CS 784: Assignment 2 (100 + 10 Points)

Instructor: Freda Shi
Due Date: Wednesday, March 26, 2025, 11:59 pm (ET; Waterloo Time)

Instructions

- Submit the code solutions (§ 1) in a file named `code.zip` to the `Assignment 2 Code` Dropbox on `LEARN`.

  The zip file should contain the following under the root directory:

  - `src/` directory containing all the source code. You may use Jupyter notebooks or any other format you prefer.
  - `README.md` with instructions on how to run the code to reproduce the results.
  - `report.pdf` with the corresponding answers to the questions in § 1. If you use Jupyter notebooks, feel free to export the notebook as a PDF.

  You are strongly encouraged to use Python for this assignment to streamline things, but you can use any programming language you are comfortable with.

- Submit your paper review (§ 2) in a file named `review.pdf` to the `Assignment 2 Paper Review` Dropbox on `LEARN`.
- You have a 3-day (72 hours) grace period to submit the assignment with no penalty.

  No late assignment will be accepted after March 29, 2025, 11:59 pm (ET).

- This assignment is to be done individually.

1 Programming (50 + 10 points): Conditional Entropy with PCFGs and Transformers

The conditional entropy of the next word given the previous words in a sentence measures the information conveyed by each word [1]. Formally, given a sentence $w_1, w_2, \ldots, w_n$, the conditional entropy of the next word w_{n+1} is

$$H(w_{n+1} \mid w_1, w_2, \ldots, w_n) = -\sum_{w_{n+1}} p(w_{n+1} \mid w_1, w_2, \ldots, w_n) \log p(w_{n+1} \mid w_1, w_2, \ldots, w_n),$$

where $p(w_{n+1} \mid w_1, w_2, \ldots, w_n)$ is the probability of the next word given the previous words. Different probabilistic models can be used to estimate this probability. Among them, we are now interested in two models: probabilistic context-free grammars (PCFGs) and transformers.

The following (poorly documented) codebase

- `https://github.com/ExplorerFreda/conditional-entropy`

reimplements `https://github.com/timhunter/ccpc` to calculate the conditional entropy of the conditional entropy above. Part of your task is to understand the codebase and make it work with a grammar that is in a different format from what it supports.

- **Task 1 (10 points):** Read the codebase documentation. Report the **conditional entropy** $H(w_i \mid w_{<i})$ and **conditional probability** $P(w_i \mid w_{<i})$ of each word in the sentence "`Jon hit the stick with the dog`" under the PCFG specified in `data/strauss.pcfg` (in the GitHub repo). Here, $w_{<i}$ denotes the words before the i-th word in the sentence.
- **Task 2 (10 points):** Report the **conditional entropy** $H(w_i \mid w_{<i})$ and **conditional probability** $P(w_i \mid w_{<i})$ of each word in the sentence "`Jon hit the stick with the dog`" under the GPT-2 (`gpt2`, `gpt2-medium`, and `gpt2-large`) models. You may consider the conditional probability is given by

  $$P(w_i \mid w_{<i}) = \frac{\exp\left(logits(w_i \mid w_{<i})\right)}{\sum_{w' \in \mathcal{V}} \exp\left(logits(w' \mid w_{<i})\right)},$$

  where $\mathcal{V}$ is the vocabulary of GPT-2, and $logits(w_i \mid w_{<i})$ is the logit of the word w_i given the previous words $w_{<i}$ produced by the GPT-2 model. Note that $\mathcal{V}$ may contain subwords, and we will simply consider them as words when calculating the probability.

  Hint: You may wish to check out the GPT-2 models here: `https://huggingface.co/openai-community/gpt2`.
- **Task 3 (15 points):** Now you are given the weighted CFG in `data/ptb.wcfg` (provided in the GitHub repo). All the weights are positive integers.

Convert this WCFG to a PCFG by linearly normalizing the weights associated with each non-terminal and pre-terminal (i.e., the left-hand side), report the **conditional entropy** $H(w_i \mid w_{<i})$ and **conditional probability** $P(w_i \mid w_{<i})$ of each word in the sentence "`colorful green ideas sleep furiously`" under this PCFG.

Compare the results with those produced by GPT-2 models (`gpt2`, `gpt2-medium`, and `gpt2-large`).

