Language Models

Rafael Mejía, Santiago Pineda, Juan López, Andrés Álvarez [1]

{rmejiaz, spinedaq, jslopezvi, aalvarezme}@unal.edu.co[1]
Universidad Nacional de Colombia

July 7, 2023

Overview

Introduction

What is a language model?

- At its core, a language model is a computational framework that aims to capture the underlying structure, patterns, and semantics of natural language.
- They serve as an intelligent system capable of predicting the likelihood of a sequence of words, given the context of previously observed words.
- Language models provide a way to estimate the probability distribution of words, enabling us to generate coherent and contextually appropriate sentences.

Importance and applications

- Language models act as a fundamental building block in many natural language processing (NLP) tasks, allowing machines to understand and generate human-like text.
- By modeling the patterns and dependencies within language, these models empower us to automate language-related tasks, enhance communication, and unlock new possibilities in various fields.
- Language models have found applications in diverse domains, including machine translation, question-answering systems, sentiment analysis, content generation, personalized recommendations, and much more.

Evolution

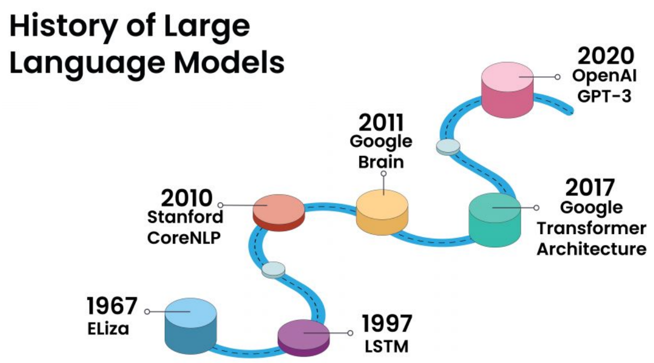
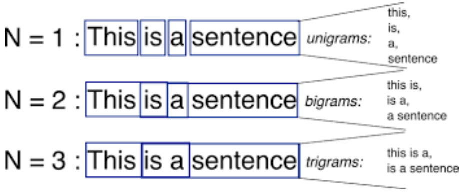


Figure: Evolution of language models

Basic Language Models

Bigram Model

- The Bigram Model is a simple language model that focuses on capturing dependencies between consecutive pairs of tokens in a given sequence.
- It assumes that the probability of a word depends only on the preceding word. In other words, it estimates the probability of a word based on the occurrence of the previous word.
- Key Assumption:
  $P(token_i | token_i - 1) \approx P(token_i | token_i - 1, token_i - 2, ..., token_1)$



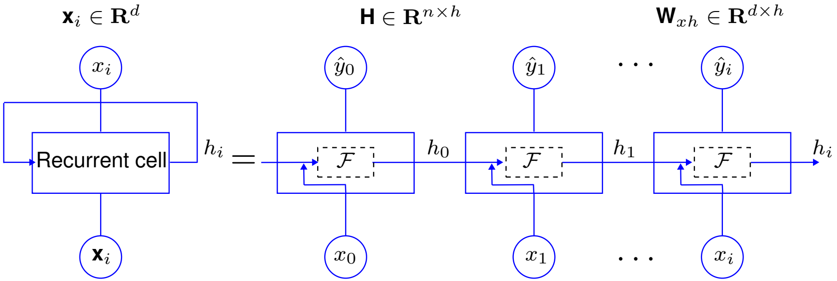
Calculating probabilities

- To estimate the probability of a word given the previous word, we can calculate the frequency of their co-occurrence in a large corpus.
- Example:
  Let's consider the sentence: "The cat is sitting on the mat." Using the Bigram Model, we calculate the probability of the word "sitting" given the previous word "is" as the frequency of the bigram "is sitting" divided by the frequency of the word "is".

Problems

- The Bigram Model assumes that the probability of a word depends only on its immediate predecessor, neglecting the influence of words beyond the previous one.
- It fails to capture long-range dependencies and contextual information, leading to limitations in generating coherent and contextually accurate sentences.

Recurrent Neural Networks (RNN)



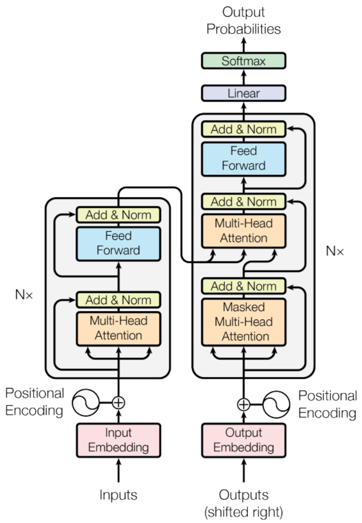
$$\hat{y}_i = \mathcal{F}(x_i, h_{i-1}) \tag{1}$$

Problems

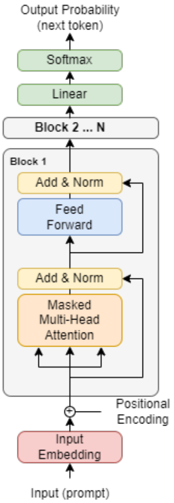
- Distant positions in the sequence can be disregarded
- Parallelizing the work is challenging because it processes variables sequentially

Transformer Architecture

Transformer Architecture



Decoder only Transformer



Encoder-Decoder vs Decoder Only Transformer

- Encoder-Decoder architectures are used for sequence to sequence tasks, such as language translation. In this case, the encoder generates a representation of the input sequence and passes it to the decoder via cross-attention. Then, the decoder is fed with a specific token to begin the generation of the target sequence. We can say that the encoder gives "context" to the decoder.
- Decoder Only architectures are used for natural language generation. In this case, no "context" is given to the decoder via cross-attention, meaning the output probabilities depend only on the inputs of the decoder.

Thank you!