Benifit of Self Attention

Parallel Training

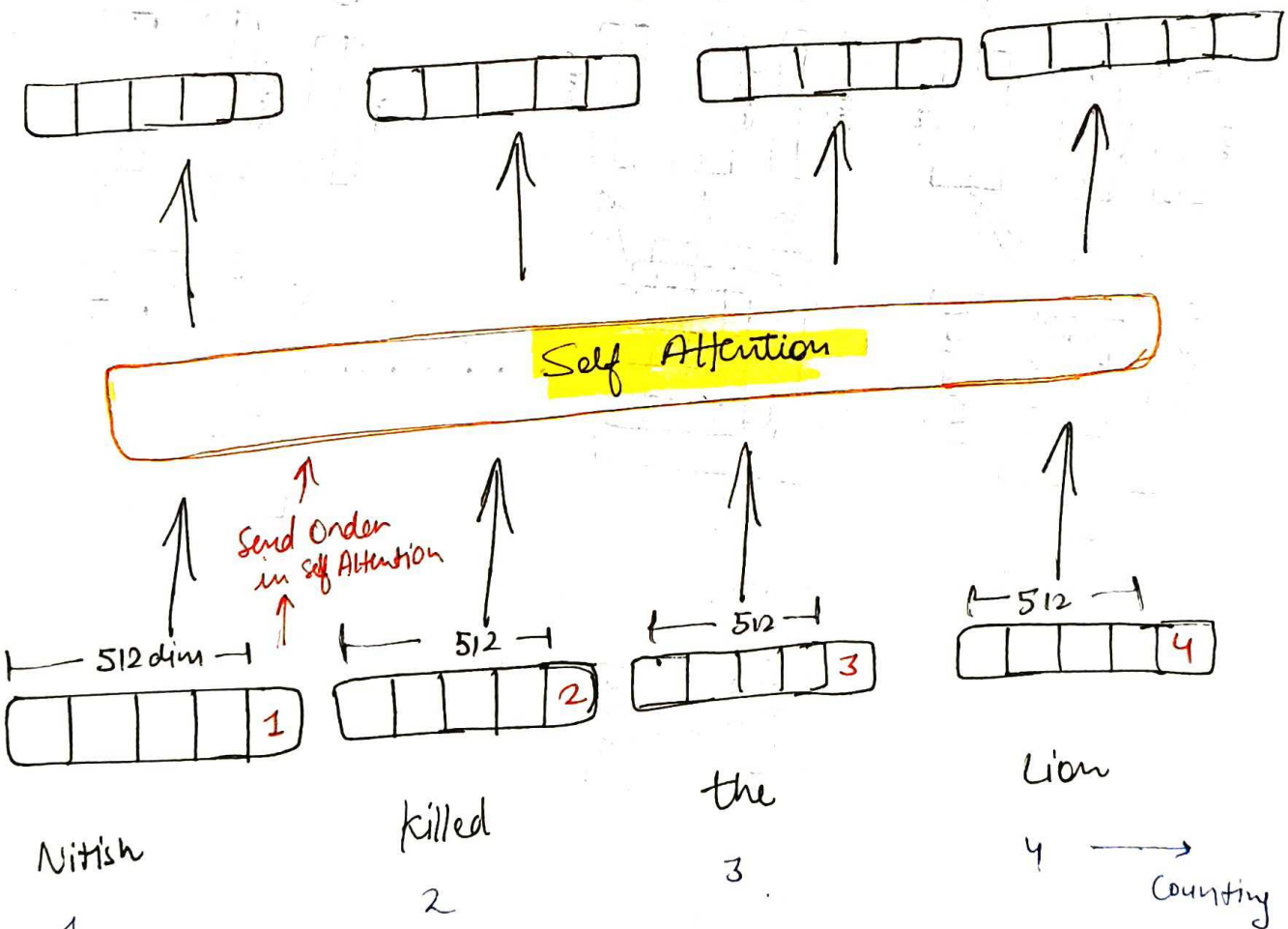
Drawback of Self Attention

word Order

eg:- "Nitish killed Lion" is same "Lion killed Nitish"

Positional Encoding in Transformer

Proposing a Simple Solution



Problem: 1) unbound → upper limit

eg:- 1 Lakh word document → last loop is 100000

And we use NN architecture → backpropagation

* create unstability ← Hate big number
* gradient unstable

Sol. counting Number divide with Total Number then range convert into 0 to 1.

for: $\frac{1}{100000} \approx 0$, $\frac{2}{100000}$, ---- $\frac{100000}{100000} = 1$

But this solution is not good.