- **Task 4 (15 points):** Compare the conditional probability and entropy of the next word under `data/ptb.wcfg` and GPT-2 models.

  Explore a few sentences of your choice that receive non-zero probability under both models. In what cases do the two models agree or disagree? What are the implications of these differences?

  Note that this is an open-ended question without a single correct answer. You are encouraged to explore different sentences and provide your insights.

  Hint: For simplicity, you may consider sentences with all words in the vocabulary of GPT-2, which can be checked by looking at the tokenization result.

- **Extra Credit (10 points):** What does "left recursion" (mentioned in the GitHub repo FAQ) mean? Why does it matter in calculating conditional entropy?

Hints:

- You may find the Google Colab (`https://colab.google/`) environment helpful for running the code. The free version should be sufficient for this assignment.
- For old codebases, it's common to have (in)compatibility issues with the dependencies—even though a newer version of a libaray should in principle work, this happens from time to time. If you encounter such issues, you may need to pay some additional effort to resolve them, by checking out the old version of the library or modifying the code according to the error messages.
- Your conditional probability and entropy should match the standard answer up to a reasonable precision of 1e-3. Note: you may need to adjust the code to get accurate results.

2 Paper Review (50 points)

Write a review for the paper

Visually Grounded Reasoning across Languages and Cultures

Fangyu Liu, Emanuele Bugliarello, Edoardo Maria Ponti, Siva Reddy, Nigel Collier, Desmond Elliott

In EMNLP (2021)

`https://aclanthology.org/2021.emnlp-main.818/`

Your review should include the following parts:

- Summary of the paper (10 points)
- Strengths of the paper (10 points)
- Weaknesses of the paper (10 points)
- Detailed comments, suggestions, and questions for the authors (20 points)

Please limit your review to less than 2 pages (A4 paper, 12pt font, single-spaced).
Each part will be graded separately by the following criteria:

$$\textit{Your Grade}(\%) = \min\left\{F_1(\textit{Your Review}, \textit{Freda's Review} \cup \textit{Filtered Review from Class})/0.6, 100\right\},$$

where F_1 is the F_1 score between your review and the union of Freda's review and a good peer review. *Filtered Review from Class* refer to the review arguments from the class that are endorsed by Freda. *Freda's Review* $\cup$ *Filtered Review from Class* will be a set of unique "ground-truth" arguments for the review of this paper. While all arguments will be weighted equally when calculating precision, the arguments will be weighted differently when calculating your recall based on the importance and levels of detail.[1] Your review will also be considered as a set of arguments, so you are encouraged (though not required) to write in bullet points.

Everyone might misunderstand or simply miss some points, so you will receive full marks if your review gets higher than 60% F_1 score compared to the ground-truth arguments.

Below is an example of grading. Suppose you raised 7 points about the paper (1-7), the "ground-truth" set consists of 6 points (1, 2, 3, 7, 8, 9). In the "ground-truth" set, points 1, 2 are weighted 2 and points 3, 7, 8, 9 are weighted 1. Then your recall will be

$$\text{Recall (weighted)} = \frac{2+2+1}{2+2+1+1+1+1} = \frac{5}{8} = 0.625.$$

Your precision will be

$$\text{Precision (unweighted)} = \frac{4}{7} = 0.571.$$

Your F_1 score will be the harmonic mean of the recall and precision

$$F_1 = \frac{2 \times 0.625 \times 0.571}{0.625 + 0.571} = 0.597.$$

[1] There may be even some arguments weighted zero; that is, you will not be penalized for missing them through recall, but you will be rewarded for mentioning them through the precision score.

That is, you will receive $0.597/0.6 = 99.5\%$ of the marks for this part.

The ACL instruction (`https://2023.aclweb.org/blog/review-acl23/`) to reviewers might be helpful. You may also find a few good reviews Freda received in the past on `LEARN`, as well as an example review she wrote for a recent paper.

Please note that this is a conference paper published in late 2021, so please avoid suggestions that involve work **in or after early 2021**, such as comparing to GPT-4 performance.

References

[1] John Hale. The information conveyed by words in sentences. Journal of psycholinguistic research, 32:101–123, 2003. Springer.