

Embedding

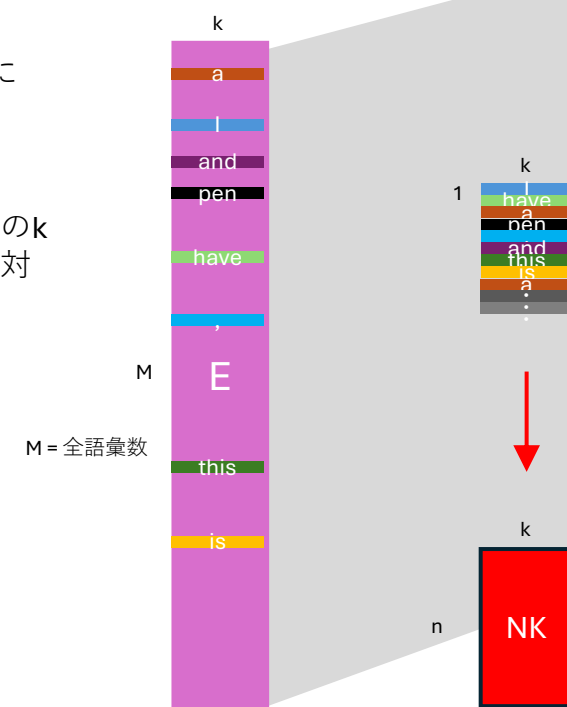
n (トークン数≒学習に用いる文の長さ)

I have a pen, and this is a

各トークンに対応するベクトルの取り出しと積み重ね

全語彙(トークン)にindex(番号)を割り振っておく

一つの語彙に一つのk次元のベクトルが対応する。



1つの学習用文章(I have a pen, and this is a....)について、
I → have
I have → a
I have a → pen
I have a pen → ,
I have a pen, → and
... のように文の途中までに対して続く1トークンを予測するように学習させる。

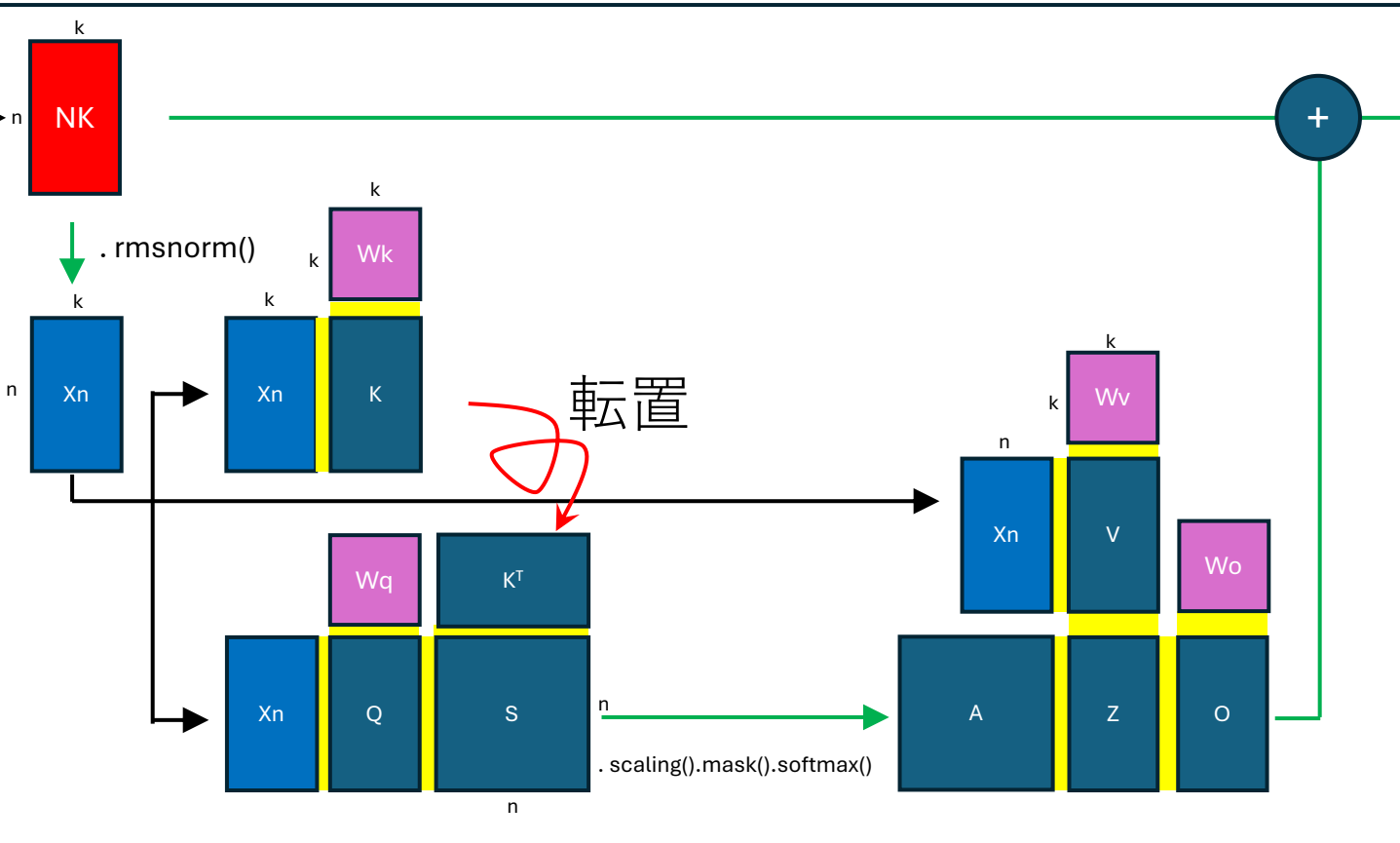
Embedding層では文章をトークンに分割し、トークンに対応するベクトルを積み重ねて行列にする。
行列は**Transformer**層(同じ構成のブロックの繰り返しによりなる)で処理され、**Unembedding**層に渡される。
Unembedding層では、入力文章中のトークンそれぞれについて、次の位置のトークンを予測する。学習(=パラメータの更新)対象となる行列はピンク色で示した。

