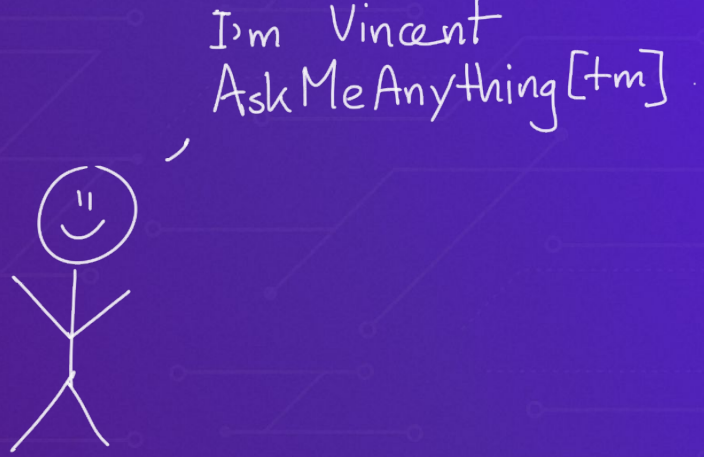
Why Transformers Work.



Think about all the actions in this dialogue →


Hello

Hi 


I'd like a pizza!

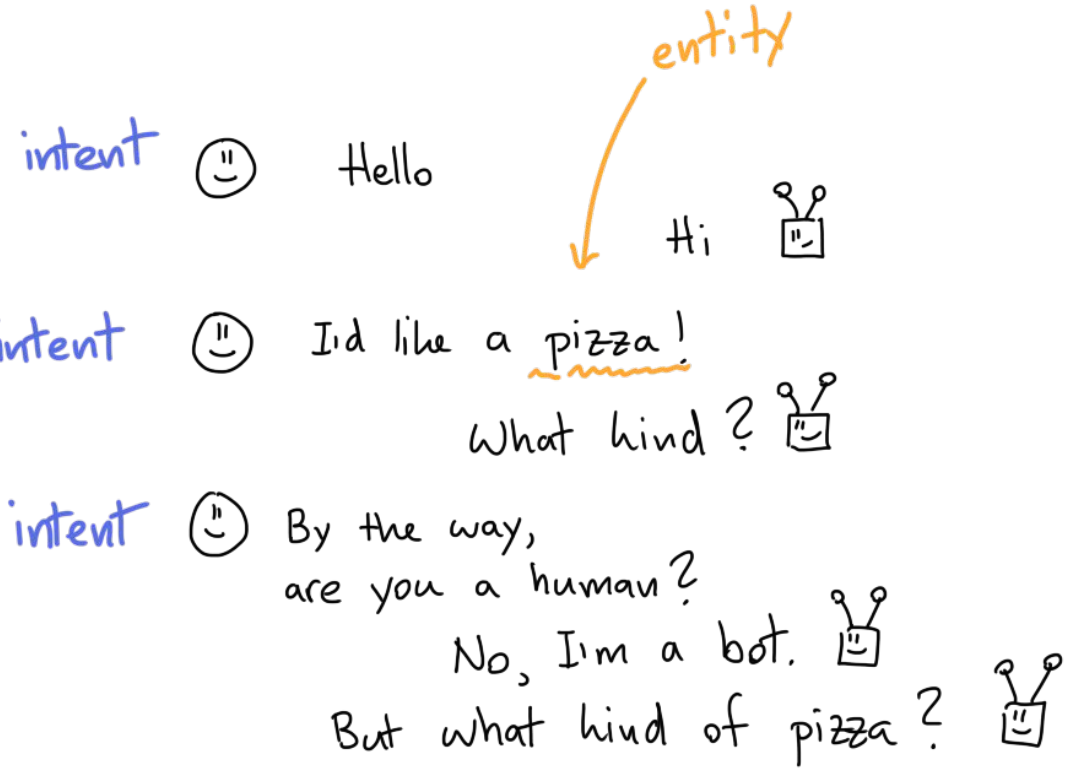
What kind? 


By the way,
are you a human?

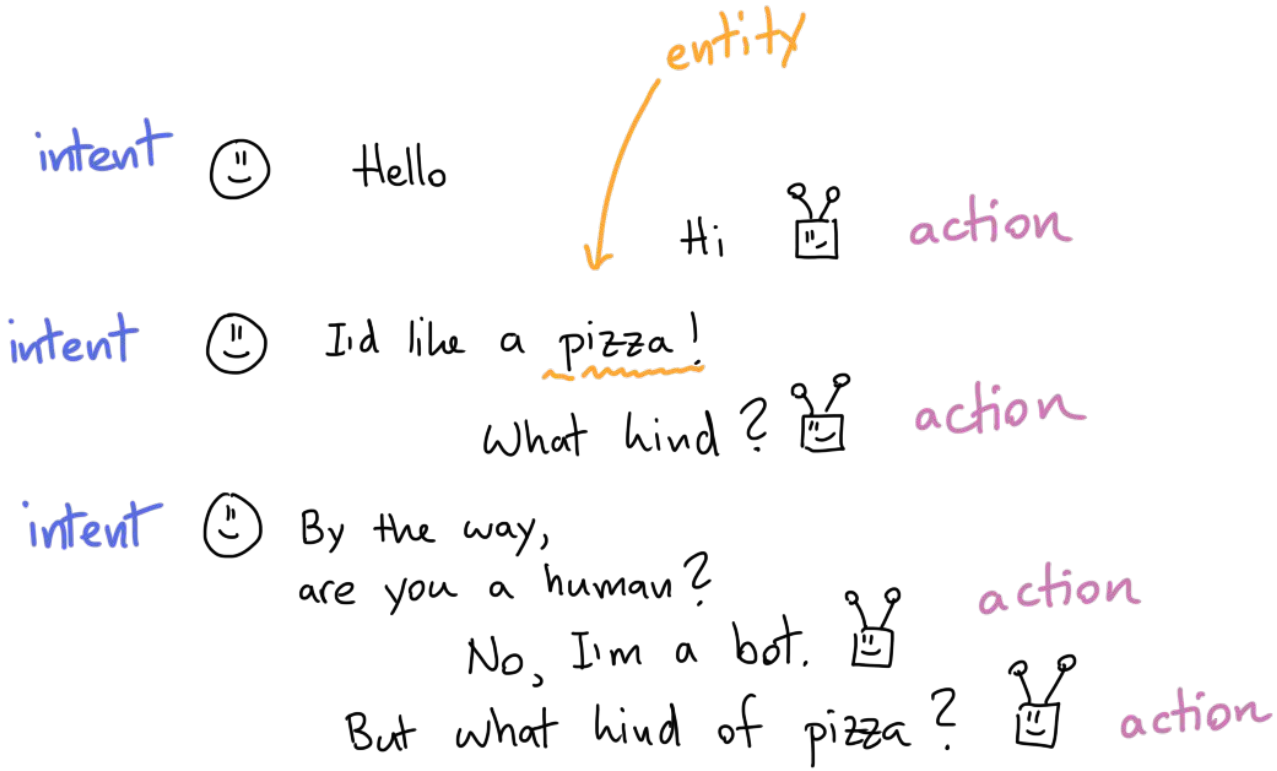
No, I'm a bot. 

But what kind of pizza? 

*More info blablabla



*More info blablabla



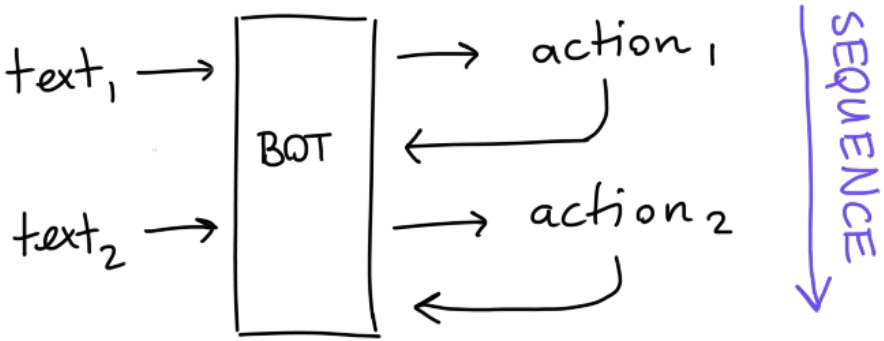
*More info blablabla

I wanna buy a pizza.

