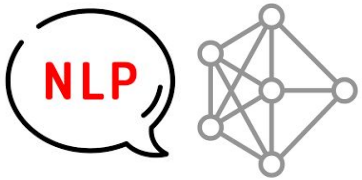




CHULA ENGINEERING Foundation toward Innovation COMPUTER



Decoding

2110572: Natural Language Processing Systems

Peerapon Vateekul & Ekapol Chuangsuwanich
Department of Computer Engineering,
Faculty of Engineering, Chulalongkorn University

Credit:

- Kasidis Kanwatchara
- Can Udomcharoenchaikit & Nattachai Tretasayuth

Outline

- Part1) Introduction
- Part2) Greedy decoding
- Part3) Random sampling
- Part4) Beam search

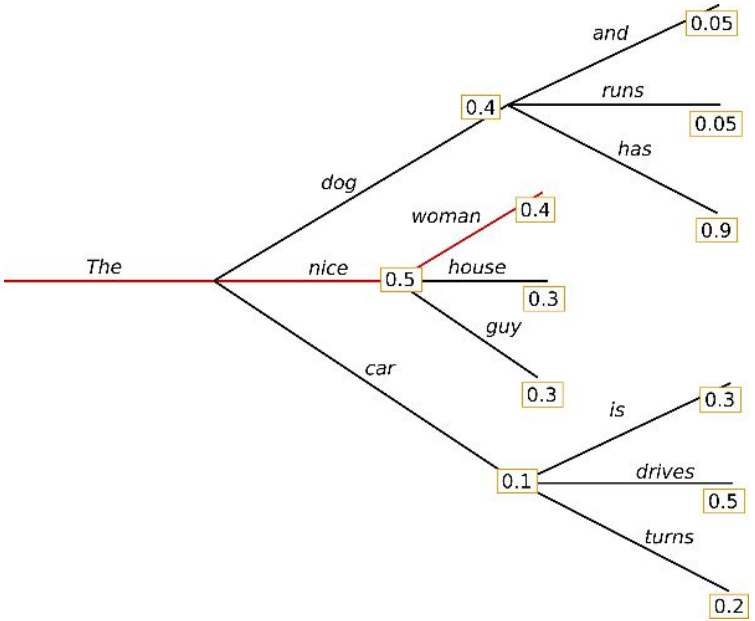
Part1) Introduction

Introduction

In sequence generation tasks, the task of selecting what the model outputs as a prediction is called **decoding**.

There are 3 methods for decoding:

1. Greedy decoding
2. Random sampling
3. Beam search



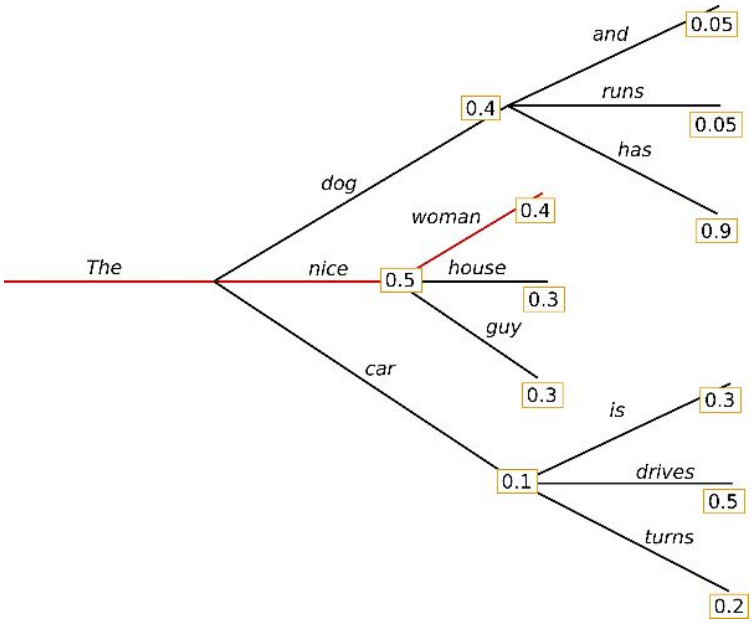
https://huggingface.co/blog/how-to-generate

Part2) Greedy decoding

Greedy decoding

Greedy decoding simply selects the token with **the highest probability** as the next token.

As shown in the picture, after "the", the continuation with the highest probability is the word "nice" therefore it is selected as the next token. This is done until it reaches the model's max sequence length or upon encountering an end-of-sentence token.



