago	10

Test String

CS 421 Statistical Natural Language Processing University of Illinois Chicago

Example: Perplexity

Training Set

Word	Frequency
CS	10
421	10
Statistical	10
Natural	10
Language	10
Processing	10
University	10
of	10
Illinois	10
Chicago	10

Test String

CS 421 Statistical Natural Language Processing University of Illinois Chicago

$$PP(W) = \sqrt[n]{\frac{1}{P(w_1 w_2 \ldots w_n)}} = \sqrt[n]{\prod_{i=1}^{n} \frac{1}{P(w_i | w_1 \ldots w_{i-1})}}$$

Example: Perplexity

Training Set

Word	Frequency
CS	10
421	10
Statistical	10
Natural	10
Language	10
Processing	10
University	10
of	10
Illinois	10
Chicago	10

Test String

CS 421 Statistical Natural Language Processing University of Illinois Chicago

$$PP(W) = \sqrt[n]{\frac{1}{P(w_1 w_2 \ldots w_n)}} = \sqrt[n]{\prod_{i=1}^{n} \frac{1}{P(w_i | w_1 \ldots w_{i-1})}}$$

P("CS") = C("CS") / C(<all unigrams>) = 10/100 = 0.1

Example: Perplexity

Training Set

Word	Frequency
CS	10
421	10
Statistical	10
Natural	10
Language	10
Processing	10
University	10
of	10
Illinois	10
Chicago	10

Test String

CS 421 Statistical Natural Language Processing University of Illinois Chicago

$$PP(W) = \sqrt[n]{\frac{1}{P(w_1 w_2 \ldots w_n)}} = \sqrt[n]{\prod_{i=1}^{n} \frac{1}{P(w_i | w_1 \ldots w_{i-1})}}$$

P(“CS”) = C(“CS”) / C(<all unigrams>) = 10/100 = 0.1

P(“421”) = C(“421”) / C(<all unigrams>) = 10/100 = 0.1

Example: Perplexity

Training Set

Word	Frequency	P(Word)
CS	10	0.1
421	10	0.1
Statistical	10	0.1
Natural	10	0.1
Language	10	0.1
Processing	10	0.1
University	10	0.1
of	10	0.1
Illinois	10	0.1
Chicago	10	0.1

Test String

CS 421 Statistical Natural Language Processing University of Illinois Chicago

$$PP(W) = \sqrt[n]{\frac{1}{P(w_1 w_2 \dots w_n)}} = \sqrt[n]{\prod_{i=1}^{n} \frac{1}{P(w_i | w_1 \dots w_{i-1})}}$$

Example: Perplexity

Training Set

Word	Frequency	P(Word)
CS	10	0.1
421	10	0.1
Statistical	10	0.1
Natural	10	0.1
Language	10	0.1
Processing	10	0.1
University	10	0.1
of	10	0.1
Illinois	10	0.1
Chicago	10	0.1

Test String

CS 421 Statistical Natural Language Processing University of Illinois Chicago

$$PP(W) = \sqrt[n]{\frac{1}{P(w_1 w_2 \dots w_n)}} = \sqrt[n]{\prod_{i=1}^{n} \frac{1}{P(w_i | w_1 \dots w_{i-1})}}$$

PP("CS 421 Statistical Natural Language Processing University of Illinois Chicago")

$$= \sqrt[10]{\frac{1}{0.1*0.1*0.1*0.1*0.1*0.1*0.1*0.1*0.1*0.1}} = 10$$

Example: Perplexity

Training Set

Word	Frequency	P(Word)
CS	1	
421	1	
Statistical	1	
Natural	1	
Language	1	
Processing	1	
University	1	
of	1	
Illinois	1	
Chicago	91	

Test String

Illinois Chicago Chicago Chicago Chicago Chicago Chicago Chicago Chicago Chicago

$$PP(W) = \sqrt[n]{\frac{1}{P(w_1 w_2 \ldots w_n)}} = \sqrt[n]{\prod_{i=1}^{n} \frac{1}{P(w_i|w_1 \ldots w_{i-1})}}$$

Example: Perplexity

Training Set

Word	Frequency	P(Word)
CS	1	0.01
421	1	0.01
Statistical	1	0.01
Natural	1	0.01
Language	1	0.01
Processing	1	0.01
University	1	0.01
of	1	0.01
Illinois	1	0.01
Chicago	91	0.91

Test String

Illinois Chicago Chicago Chicago Chicago Chicago Chicago Chicago Chicago Chicago

$$PP(W) = \sqrt[n]{\frac{1}{P(w_1 w_2 \dots w_n)}} = \sqrt[n]{\prod_{i=1}^{n} \frac{1}{P(w_i | w_1 \dots w_{i-1})}}$$