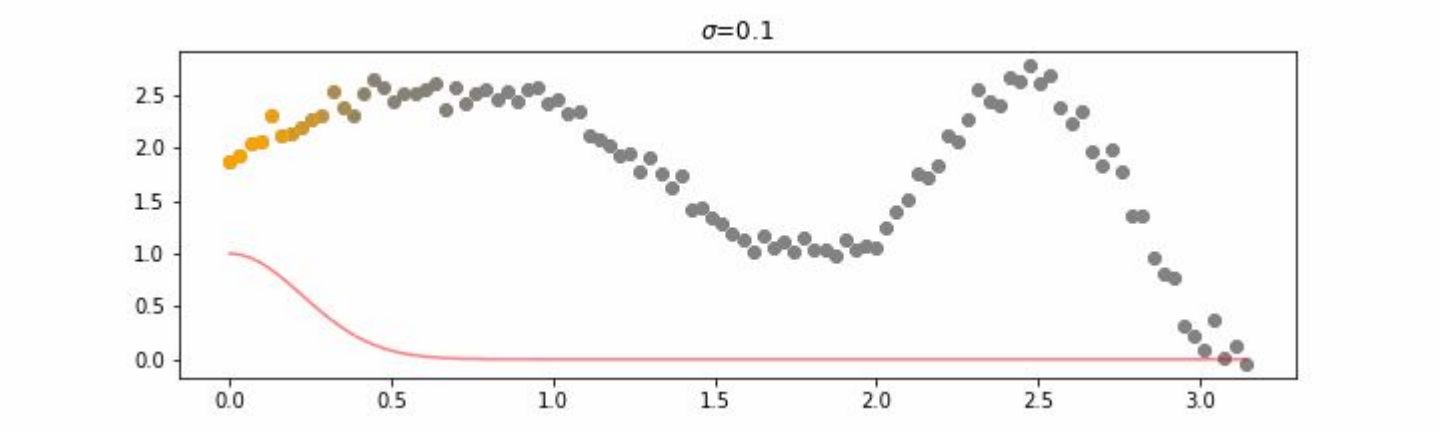
*More info blablabla

I wanna buy a pizza.

