Iran University of Science & Technology

School of Computer Engineering

Assignment #6

Natural language processing

BY:

DR. Behrouz Minaei, Fall 2024

Teaching Assistants:

Nafiseh Ahmadi

Reza Alidoost

Due: 1403/11/15

Contents

Notes

1. **There is no option to submit this assignment after the deadline.**
2. Submit the answers in a complete PDF file and the code for the questions in the .ipynb format (including the notebook cell outputs) in a compressed file named HW6_StudentID.zip by the specified deadline.
3. If a student submits the project earlier than the deadline and achieves 75% of the score, up to 24 hours will be added to their allowable delay time.
4. It is important to note that the explanation of the code and the obtained results must be included in the PDF file. Code without a report will result in a score deduction.
5. The evaluation of the assignment will be based on the correctness of the solution and the completeness and accuracy of the report.
6. Assignments must be completed individually, and group work on assignments is not allowed.
7. Please allocate sufficient time for the assignment and avoid leaving it until the last days.
8. You can ask your questions in the relevant group.

good luck.

Problem 1

a) A sentence can easily have more than one parse tree that is consistent with a given CFG. How do PCFGs and non-probability-based CFGs differ in terms of handling parsing ambiguity? **(5 points)**
Consider the following PCFG for problems (b)-(e).

production rule	probability
S → VP	1.0
VP → Verb NP	0.7
VP → Verb NP PP	0.3
NP → NP PP	0.3
NP → Det Noun	0.7
PP → Prep Noun	1.0
Det → the	0.1
Verb → Cut | Ask | Find| ...	0.1
Prep → with | in | ...	0.1
Noun → envelope | grandma | scissors | men | suits | summer | ...	0.1

b) Draw the top-ranked parse tree for the sentence below by applying the given PCFG. Does the result seem reasonable to you? Why or why not? **(5 points)**
Cut the envelope with scissors.

c) Draw the top-ranked parse tree for the sentence below by applying the given PCFG. Does the result seem reasonable to you? Why or why not? **(5 points)**
Ask the grandma with scissors.

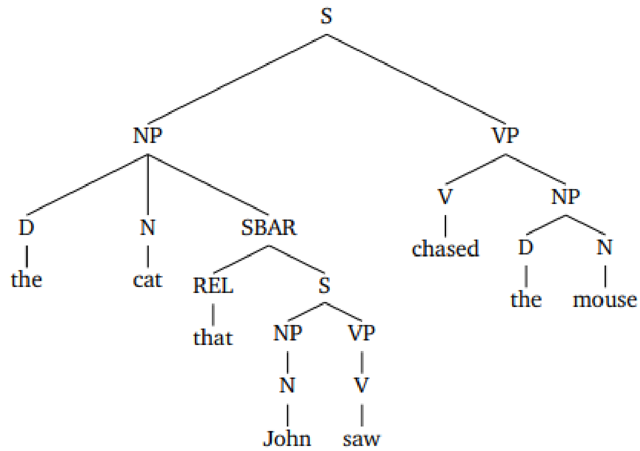
d) Describe how you would lexicalize the given PCFG in order to address the problem you hopefully noticed in (b) and/or (c). Then show specifically how the production rules below should be modified according to your lexicalization scheme. **(5 points)**

production rule	probability
VP → Verb NP	0.7
VP → Verb NP PP	0.3

e) The following two sentences exhibit parsing ambiguities. How would your lexicalized PCFG from (d) handle these ambiguities? **(5 points)**
Find the men in suits.
Find the men in summer

Problem 2

Given the sentence The cat that John saw chased the mouse, with the following parse tree: **(20 points)**



And the following grammar rules (where the superscript + indicates the head):