Example: Perplexity

Training Set

Word	Frequency	P(Word)
CS	1	0.01
421	1	0.01
Statistical	1	0.01
Natural	1	0.01
Language	1	0.01
Processing	1	0.01
University	1	0.01
of	1	0.01
Illinois	1	0.01
Chicago	91	0.91

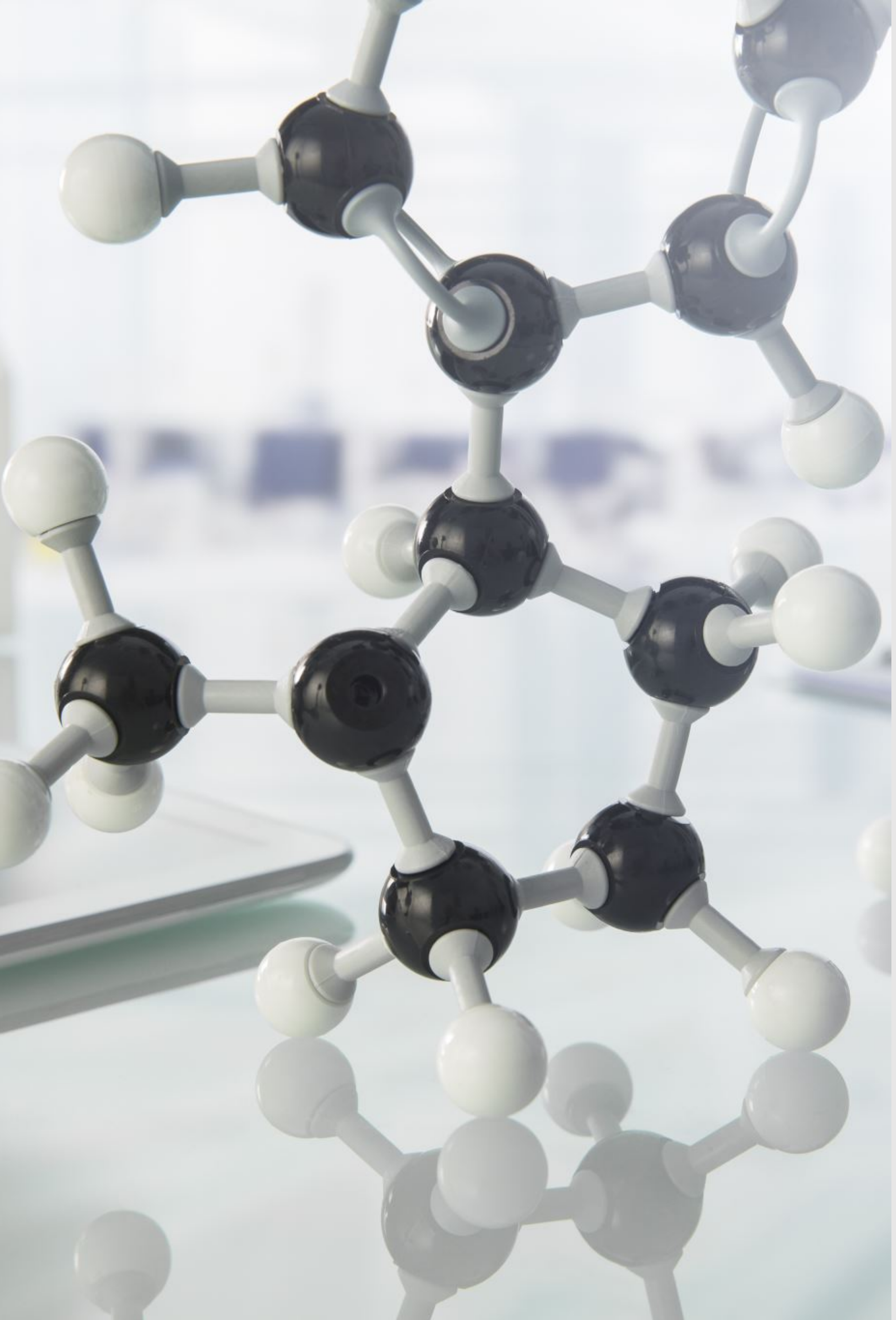
Test String

Illinois Chicago Chicago Chicago Chicago Chicago Chicago Chicago Chicago Chicago

$$PP(W) = \sqrt[n]{\frac{1}{P(w_1 w_2 \ldots w_n)}} = \sqrt[n]{\prod_{i=1}^{n} \frac{1}{P(w_i|w_1 \ldots w_{i-1})}}$$