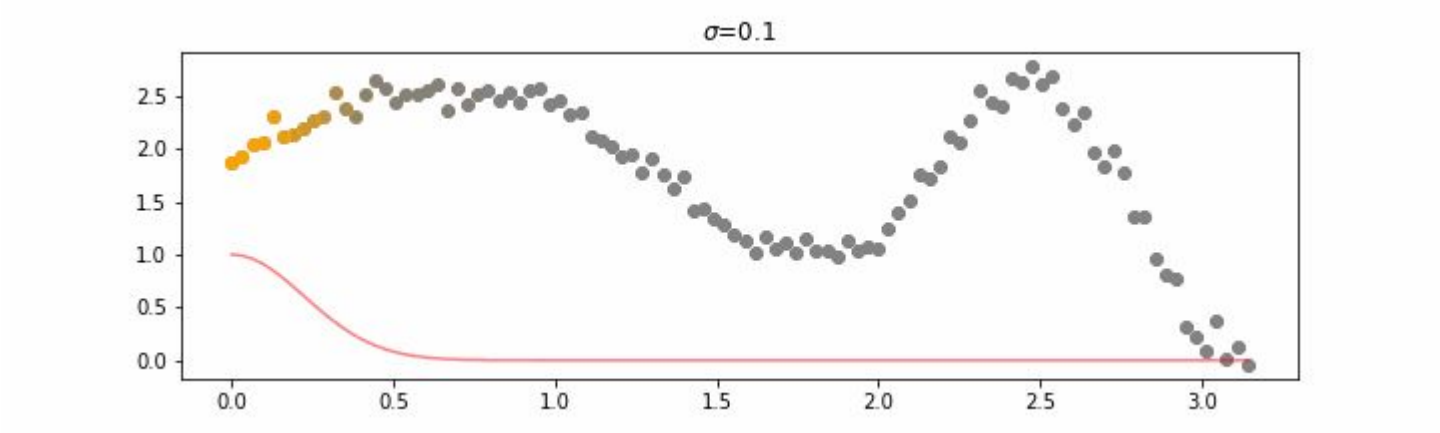
SEQUENCE

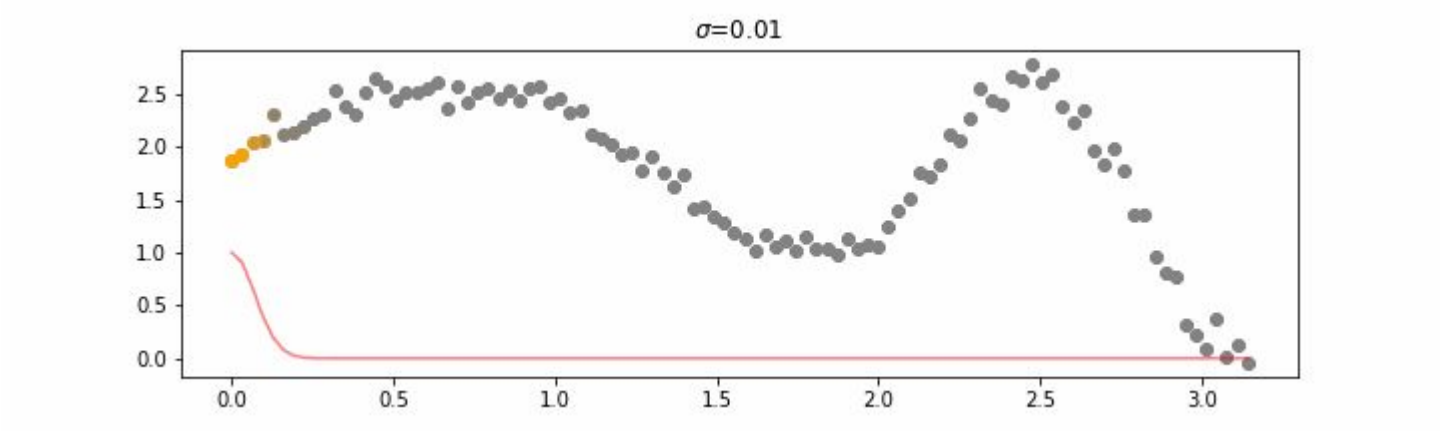


*More info blablabla

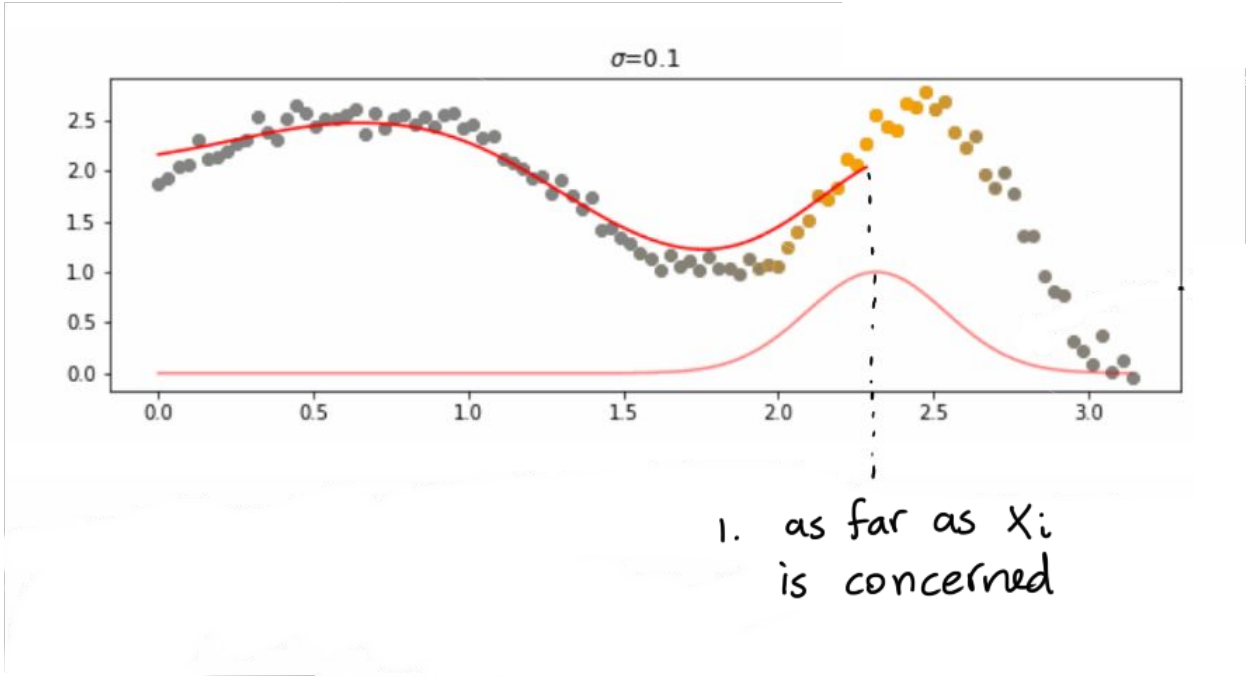


*More info blablabla

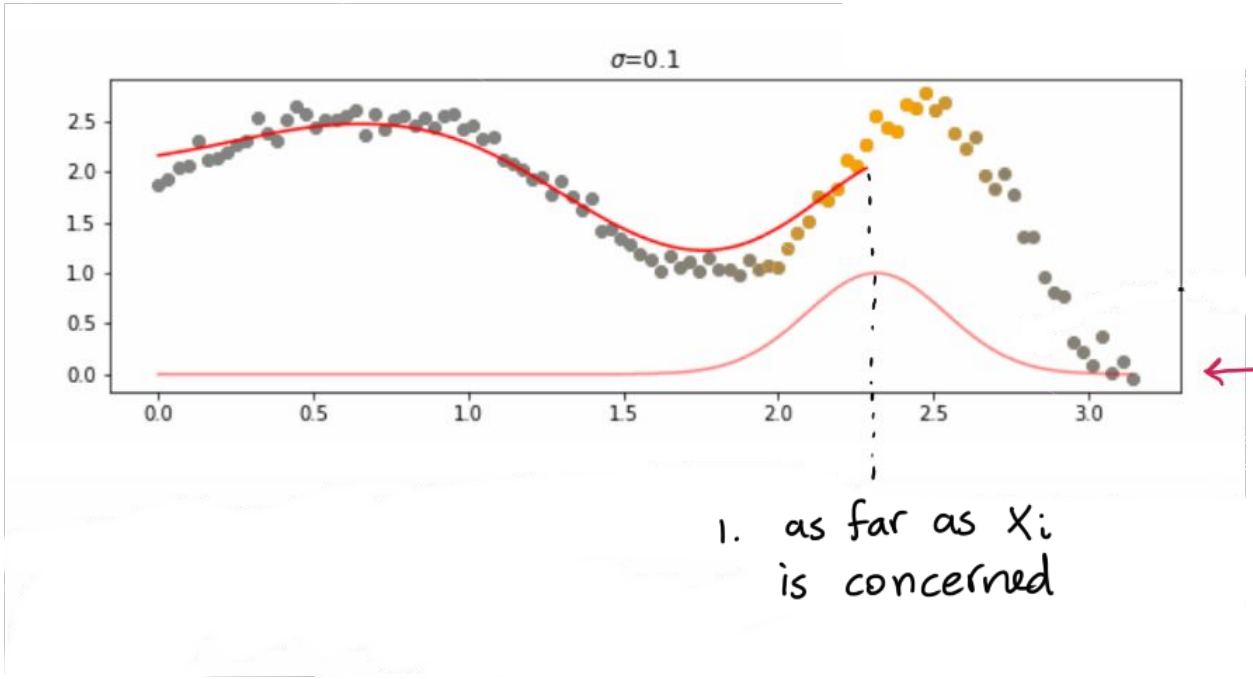




*More info blablabla



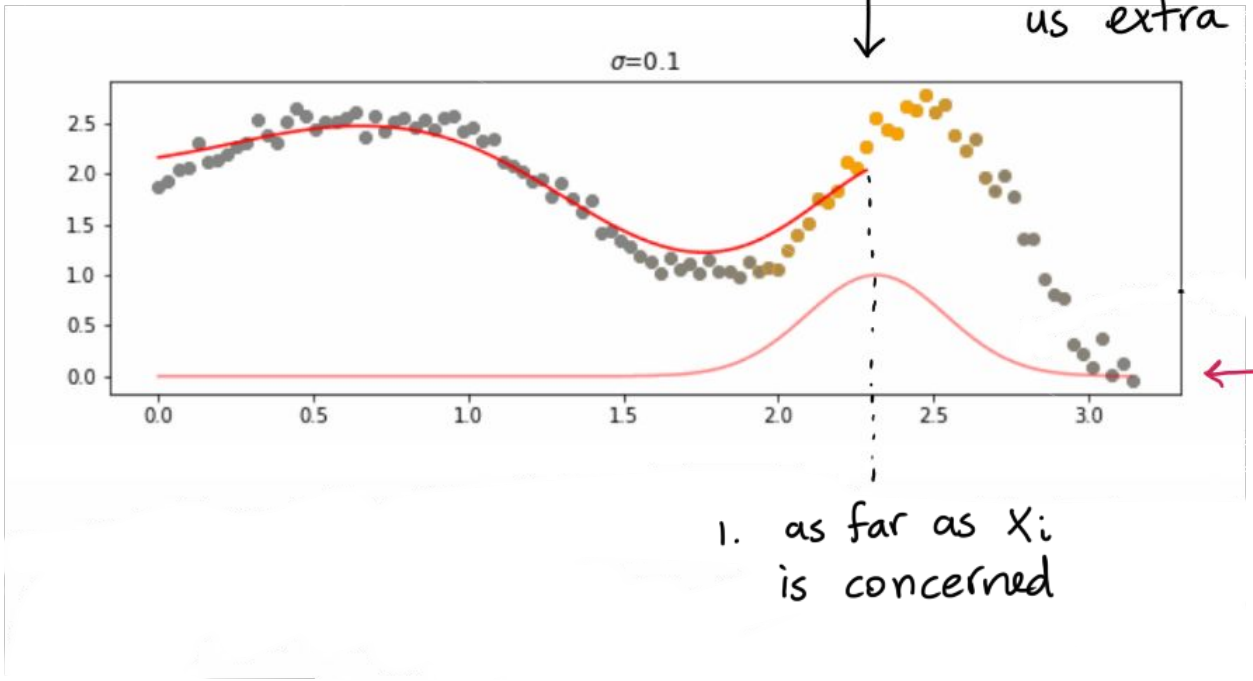
*More info blablabla



1. as far as X_i is concerned

2. this is what we need to pay attention to

*More info blablabla



3. this allows us to re-weigh and this operation hopefully gives us extra context

1. as far as X_i is concerned

2. this is what we need to pay attention to

*More info blablabla

Bank of the river.

t_1	t_2	t_3	t_4
V_1	V_2	V_3	V_4

Money on the bank.

t_1	t_2	t_3	t_4
V_1	V_2	V_3	V_4

*More info blablabla

Bank of the river.

t_1	t_2	t_3	t_4
v_1	v_2	v_3	v_4

Money on the bank.

t_1	t_2	t_3	t_4
v_1	v_2	v_3	v_4

If v_1 and v_2 share info

$v_1 \cdot v_2$ is big

If v_1 and v_2 don't

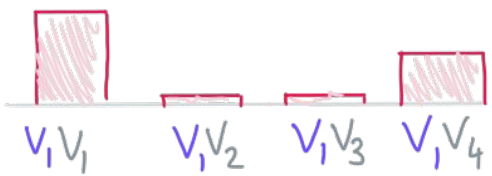
$v_1 \cdot v_2 \approx 0$

*More info blablabla

Bank of the river.

t_1 t_2 t_3 t_4

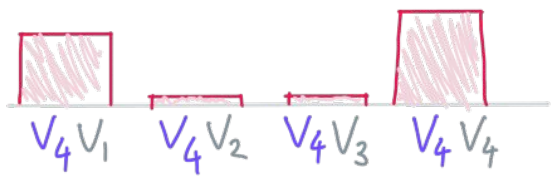
V_1 V_2 V_3 V_4



Money on the bank.

t_1 t_2 t_3 t_4

V_1 V_2 V_3 V_4

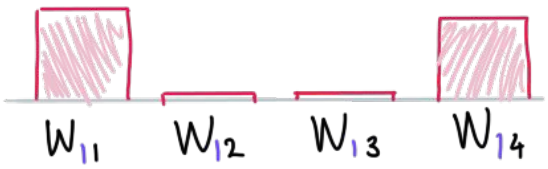


*More info blablabla

Bank of the river.

t_1 t_2 t_3 t_4

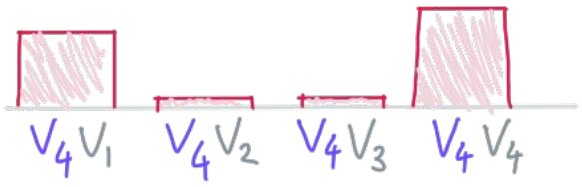
V_1 V_2 V_3 V_4

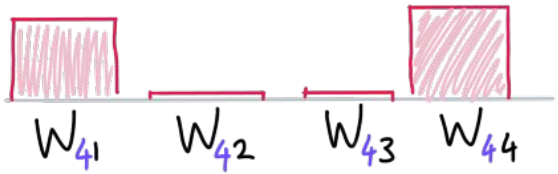


Money on the bank.

t_1 t_2 t_3 t_4

V_1 V_2 V_3 V_4





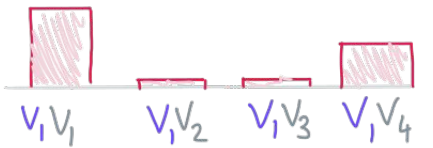
NORMALISE

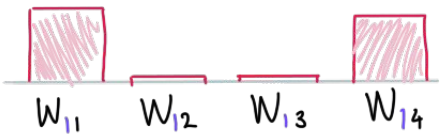
*More info blablabla

Bank of the river.

t_1 t_2 t_3 t_4

V_1 V_2 V_3 V_4



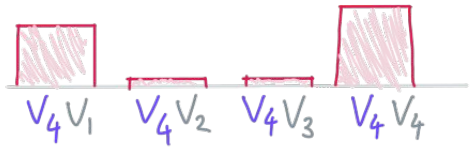


$$V_1^* = \sum_j W_{1j} V_j$$

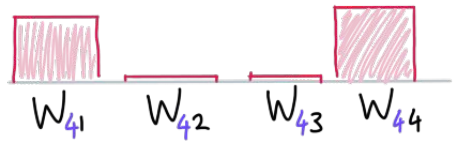
Money on the bank.