Sentence 1 → Thank | You → 2nd position value is 1.

1 | 2
0 | 0

Sentence 2 → Nitish | Killed | the | Lion

1/4 | 2/4 | 3/4 | 4/4

→ 2nd position value is 0.5

So, there is no consistency. Position value will be change. NN confused → is 2nd position is 1 value or 0.5 value.

* Normalize Not Work

2) 

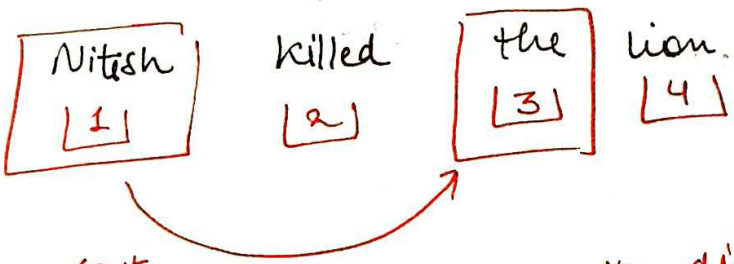
Nitish Killed the lion

→ This number is discrete and it is not good for NN.

→ NN generally prefer smooth Transition like ~~Number~~ continous Number.

→ Numerical Stability Problem.

3) Can't relative positions



(can't we find the distance with discrete function. (relative position)

(3 − 1 = 2)

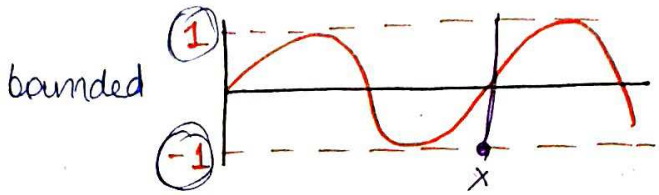
Summary of problem

→ unbounded —we want→ (bounded)

→ discrete ⟶ (continous)

→ Relative position ⟶ (periodical)

Sol Use Sin function (trignometric)