推論時には、与えられた文章に続く次の1トークンを出力する。出力トークンを文章末尾に付け足して、次の出力を得る。これを繰り返す。

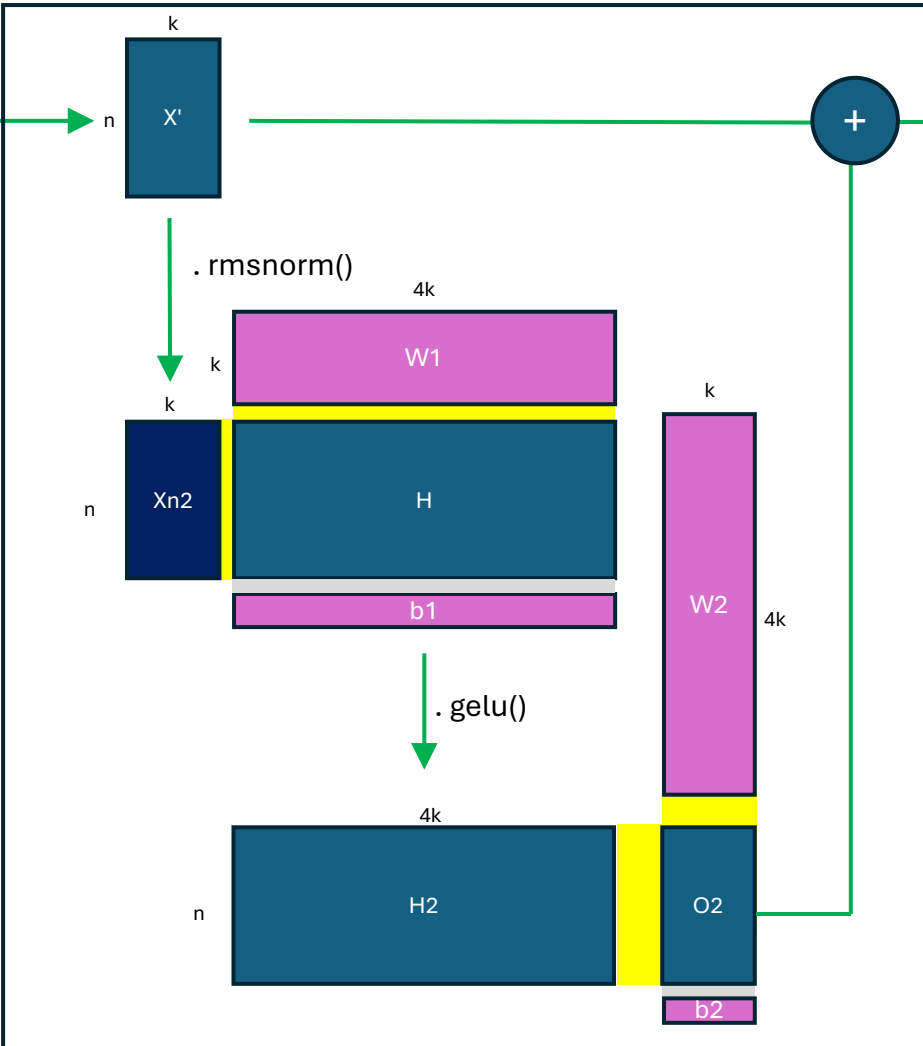
L回繰り返し

Transformer

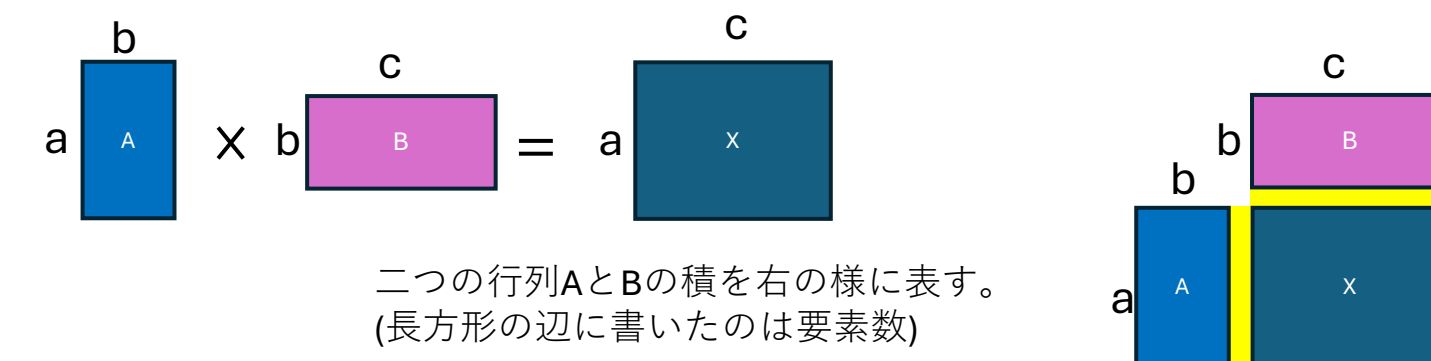
multi-head self-attention



FFN (Feed Forward Network)



この絵の行列計算の表し方



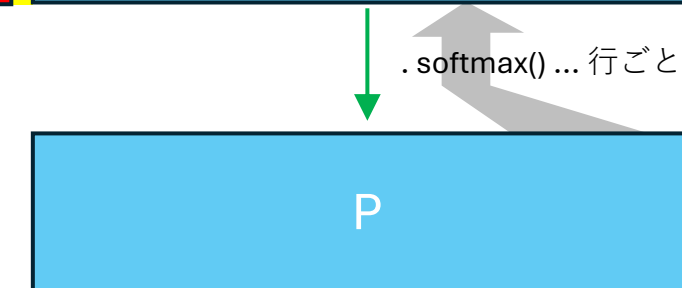
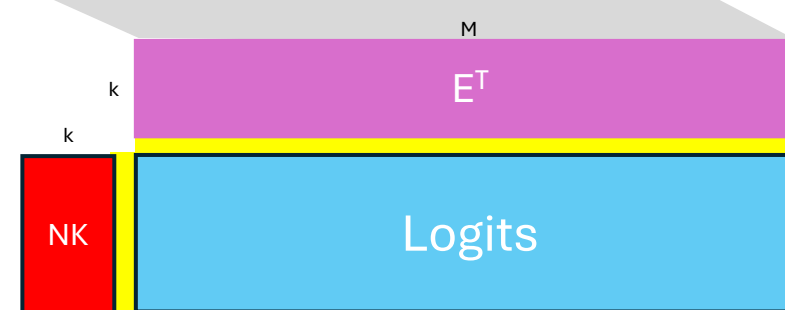
二つの行列AとBの積を右の様に表す。
(長方形の辺に書いたのは要素数)



これはD(1行)を、Cの各行に足す演算を表す。