t_1 t_2 t_3 t_4

V_1 V_2 V_3 V_4



NORMALISE



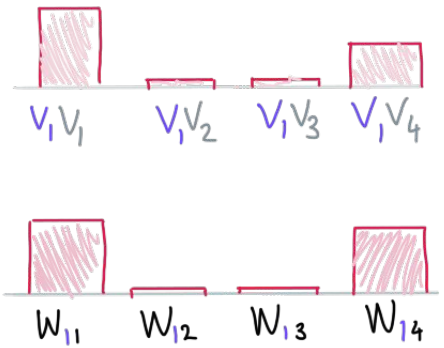
$$V_4^* = \sum_j W_{4j} V_j$$

*More info blablabla

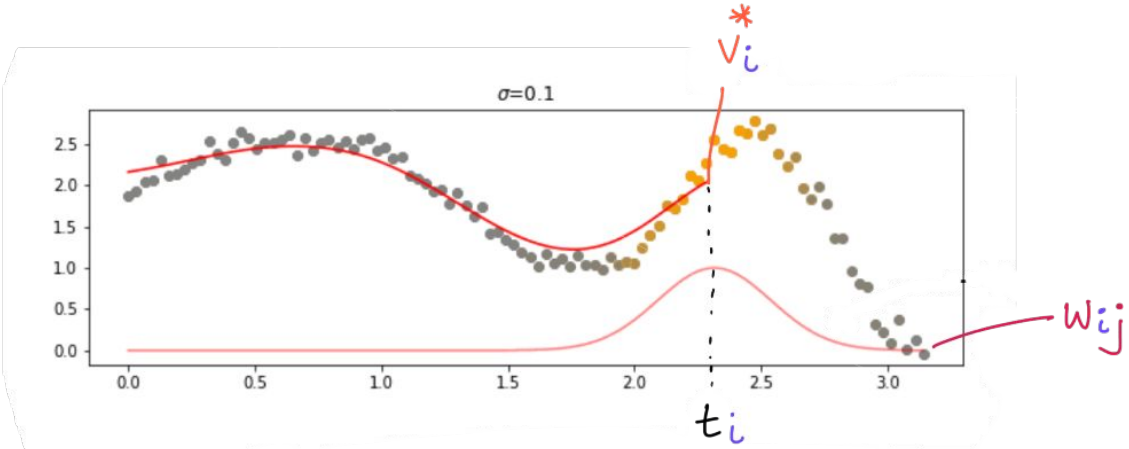
Bank of the river.

t_1 t_2 t_3 t_4

v_1 v_2 v_3 v_4



$$v_1^* = \sum_j w_{1j} v_j$$

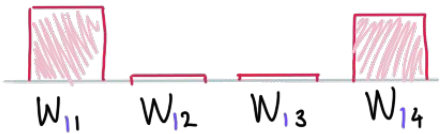


*More info blablabla

Bank of the river.

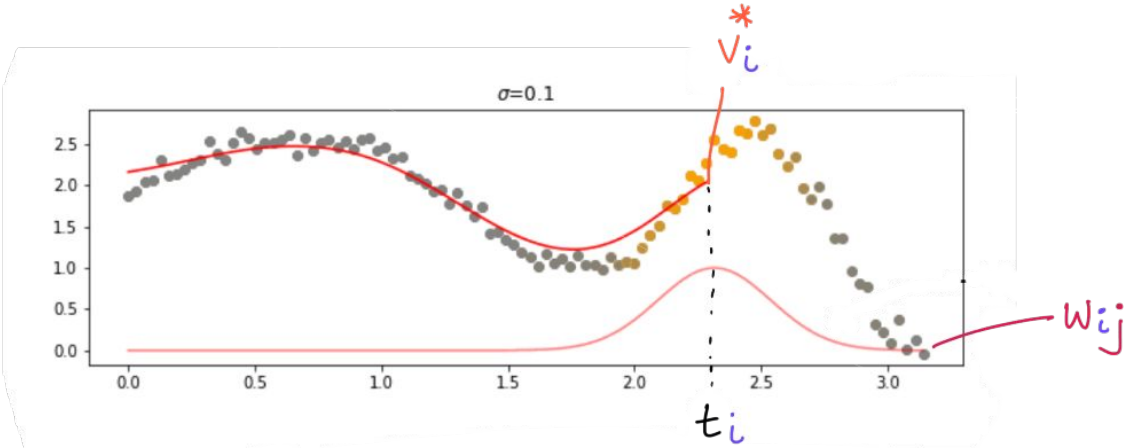
t_1 t_2 t_3 t_4

v_1 v_2 v_3 v_4



$$v_1^* = \sum_j w_{1j} v_j$$

reweighing based on time-distance

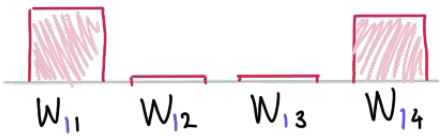


*More info blablabla

Bank of the river.

t_1 t_2 t_3 t_4

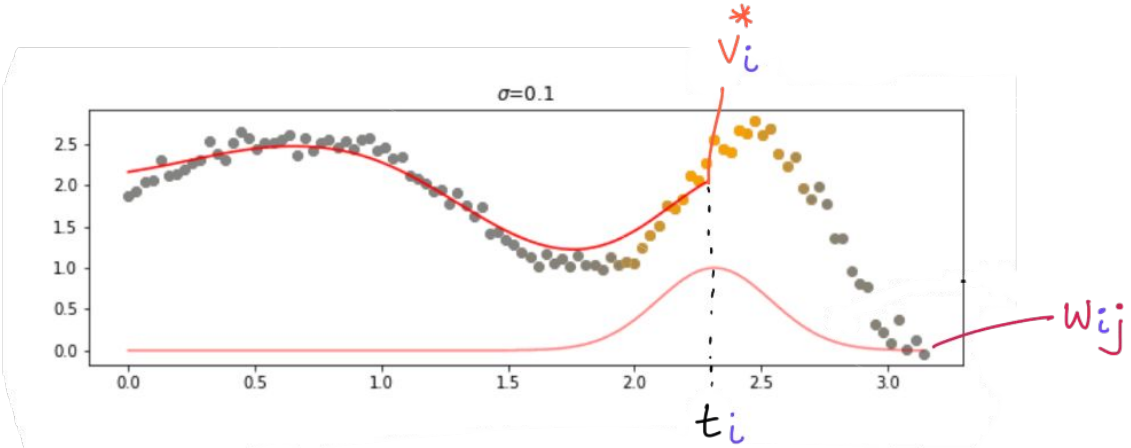
v_1 v_2 v_3 v_4



$$v_1^* = \sum_j w_{1j} v_j$$

reweighing based on embedding similarity

reweighing based on time-distance

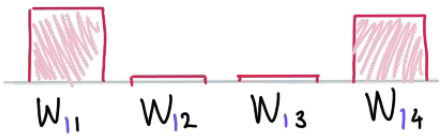


*More info blablabla

Bank of the river.

t_1 t_2 t_3 t_4

V_1 V_2 V_3 V_4

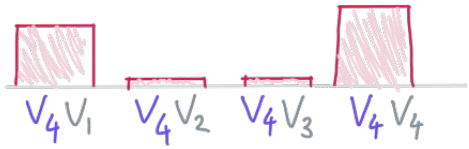


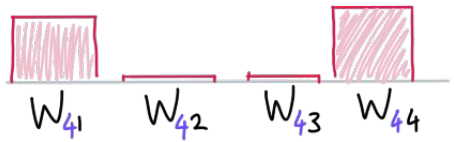
$$v_1^* = \sum_j w_{1j} v_j$$

Money on the bank.

t_1 t_2 t_3 t_4

V_1 V_2 V_3 V_4

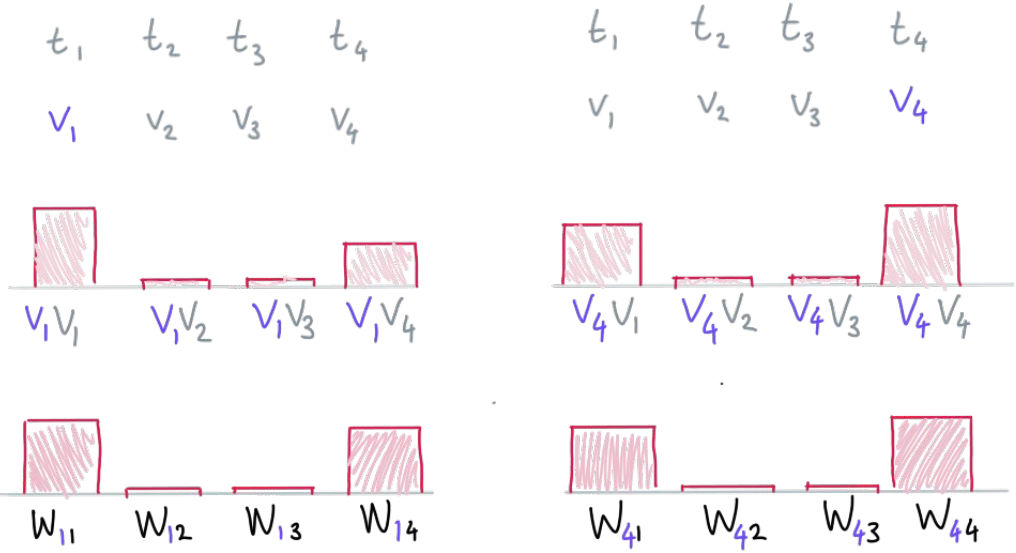




$$v_4^* = \sum_j w_{4j} v_j$$

*More info blablabla

Bank of the river.



