1 PCFGS/CKY

Here’s a PCFG with start symbol VP, nonterminals \{VP, PP, P, V, NNS\}, and terminals \{sells, to, books, meetings\}.

\[
\begin{align*}
VP & \rightarrow V \text{ NNS PP} [0.5] \\
VP & \rightarrow V \text{ NNS} [0.5] \\
PP & \rightarrow P \text{ NNS} [1.0] \\
V & \rightarrow \text{sells} [0.5] \\
V & \rightarrow \text{books} [0.5] \\
P & \rightarrow \text{to} [1.0] \\
NNS & \rightarrow \text{books} [0.5] \\
NNS & \rightarrow \text{meetings} [0.5]
\end{align*}
\]

1. Apply lossless binarization to this grammar to obtain a binary grammar.

2. Fill in the CKY chart for \text{sells books}. Use log base 2.

3. Fill in the CKY chart for \text{books meetings}. Use log base 2.

4. Fill in the CKY chart for \text{sells books to books}. Use log base 2.
2 Skip-gram

The skip-gram model is defined by

\[ P(\text{context} = y | \text{word} = x) = \frac{\exp (v_x \cdot c_y)}{\sum_{y'} \exp (v_x \cdot c_{y'})} \]

where \( x \) is the “main word”, \( y \) is the “context word” being predicted, and \( v, c \) are \( d \)-dimensional vectors corresponding to words and contexts, respectively. Note that each word has independent vectors for each of these, so each word really has two embeddings.

The skip-gram model considers the neighbors of a word to be words within a \( k \)-word window on either side (i.e., \( k = 1 \) gives the two immediately adjacent words). The skip-gram objective, log likelihood of this training data, is \( \sum_{(x,y)} \log P(y|x) \), where the sum is over all training examples.

1. What happens to the number of training examples if \( k = 5 \)?

2. How does the runtime change with larger \( k \)?

3. Think about the context of the word balloons in the following sentences:

   - He blew up balloons for the birthday party
   - I popped balloons using a pin because I didn’t like seeing the bright colors

In these contexts (and more generally), what do you think will change about what the skip-gram model learns for balloons as you change \( k = 1 \) to \( k = 3 \)? What about \( k = 10 \)?