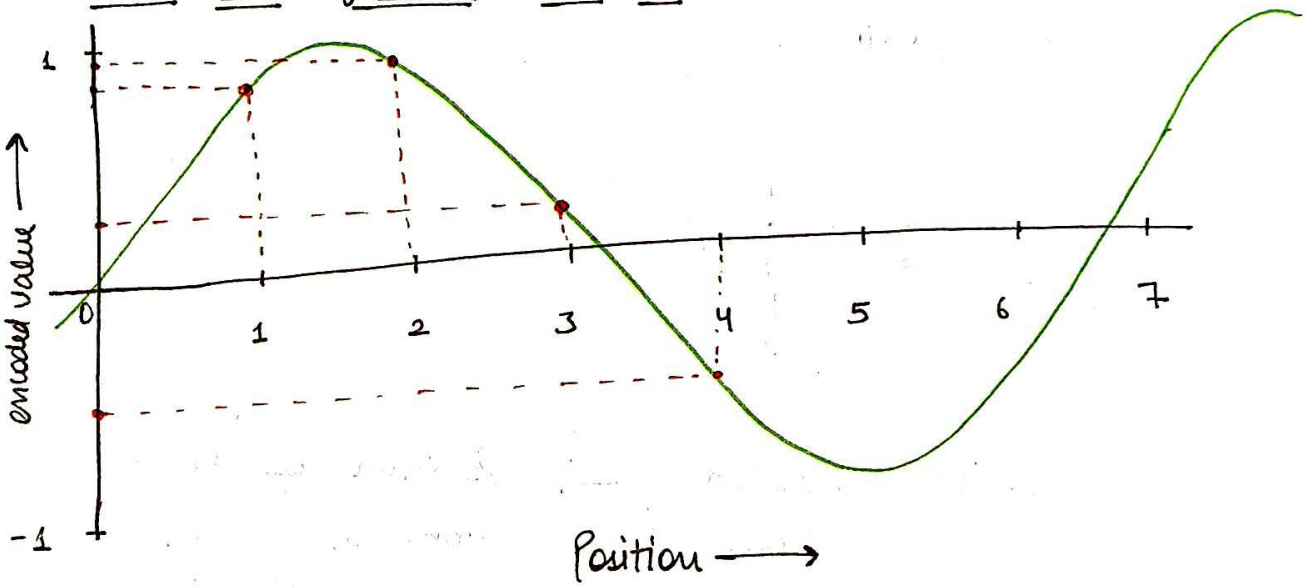


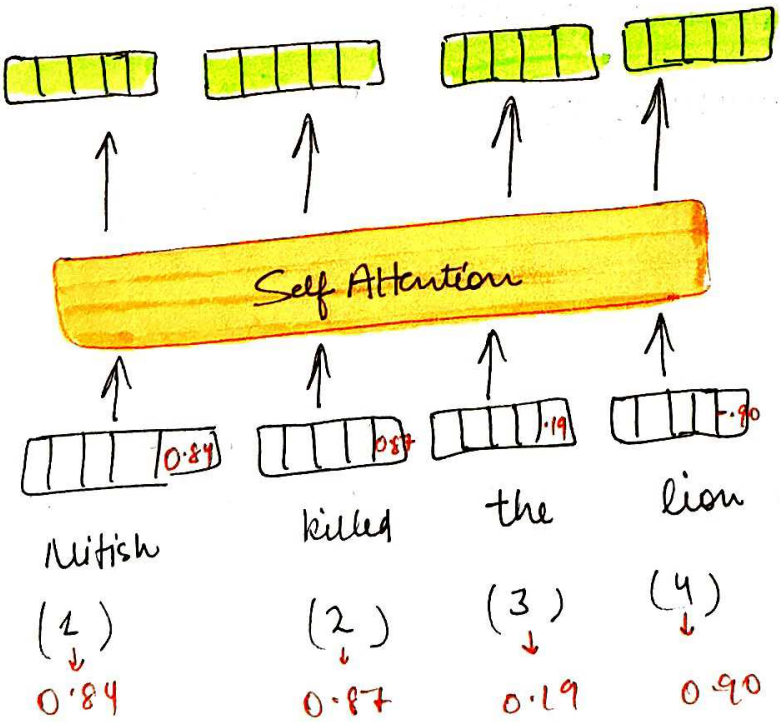
→ continous → every x has y value

→ value repeat after certain action

Positional encoder → Sin → better solution.

The Sine function as a solution





encoded value

$y = \sin(\text{position})$

$y = \sin(1) = 0.84$

$y = \sin(2) = 0.87$

$y = \sin(3) = 0.19$

$y = \sin(4) = -0.90$

Big Problem with Sin function

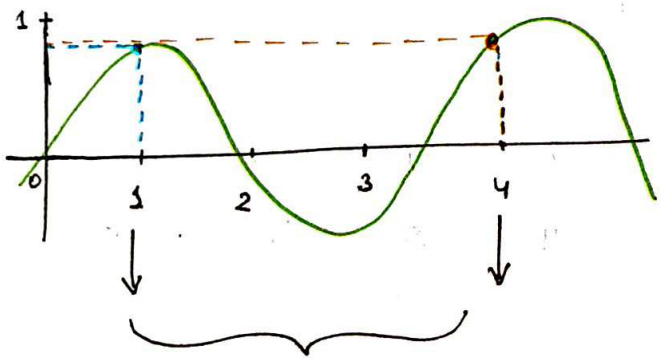
 → encoded value = 0.11

Nitish

* Koi aur word ka positional encoding same (0.11) nhi aana chahiye.

If positional encoded value is same then position of both word is same. It will create issue.