Greedy decoding

Greedy decoding is fast and simple, however, the generated text is usually sub-optimal. Sometimes, the model can even repeat itself.

```
He began his premiership by forming a five-man war cabinet which included
Chamerlain as Lord President of the Council, Labour leader Clement Attlee as
Lord Privy Seal (later as Deputy Prime Minister), Halifax as Foreign Secretary
and Labour's Arthur Greenwood as a minister without portfolio. In practice,
+ the cabinet was divided into three parts: the Cabinet of Ministers, the
+ Cabinet of Ministers of the Crown, and the Cabinet of Ministers of the Crown.
+ The Cabinet of Ministers was the most important part of the government. The
+ Cabinet of Ministers was the most important part of the government. The
+ Cabinet of Ministers was the most important part of the government. The
+ Cabinet of Ministers was the most important part of the government. The
+ Cabinet of Ministers was the most important part of the government. The
+ Cabinet of Ministers was the most important part of the government. The
+ Cabinet of Ministers was the most important part of the government. The
+ Cabinet of Ministers...
```

https://towardsdatascience.com/the-three-decoding-methods-for-nlp-23ca59cb1e9d

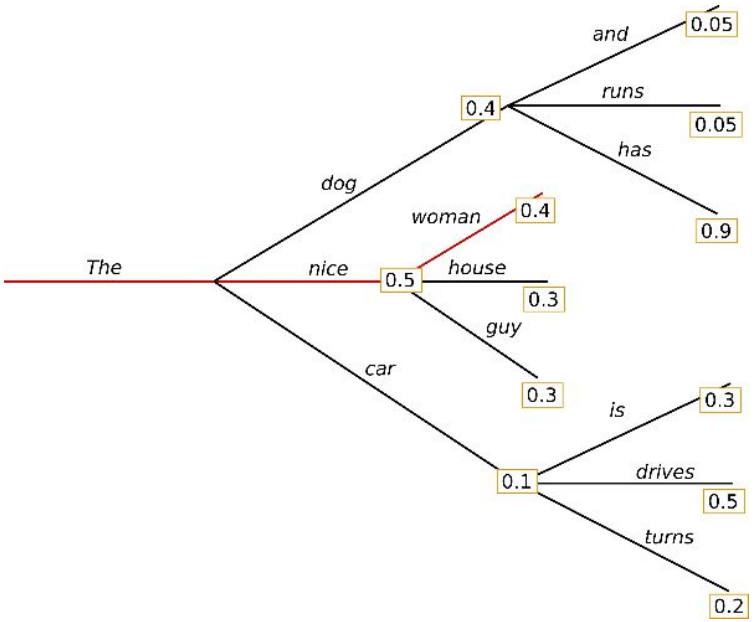
+

Part3) Random sampling

Random sampling

Random sampling chooses the next token **based on the probabilities**.

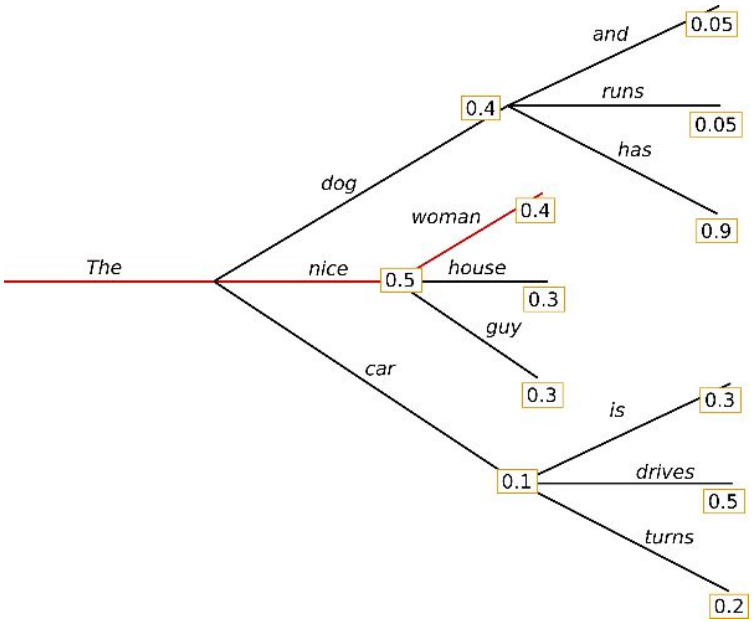
Using random sampling, the probability of selecting the token "nice", "dog", and "car" as the continuation is 50%, 40%, and 10%, respectively. The decoding also proceeds until reaching max sequence length or encountering the end-of-sentence token.



Random sampling

By the laws of probability, you are bound to eventually generating something gibberish by selecting multiple low probability tokens in a row.