PP("CS 521 Statistical Natural Language Processing University of Illinois Chicago")

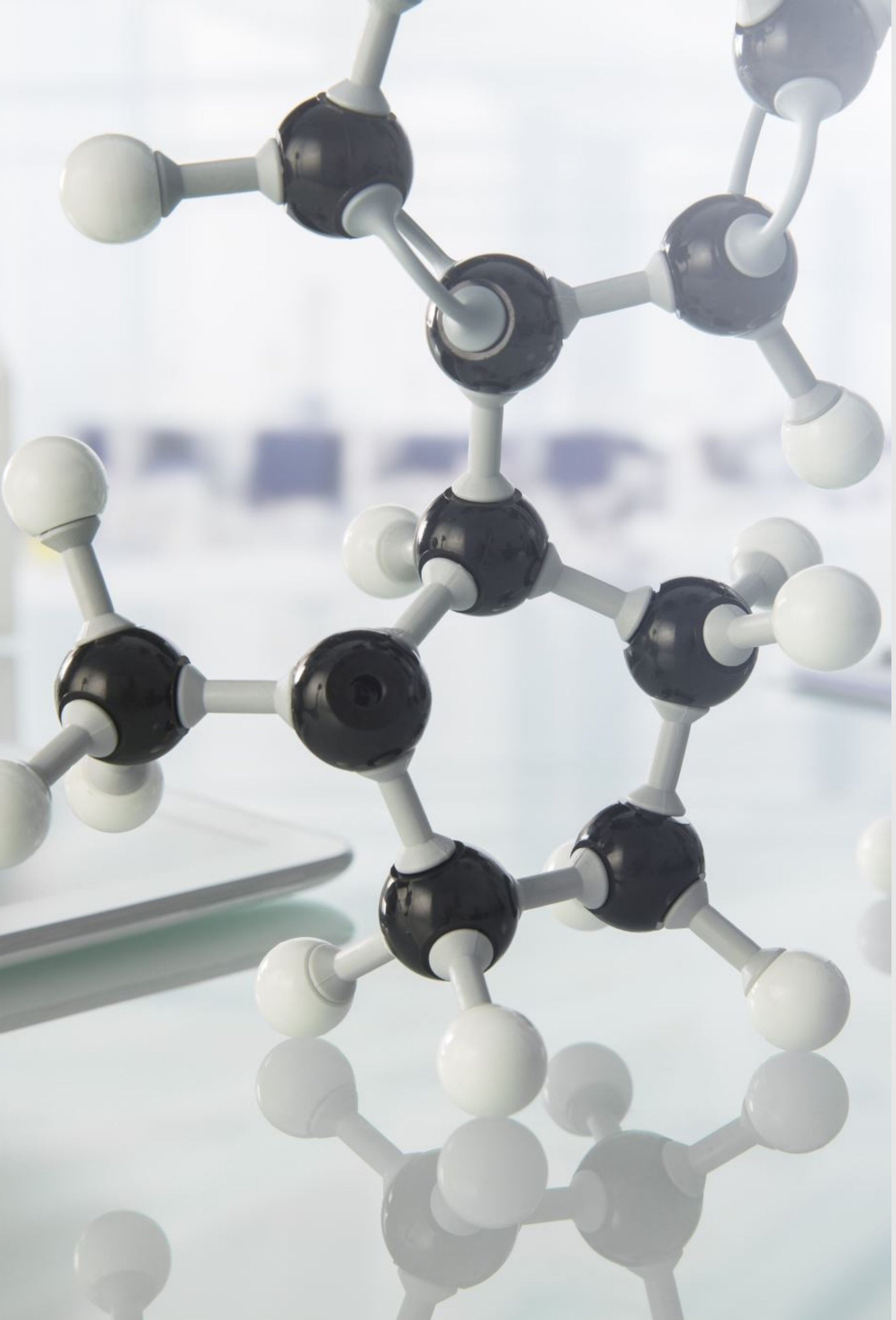
$$= \sqrt[10]{\frac{1}{0.01*0.91*0.91*0.91*0.91*0.91*0.91*0.91*0.91*0.91}} = 1.73$$



Perplexity can be used to compare different language models.

Which language model is best?

- Model A: Perplexity = 962
- Model B: Perplexity = 170
- Model C: Perplexity = 109



Perplexity can be used to compare different language models.

Which language model is best?

- Model A: Perplexity = 962
- Model B: Perplexity = 170
- Model C: Perplexity = 109

A cautionary note….

- Improvements in perplexity do not guarantee improvements in task performance!
- However, the two are often correlated (and perplexity is quicker and easier to check)
- Strong language model evaluations also include an extrinsic evaluation component