Sin curve → periodic

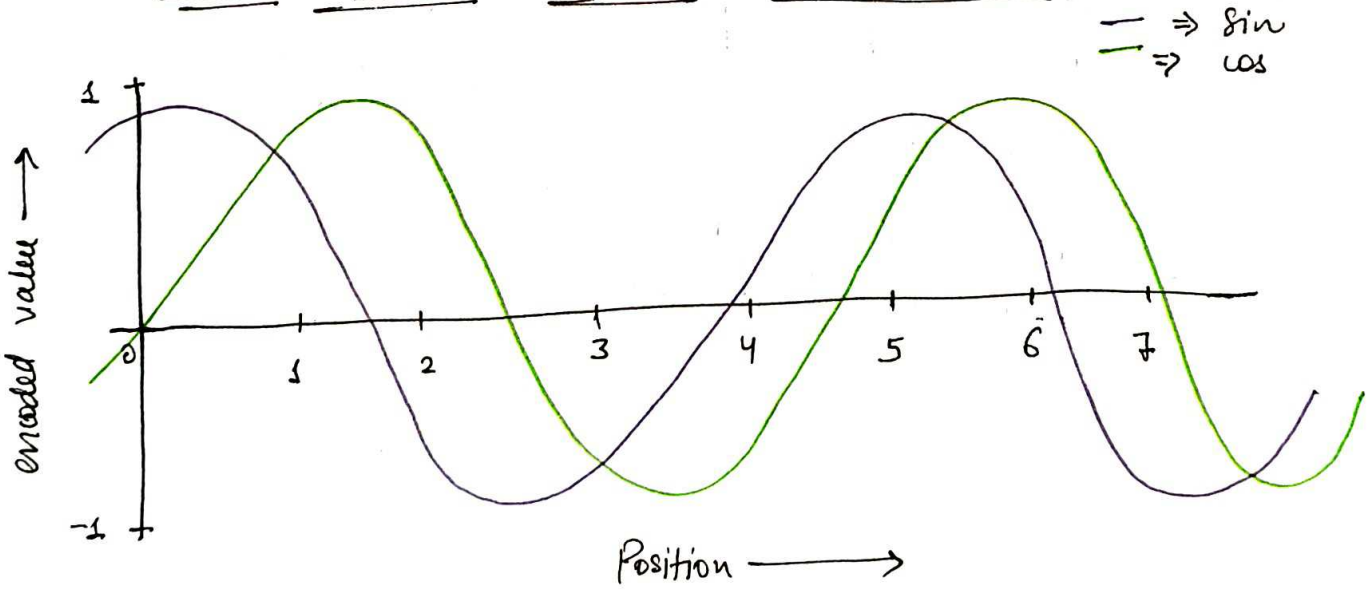


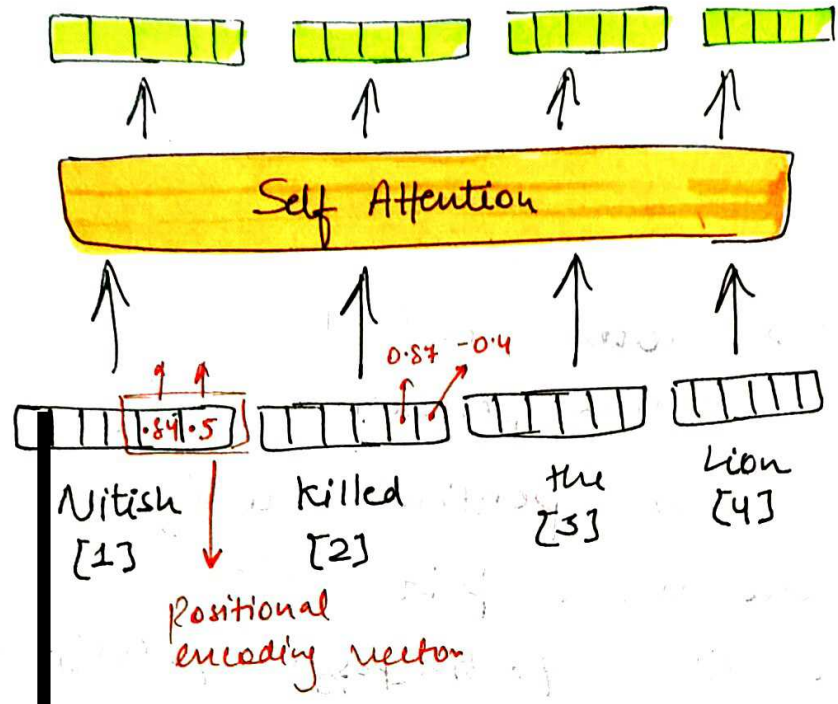
position encoded value is same → Position of both word is same according to Sin function

