The Markov approximation

- From previous slide, notice that can estimate ***some*** of the factors of the joint distribution:

$$P(sentence) = P(\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(T)}) = \underbrace{P(\mathbf{x}^{(1)})}_{\text{yes}} \cdot \underbrace{P(\mathbf{x}^{(2)} \mid \mathbf{x}^{(1)})}_{\text{yes}} \cdot \underbrace{P(\mathbf{x}^{(3)} \mid \mathbf{x}^{(1)}, \mathbf{x}^{(2)})}_{\text{yes}} \ldots \underbrace{P(\mathbf{x}^{(T)} \mid \mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(T-1)})}_{\text{No!}}$$

- The Markov assumption: $P(\mathbf{x}^{(t)} \mid \mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(t-1)}) = P(\mathbf{x}^{(t)} \mid \mathbf{x}^{(t-n)}, \ldots, \mathbf{x}^{(t-1)})$

- Example:

$$P(dog \mid the, quick, brown, fox, jumped, over, the, lazy) = P(dog) \quad \leftarrow \quad (n = 1)$$

$$P(dog \mid the, quick, brown, fox, jumped, over, the, lazy) = P(dog \mid lazy) \quad \leftarrow \quad (n = 2)$$

$$P(dog \mid the, quick, brown, fox, jumped, over, the, lazy) = P(dog \mid the, lazy) \quad \leftarrow \quad (n = 3)$$

$$P(dog \mid the, quick, brown, fox, jumped, over, the, lazy) = P(dog \mid over, the, lazy) \quad \leftarrow \quad (n = 4)$$

n-gram models

- The Markov assumption: $P(\mathbf{x}^{(t)} \mid \mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(t-1)}) = P(\mathbf{x}^{(t)} \mid \mathbf{x}^{(t-n)}, \ldots, \mathbf{x}^{(t-1)})$

- n-gram models are language models that use the Markov assumption with a specific selection of n.

$$P(dog \mid the, quick, brown, fox, jumped, over, the, lazy) = P(dog) \quad \leftarrow \quad (n = 1) \quad \text{Unigram}$$

$$P(dog \mid the, quick, brown, fox, jumped, over, the, lazy) = P(dog \mid lazy) \quad \leftarrow \quad (n = 2) \quad \text{Bigram}$$

$$P(dog \mid the, quick, brown, fox, jumped, over, the, lazy) = P(dog \mid the, lazy) \quad \leftarrow \quad (n = 3) \quad \text{Trigram}$$

$$P(dog \mid the, quick, brown, fox, jumped, over, the, lazy) = P(dog \mid over, the, lazy) \quad \leftarrow \quad (n = 4) \quad \text{4-gram}$$

- This should look familiar to Lecture-03 (Word2Vec)! Skip-gram modeling generalizes *n*-grams by not considering the sequence order of the *context* words.