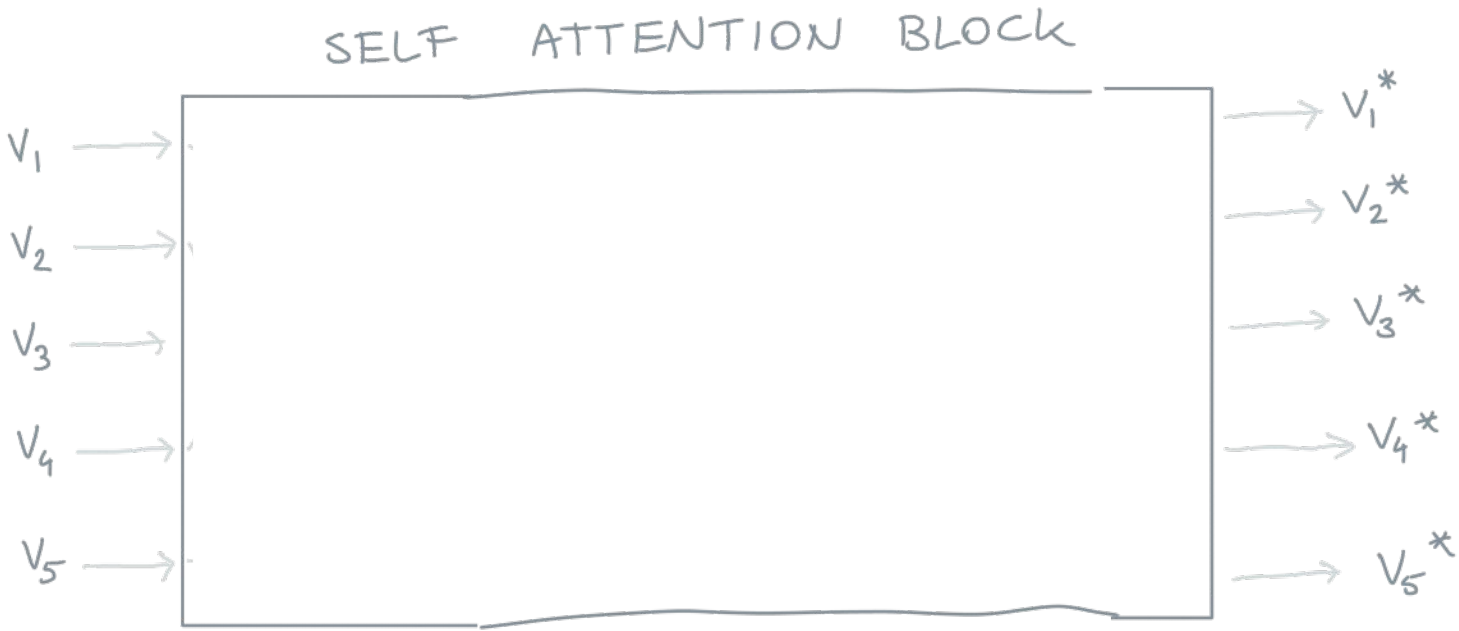
$$v_1^* = \sum_j w_{1j} v_j$$

Money on the bank.

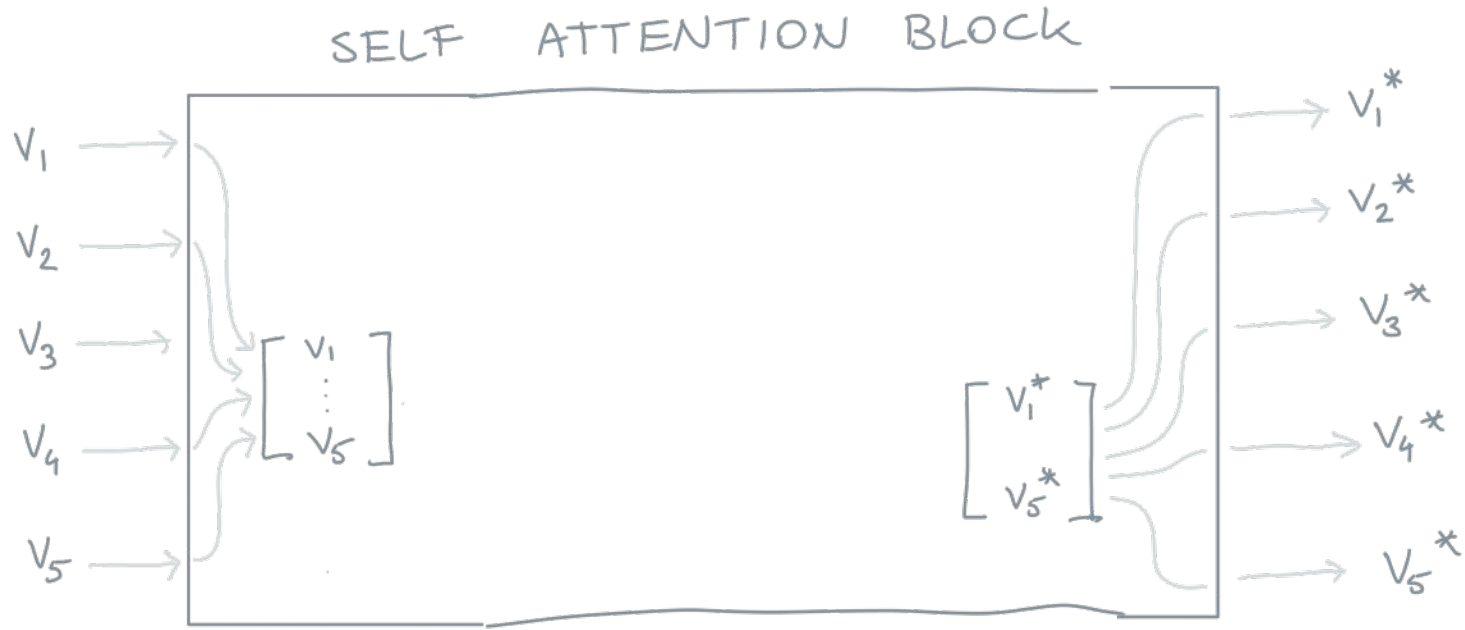
$$v_4^* = \sum_j w_{4j} v_j$$

Let's move this idea into a layer.