* Model will confuse.

Sol. Use double Trignometric function

The Sin and Cos function as a Solution





$y_1 = \sin(\text{position})$

$y_2 = \cos(\text{position})$

$[y_1, y_2] \rightarrow$ vector

Nitish = [0.84, 0.5]

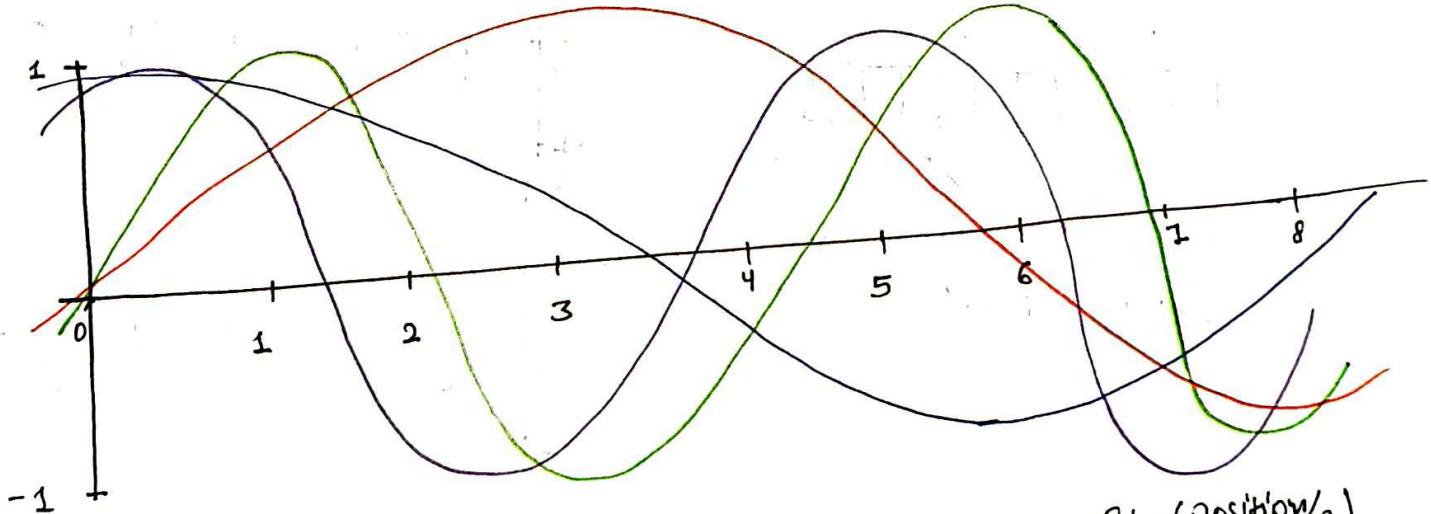
killed = [0.87, -0.4]

In this sol, also chance to get same vector of two different position word.

