Language Models
Exercises 2
Session 5

Each task is worth 1 point.

Task 1. Recall what perplexity is. Consider a language where words are sequences of digits, and:

a) consist of blocks of k digits (in a block, all digits are the same, i.e., the block has the form d^n, e.g., 888888888),

b) any block can be followed by a block of any digit (with equal probability).

We analyze a very large text from this language. Calculate the perplexity of this text for the following models: unigram, bigram, and optimal n-gram (what is n?). Provide numerical values for $k = 10$.

Task 2. A company proposed to a ministry in a certain country to create a tool to detect texts generated by a language model using perplexity(i.e., the company claimed that generated texts have lower perplexity than natural texts, so a perplexity value below a certain threshold suggests an unnatural text origin). Consider why this might work in this way. Why is this not an ideal solution, and how can a user of a model like ChatGPT control the perplexity of generated text?

Task 3. In this task, we will think about generating text using N-grams (specifically 2- and 3-grams), where the resulting text must meet additional requirements. Explain why the ńatural method"(i.e., normally generating text from left to right with a specified length, checking the condition, and possibly re-generating) is not effective in such tasks. For each variant, propose an algorithm that significantly increases the chances of successful generation (compared to the natural method):

a) generate text of length M where the word at position k is predefined,

b) generate text of length M where words at even positions are predefined (i.e., generate only odd positions, starting numbering from 0),

c) generate a relatively short text with specified first and last words,

d) for a list of strings $[s_1, \ldots, s_n]$, generate text of length n such that the i-th word has suffix s_i.

Task 4. You trained a language model (e.g., GPT). However, due to an error, all the texts were read backward, and the model saw: `['I', ' like', ' ice', 'creams']` as `['creams', ' ice', ' like', 'I']`. The model trained successfully, and the training was expensive (conducted in a paid cloud). When you noticed the error, you started new training (for the correct model), but now the only issue is justifying the expenses of training the first model. What arguments would you use to convince the management that these expenses were indeed justified?

Task 5. Steganography, according to Wikipedia, is:

> The study of communicating in such a way that the presence of the message cannot be detected.

We will use a high-quality language model (i.e., one that produces texts virtually indistinguishable from ńaturalónes), known to both parties in the communication.

The scenario is as follows: You are in prison (unjustly convicted), and you want to communicate with Comrades on the Outside. Fortunately, you can exchange printed letters without restrictions, but the Warden reads all correspondence – and nothing should seriously alarm them, as this could have Serious Consequences. In each letter, you want to send a message (a few dozen bits or slightly more), such that it is unambiguously readable but completely invisible to the Warden (for various reasons, physical modifications to the medium, such as lemon juice, are ruled out). Fortunately, in the Prison Laundry where you work, there is a computer with a language model and Python installed, and you have good access to it, with no staff interest.

How would you organize invisible communication with your Comrades, assuming all details could be discussed in advance and that the Comrades have access to exactly the same model, Python version, etc.?

Task 6. ⋆ Research what Kerckhoffs's Principle is (`https://en.wikipedia.org/wiki/Kerckhoffs's_principle`). Discuss whether your solution from the previous point complies with this principle. If not, can it be adjusted to comply? (Assume the language model is the key.)

Task 7. Assume that GPT-2 is installed on every computer as part of the operating system. Your task is to design a method for compressing Polish texts using this language model. Try to make the description detailed enough so that it can be implemented without difficulty (note: the output of the compression can be a bitstream, and you do not need to worry about how to pack it into bytes or its robustness against distortions).

Task 8. In the Papuga model, it is crucial that the prefix provided to the model **does not end with a space**. Observe in several examples how the generations differ for the same prefix with and without a space at the end. Explain why this phenomenon occurs.

Task 9. Consider using a language model to generate rhymed poetry, such as the following:

> I have only one burning desire
> Let me stand next to your fire

What characterizes such a text? Propose a realistic algorithm that uses a model like GPT-2 to generate rhymes (clearly, it requires search; the point is to make this process as efficient as possible).

Task 10. ⋆ Propose a task where predicting the next token can result in a useful application (or be interesting or entertaining). The task should not be related to natural language (or at least not solely related to it). Describe where you source (or how you generate) the corpus and how tokenization is performed.