Unembedding

入力文 : I have a pen, and this is a
正解ラベル: have a pen, and this is a

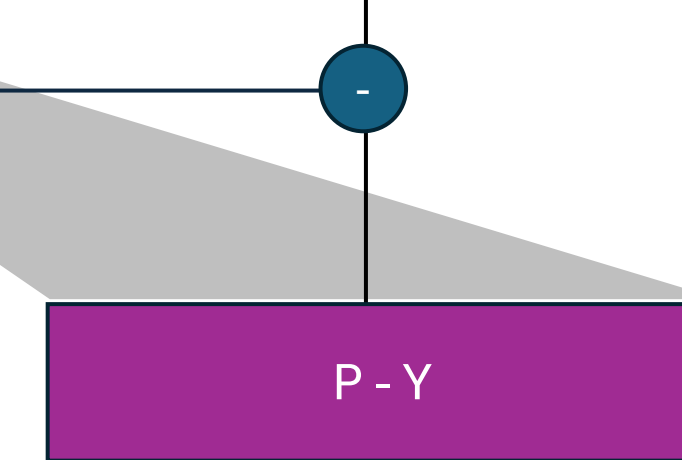


予測

各行の最大確率となるindex。Eのindexに対応する語を"予測"したものとする。



正解ラベルから作られたワンホットベクター(行ごと)



逆伝播に用いる

パラメーター数の例(Google DeepMind, Gemma 4 31B Dense):

M: 262,144

k: 5,376

L: 60

N: 262,144

作図: 大坪嘉行

よろしかったらこちらをどうぞ:

https://kyouikuwo.org/edu/REUP_main.php