So, we use 4 different trignometric function

$y^1 = \sin(pos)$ $y^2 = \cos(pos)$ $y^3 = \sin(pos/2)$ $y^4 = \cos(pos/2)$

$[y^1, y^2, y^3, y^4] \rightarrow$ Vector



—— = Sin (position)

—— = Cos (position)

—— = Sin (position/2)

—— = Cos (Pos/2)

Nitish

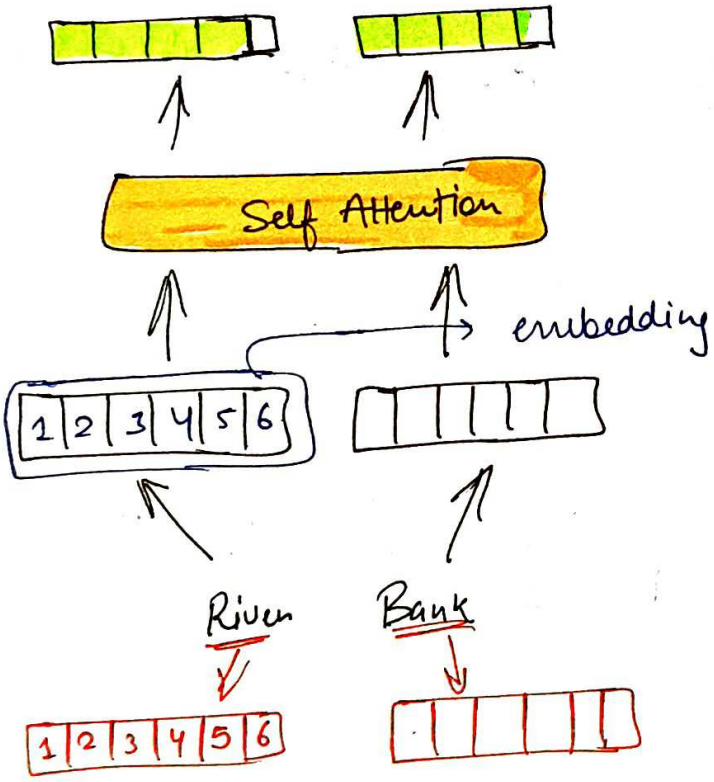
[1] → position encoding evector

[-, -, -, -]

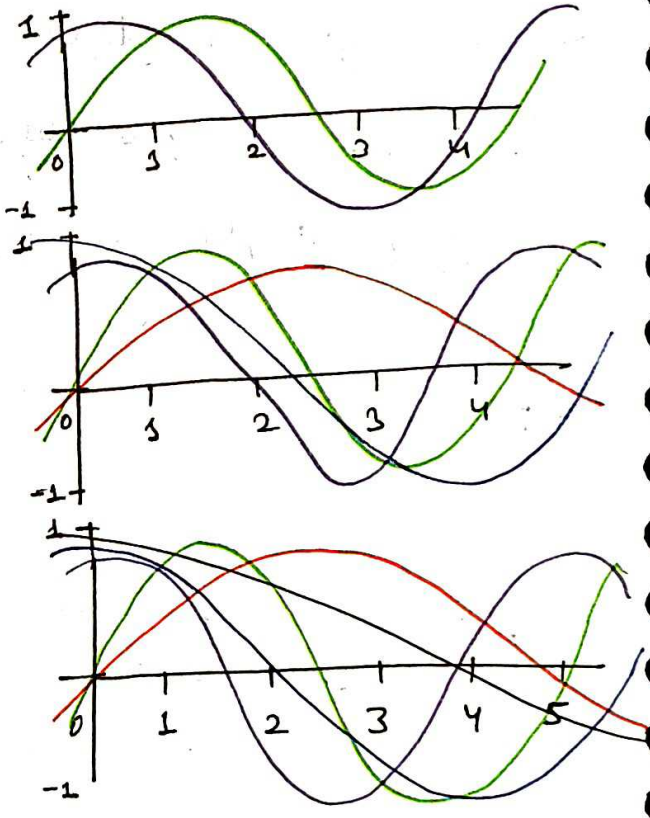
Very less probability → two word have same position encoding

But still if you see same position encoding vector then add one more pair of trignometric function like $y = \sin(pos/3)$ $y = \cos(pos/3)$

Positional Encoding in "Attention all you need"