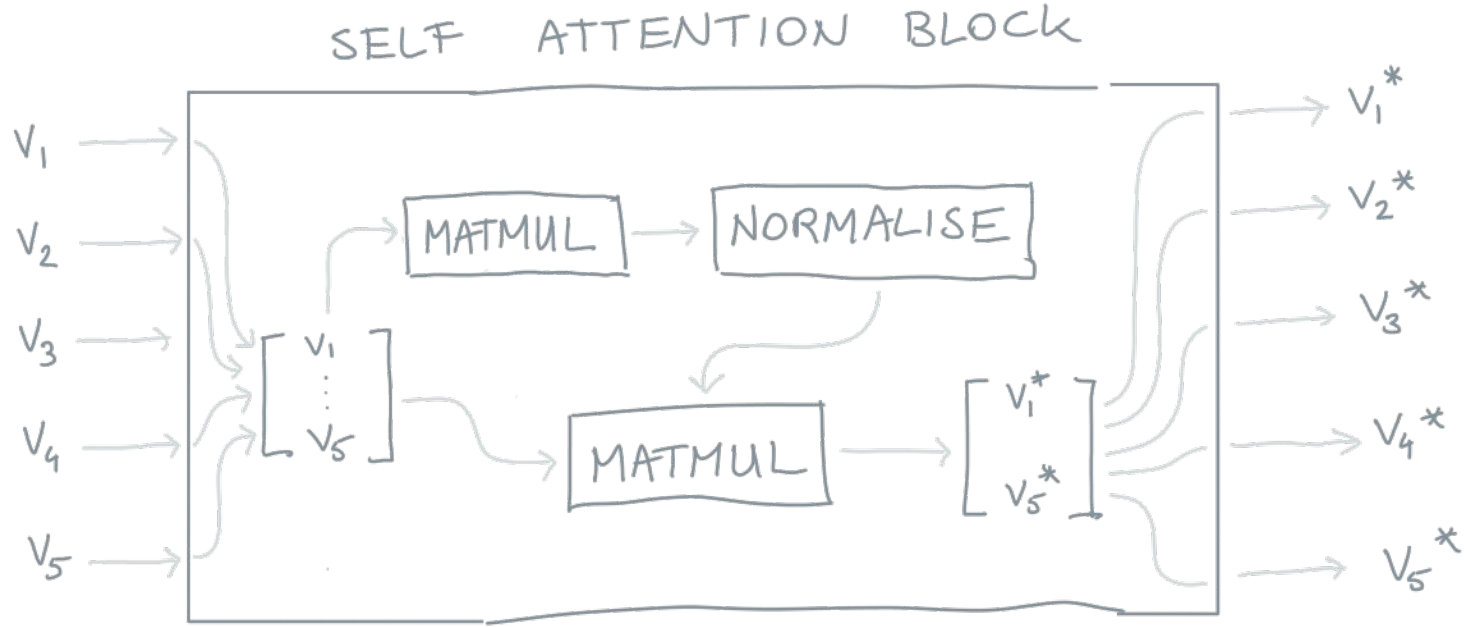
*More info blablabla



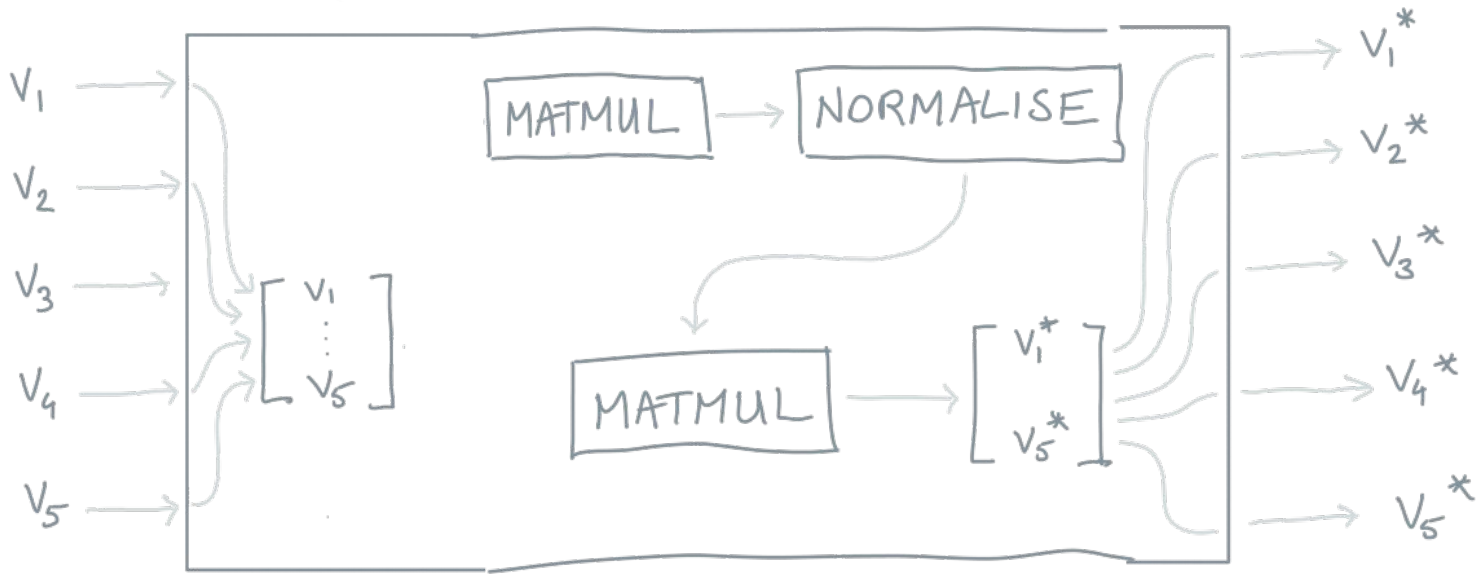
*More info blablabla



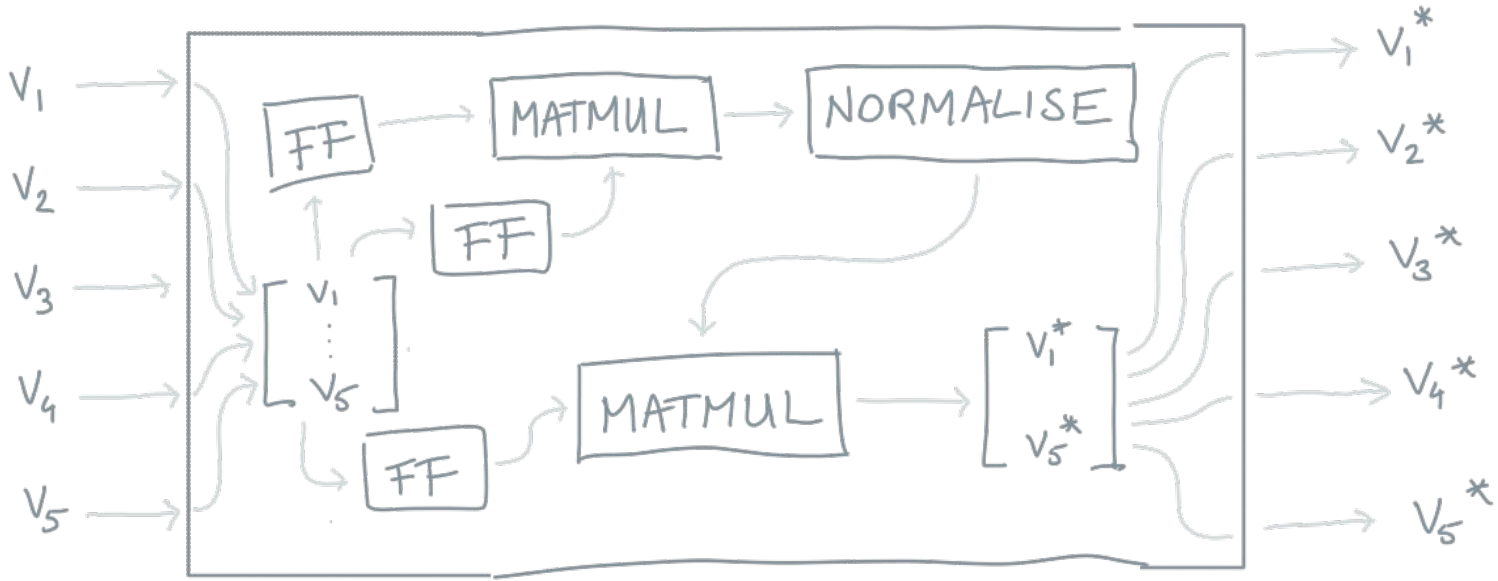
*More info blablabla



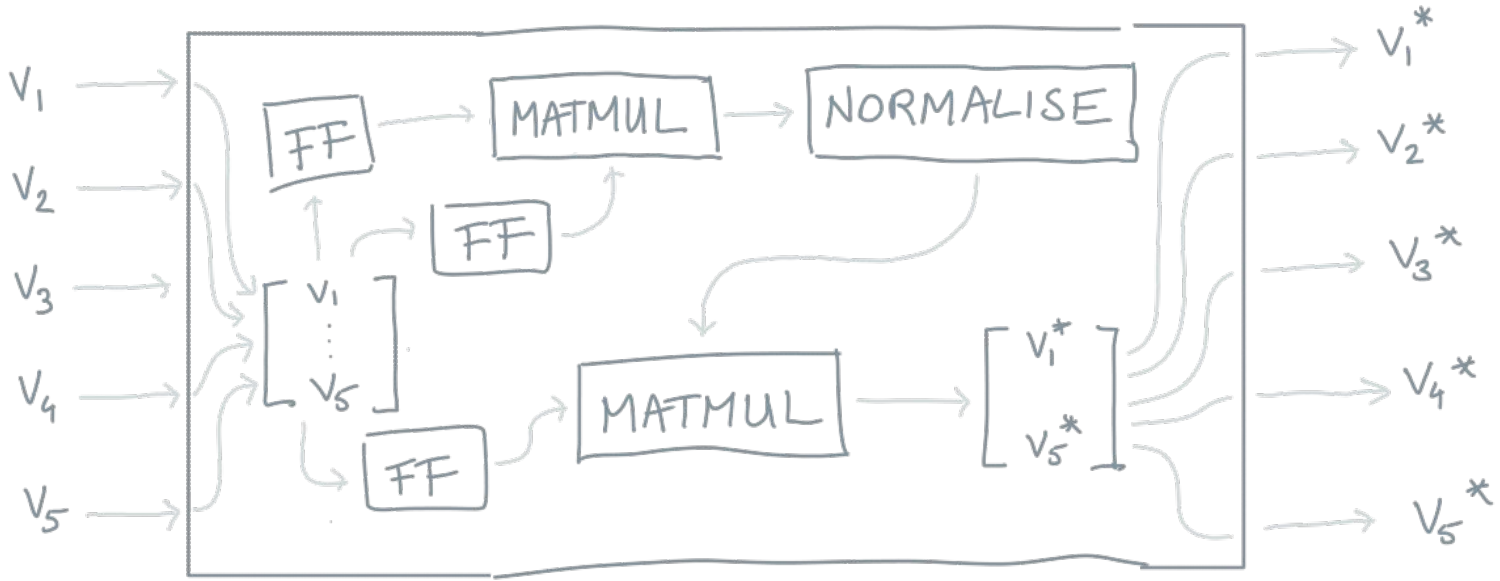
*More info blablabla



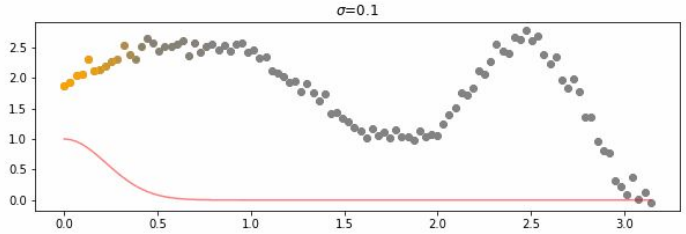
*More info blablabla



*More info blablabla

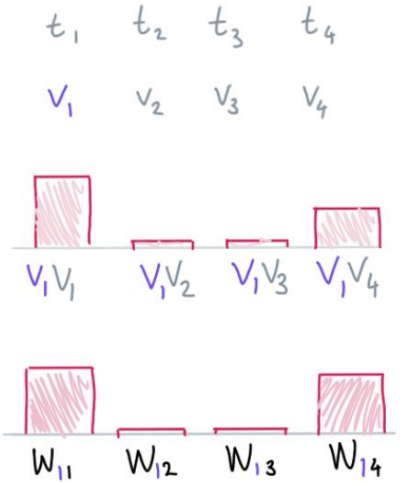


It's a more elaborate way to do stuff like this →

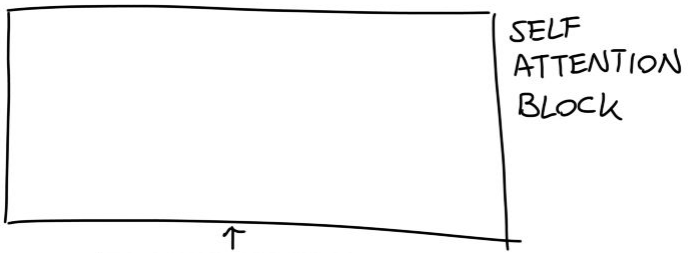


*More info blablabla

Bank of the river.