To prevent this problem, top-k and top-p (nucleus sampling) are often used to improve the generation quality.



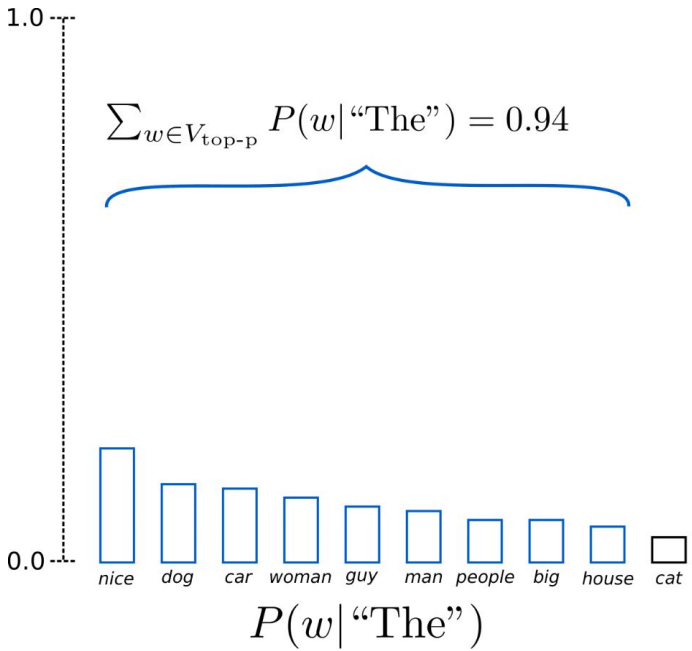
Random sampling

Top-k sampling simply limits the token selection to just top k (usually 20-40) words with the highest probabilities.

Top-p sampling or nucleus sampling dynamically limits the number of words by setting a probability threshold. Top-p sampling chooses from the smallest possible set of tokens whose cumulative probability exceeds the probability threshold.

For example, if we set p as 0.92, top-p sampling will select the *minimum* number of tokens whose cumulative probability is more than 92%

Note that both top-k and top-p can also be used together.



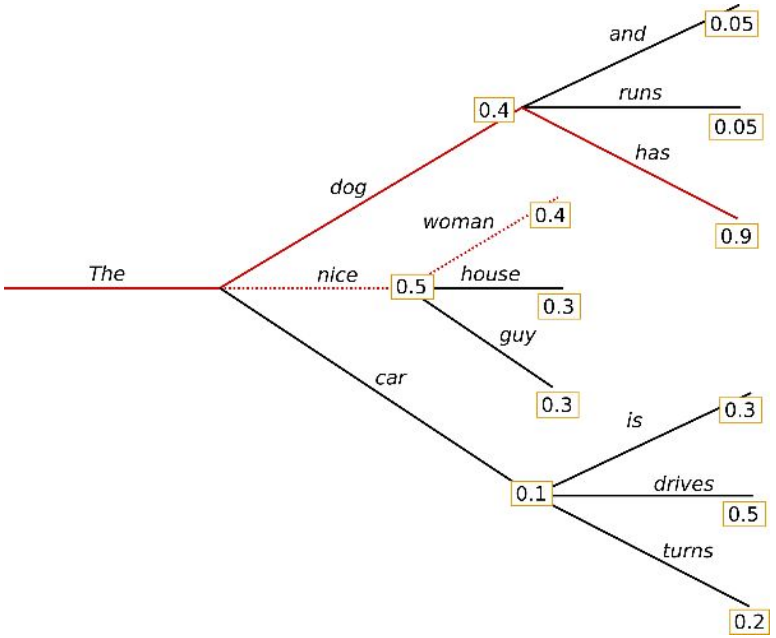
+

Part4) Beam search

Beam Search

The two techniques mentioned before selects the next token only based on its probability. On the other hand, Beam search allows us to explore further into each continuation until completion before choosing.

Beam search is relatively computationally expensive since it basically generates multiple sequences but it always find an output sequence that is more probable than greedy decoding.



Beam Search

From the example, we consider 2 "beams", i.e., we only keep the top 2 most probable sequence while we go through the generation process.

At time step 1, the beam search algorithm keeps tab on 2 most probable continuations: ("The", "nice") and ("The", "dog").

At time step 2, it continues to find the next word for each beam:

("The", "dog", "has") = 0.4*0.9 = **0.36**
("The", "nice", "woman") = 0.5*0.4 = 0.2

Suppose this is the end of the generation, the output of the algorithm will be "The dog has".