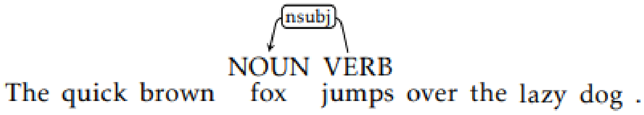
$$
\begin{aligned}
&\mathrm{S} \rightarrow \mathrm{NP}\ \mathrm{VP}^{+}\\
&\mathrm{NP} \rightarrow \mathrm{N}\\
&\mathrm{NP} \rightarrow \mathrm{D}\ \mathrm{N}^{+}\\
&\mathrm{NP} \rightarrow \mathrm{D}\ \mathrm{N}^{+}\ \mathrm{SBAR}\\
&\mathrm{VP} \rightarrow \mathrm{V}^{+}\ \mathrm{NP}\\
&\mathrm{VP} \rightarrow \mathrm{V}\\
&\mathrm{SBAR} \rightarrow \mathrm{REL}^{+}\ \mathrm{S}
\end{aligned}
$$

a) List the headwords of the following non-terminals:
- the SBAR
- the NP “The cat that John saw”
- the topmost S
- the VP “chased the mouse”

b) Draw the dependency tree resulting from the conversion using the given head rules.

Problem 3

Complete the dependency analysis of the sentence:



Use the Universal Dependency (UD) annotation scheme. Each word should have a part-of-speech tag and an incoming edge from its head, labeled with a basic dependency relation (e.g., nsubj, obj, . . .).[1] **(20 point)**

Problem 4

Give an example of an ambiguous sentence and explain the ambiguity in terms of dependency relations.

For example: John sees Mary with a telescope.
Ambiguity: the prepositional phrase ‘with a telescope’ is a dependent of either ‘sees’ or ‘Mary’.

Other ideas for difficult sentences: headlines, informal language, questions, etc. Try giving the sentence to an online parser.[2]
Show the analysis given by the parser. Is it the correct analysis? If not, show the expected analysis. If it is, repeat with a more difficult sentence. **(15 points)**

Problem 5

You have been provided with the Transformer.ipynb notebook, which contains a complete implementation of a Transformer model. Your task is to write a comprehensive report on the provided code. This report should include: **(30 points)**

1. Positional Encoding

1.1 Sine and Cosine Angles

1. What is the purpose of calculating angles using sine and cosine for positional encodings?

---

[1] Useful links:UD annotation guidelines: http://universaldependencies.org/guidelines.html, UD POS tags: http://universaldependencies.org/u/pos/index.html, UD relations: http://universaldependencies.org/u/dep/index.html

[2] For example: https://demos.explosion.ai/displacy/, https://lindat.mff.cuni.cz/services/udpipe/run.php, http://corenlp.run/

2. Why is it important to represent positional information in this way for the Transformer?

1.2 Sine and Cosine Positional Encodings

1. How does the positional_encoding function utilize the computed angles to generate positional encodings?
2. What is the shape of the positional encoding matrix, and how does it integrate with input embeddings?
3. Why is it beneficial to use sinusoidal positional encodings instead of learned embeddings?

2. Masking

2.1 Padding Mask

1. What role does the padding mask play in the Transformer architecture?
2. How is the padding mask constructed, and which parts of the input sequence does it affect?

2.2 Look-ahead Mask

1. How does the look-ahead mask prevent information leakage in the decoder during training?
2. Describe the structure of the look-ahead mask. How is it applied during self-attention computations?

3. Self-Attention

1. What is the purpose of the scaled_dot_product_attention function?
2. How does scaling the dot product in self-attention improve stability?
3. In the exercise, what are the key outputs of the scaled_dot_product_attention function, and how are they used in the Transformer?

4. Encoder

4.1 Encoder Layer

1. What are the components of an encoder layer in the Transformer?
2. How is the multi-head attention mechanism implemented in the EncoderLayer?
3. What is the role of the feedforward network in the encoder layer?

5. Decoder

5.1 Decoder Layer

1. What are the key differences between the encoder layer and the decoder layer?
2. How does the decoder layer utilize the look-ahead mask and padding mask?
3. In the DecoderLayer exercise, how are the outputs of the multi-head attention layers processed?

5.2 Full Decoder

1. How does the full decoder assemble the layers to process the input sequence?
2. How does the decoder interact with the encoder outputs during training?

6. Transformer

1. How does the Transformer class integrate the encoder and decoder components?
2. What role does the shared embedding layer play in the overall architecture?
3. In the final exercise, how is the forward pass implemented for the Transformer?