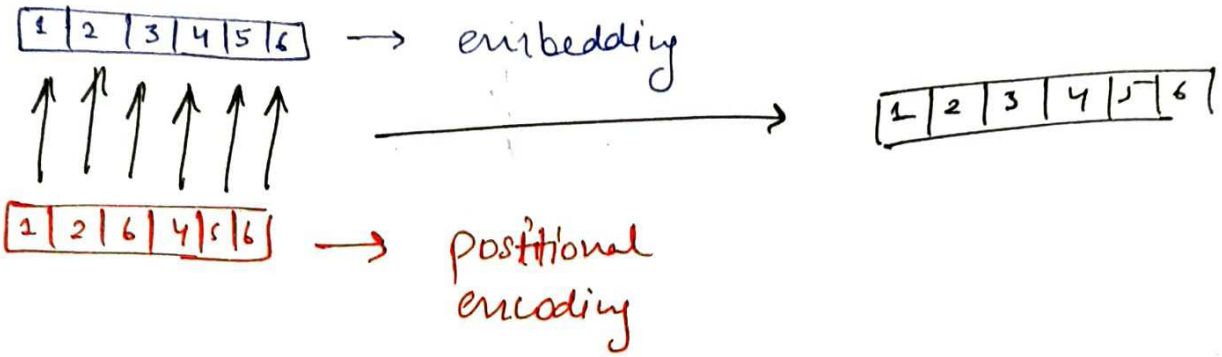


→ Vector (positional encoding)



* embedding is equal to vector size

Embedding (6 dim) + Pos encoding (6 dim) = 6 dim Vector



If concatenate → [6dim] [6dim] → [12 dim]

Training double up.

* Size of Vector increase → frequency of sin and cos function is decrease.

#So, how decide which frequency is good?

In the Research Paper → $PE_{(pos,2i)} = \sin(Pos/10000^{2i/d_{model}})$

$PE_{(pos,2i+1)} = \cos(Pos/10000^{2i/d_{model}})$

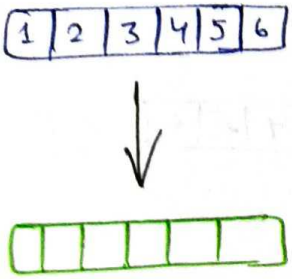
Pos → Position of word

d_{model} → Dimentionality of embedding

$i \rightarrow [0]$ to $\left[\frac{d_{model}}{2} - 1\right]$ ⇒ i=1, i=2, i=3 ...

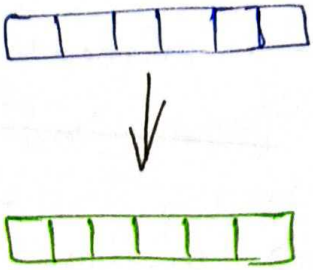
Pos = 0

River



Pos = 1

Bank



for i = 0 → pos = 0 (River)

$PE(0,0) = \sin(0/10000^{\circ}) = \boxed{0}$

$PE(0,1) = \cos(0/10000^{\circ}) = \boxed{1}$

(2i+1)

for i = 1 → pos = 0 (River)

$PE(0,2) = \sin(0/10000^{1/3}) = \boxed{0}$

$PE(0,3) = \sin(0/10000^{1/3}) = \boxed{1}$

for i = 2 → pos = 0 (River)

$PE(0,4) = \sin(0/10000^{2/3}) = \boxed{0}$

$PE(0,5) = \cos(0/10000^{2/3}) = \boxed{1}$

River → | 0 | 1 | 0 | 1 | 0 | 1 |

Bank word → Positional Encoding

for i = 0 → pos = 1 (Bank)

$PE(1,0) = \sin(1/10000^{0}) = 0.84$

$PE(1,1) = \cos(1/10000^{0}) = 0.54$

for i = 1 → pos = 1 (Bank)

$PE(1,2) = \sin(1/10000^{1/3}) = 0.04$

$PE(1,3) = \cos(1/10000^{1/3}) = 0.99$

for i = 2 → pos = 1 (Bank)

$PE(1,4) = \sin(1/10000^{2/3}) = 0.00$

$PE(1,5) = \cos(1/10000^{2/3}) = 0.99$

Bank → | 0.84 | 0.54 | 0.04 | 0.99 | 0.00 | 0.99 |

→ Positional Encoding Vectors