CAP 6640 – Natural Language Processing **Spring 2025**
Homework #6 Due date: April 4, 2025

Problem 1:

The softmax temperature is a parameter used to adjust how a language model selects words during text generation. It changes the probability distribution of possible next words, allowing us to tune the balance between randomness and predictability in the output. Given a vector of logits z_i for each possible token, the temperature scaled softmax is defined as:

$$P(w_i) = \frac{e^{z_i/T}}{\sum_j e^{z_j/T}}$$

where T is the temperature, z_i is the pre-softmax score for token i, and $P(w_i)$ is the resulting probability for token i.

When the temperature is close to 0.7 or below, the model becomes more confident and focused. It strongly favors high-probability words, resulting in precise, factual, and consistent outputs. This setting is well-suited for tasks where accuracy and coherence are important, such as summarization, and QA.

In contrast, a higher temperature which can be selected as 1.2 or more makes the output distribution flatter, giving more weight to less likely words. This leads to more creative, varied, and sometimes unexpected text, which is ideal for poetry, storytelling, or brainstorming tasks. However, very high temperatures can also introduce nonsense or off-topic responses.

Problem 2: What are the advantages and limitations of sampling-based decoding in open-ended text generation tasks like storytelling?

Problem 3: Compare and contrast BLEU, ROUGE, and BERTScore in evaluating text generation outputs. Why are word-overlap metrics insufficient for evaluating dialogue or storytelling?

Problem 4: Describe how hybrid QA architectures integrate both retrieval-based and generative models. Provide an example of such a system.

Problem 5: Define coreference resolution. Why is it critical for full-text understanding tasks such as QA and summarization?

Problem 6: How do end-to-end neural models approach coreference resolution differently from traditional rule-based systems?

Problem 7: How does the DecaNLP framework unify multiple NLP tasks without relying on task-specific modules or parameters?

Problem 8: Describe anti-curriculum learning. How does starting with harder tasks improve generalization in multitask training?

Problem 9: What is the MQAN architecture? List two mechanisms it uses to handle multiple tasks as QA problems.

Problem 10: What is the difference between a general language model and a conditional language model? Provide examples of tasks suitable for each.