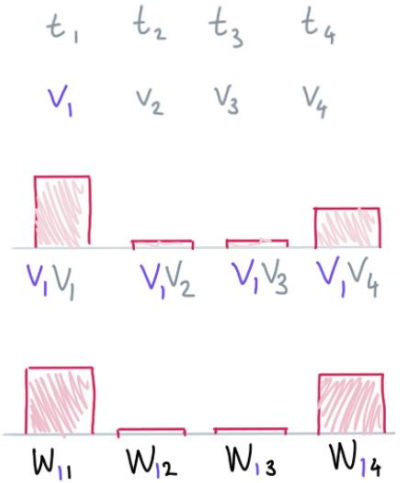


$$v_1^* = \sum_j w_{1j} v_j$$

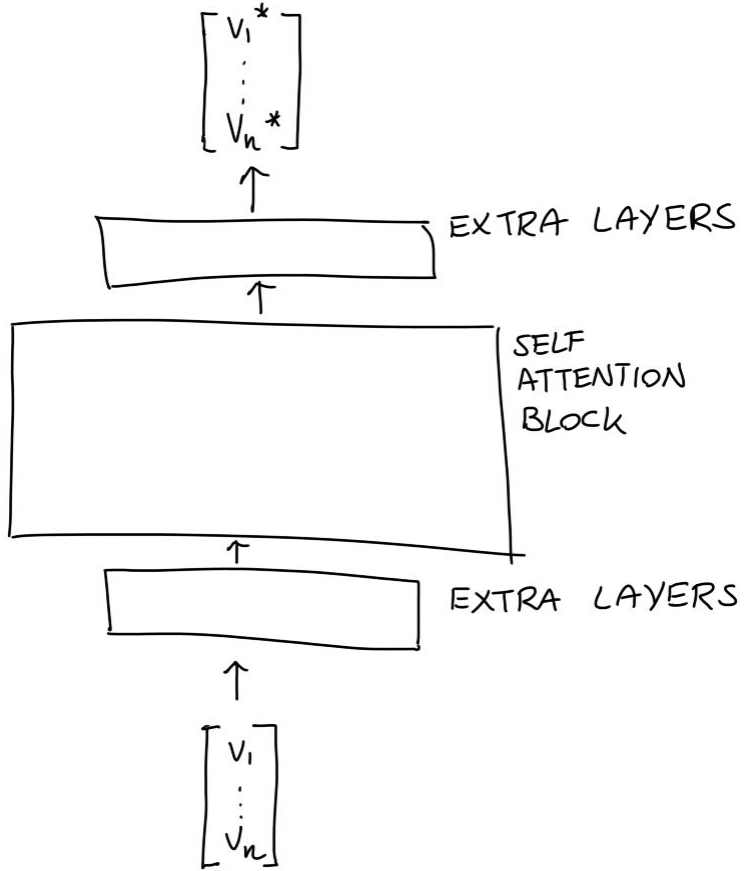


*More info blablabla

Bank of the river.

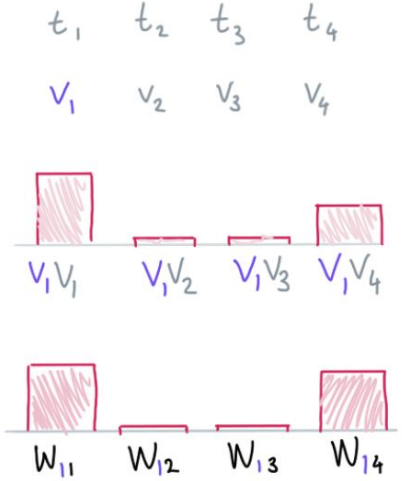


$$V_1^* = \sum_j W_{1j} V_j$$

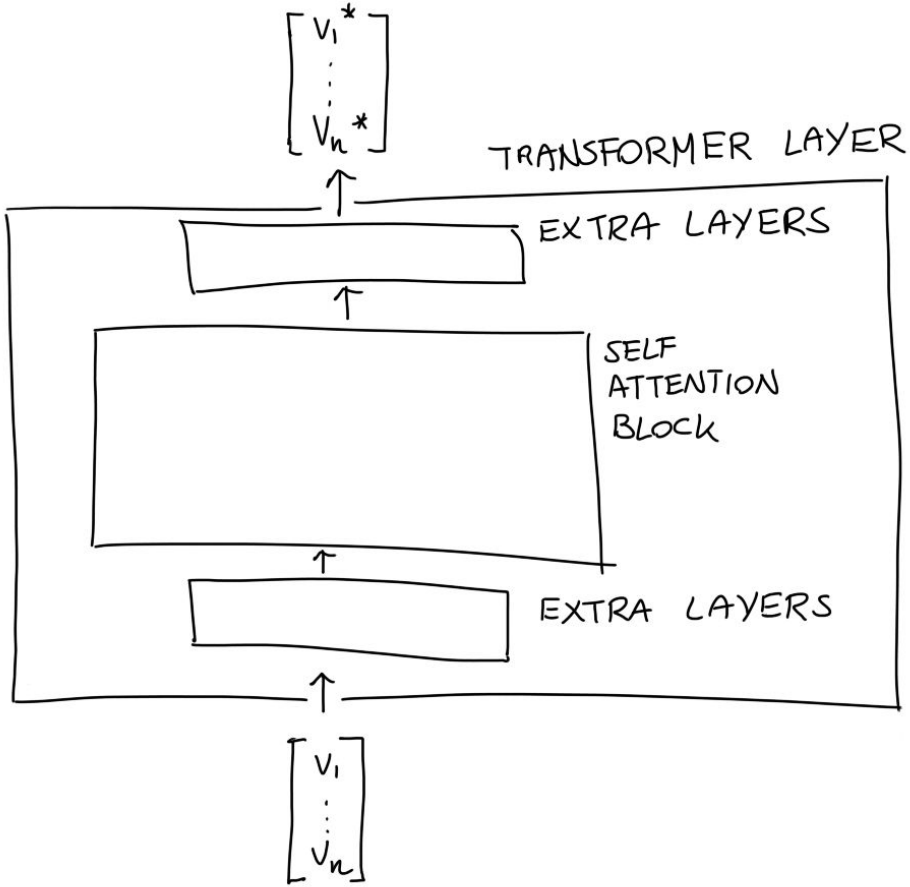


*More info blablabla

Bank of the river.

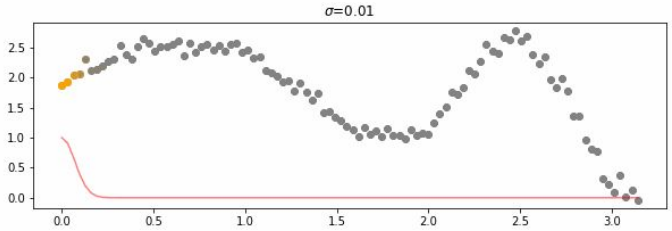


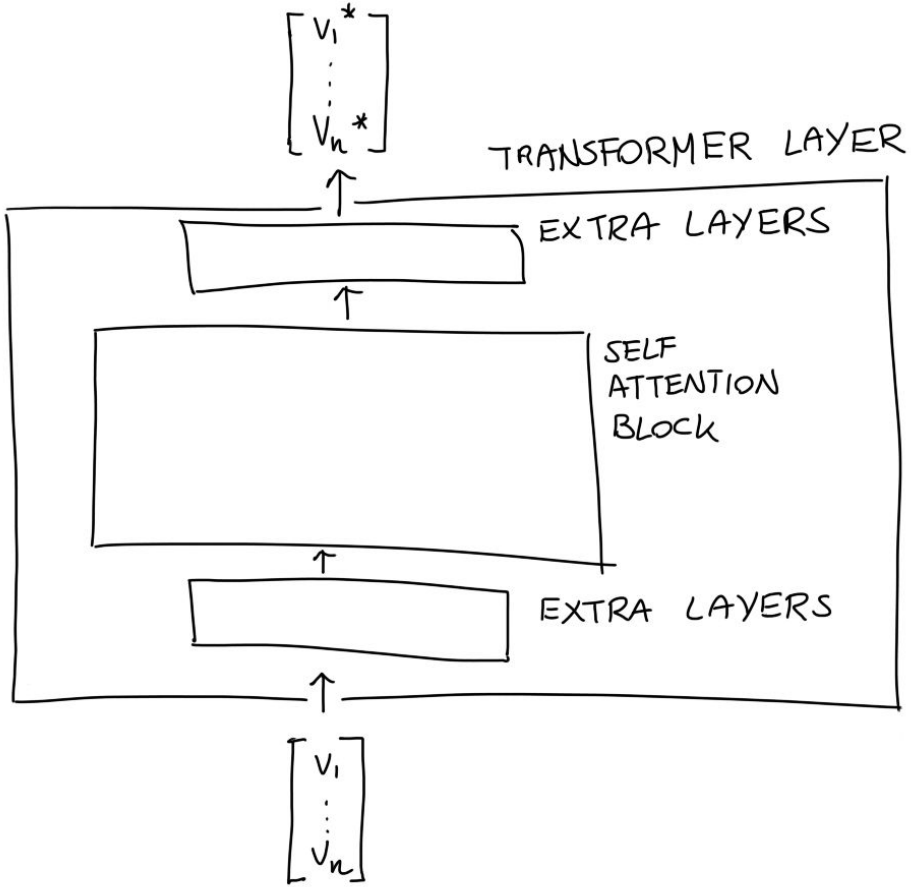
$$V_1^* = \sum_j W_{1j} V_j$$



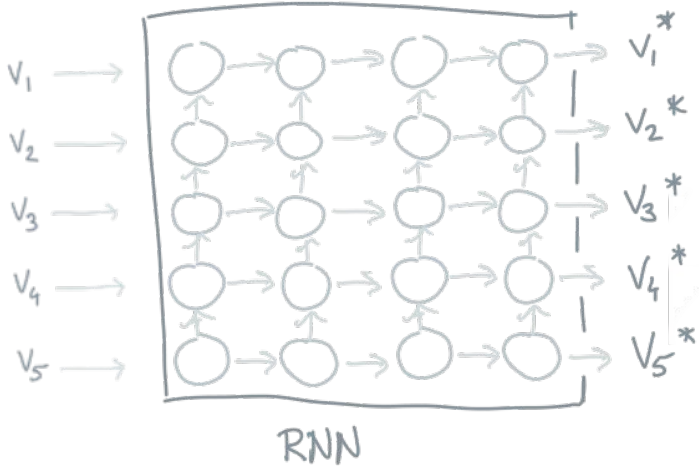
*More info blablabla

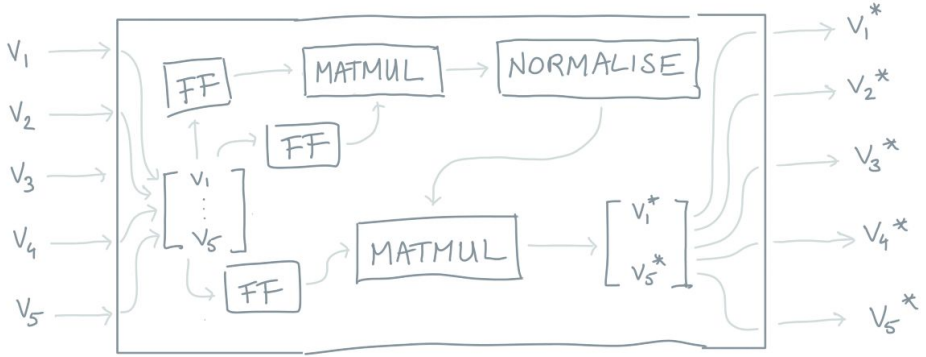
*glancing over some details
the goal here is intuition
in the interest of time



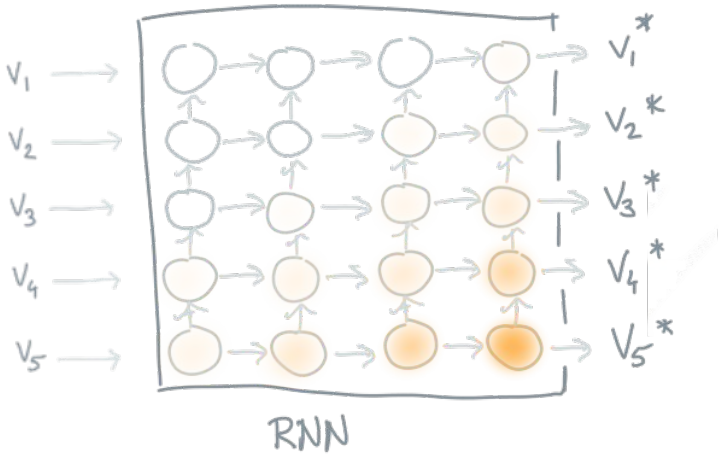


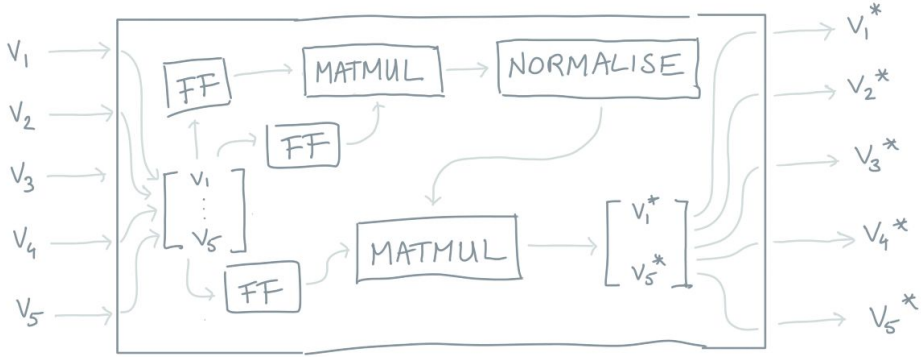
*More info blablabla



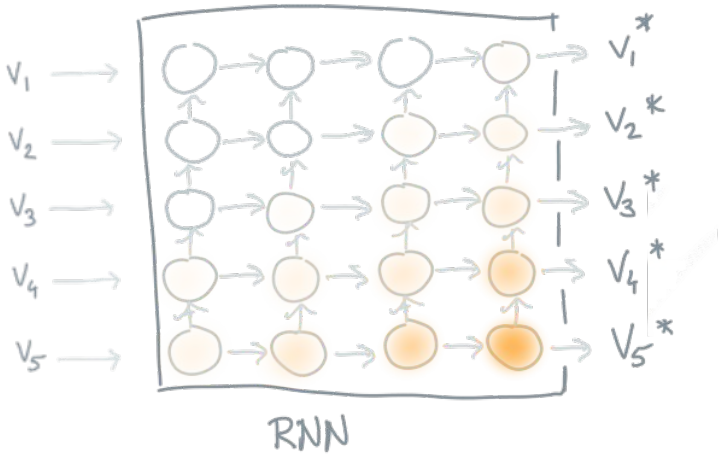


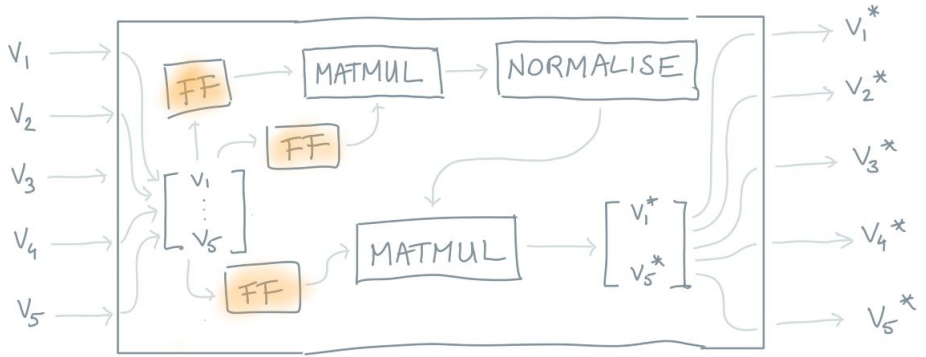
*More info blablabla





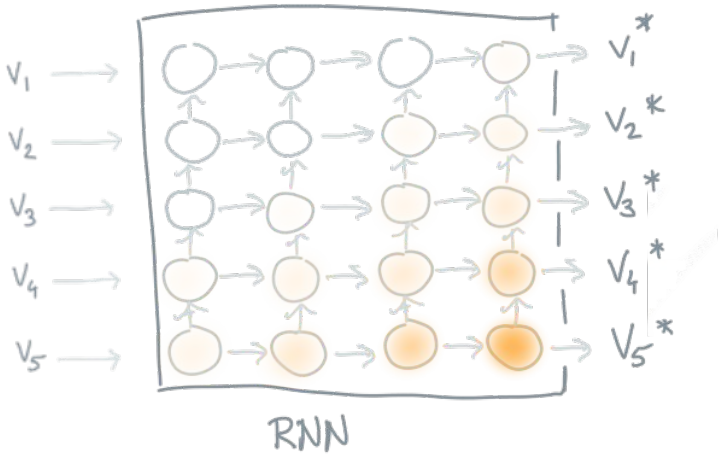
*More info blablabla





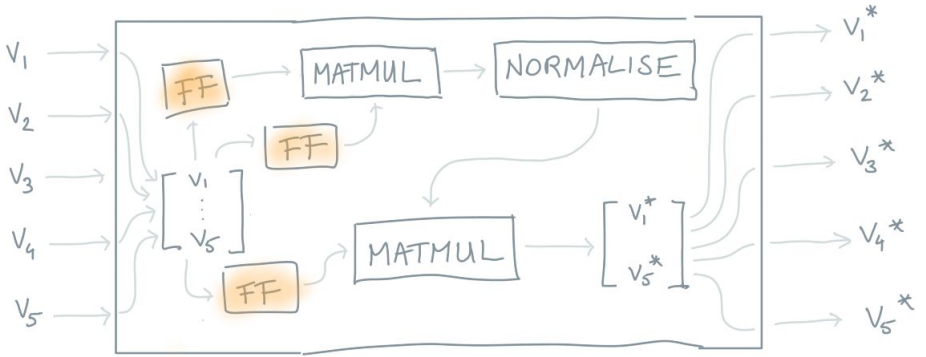
*More info blablabla

Needs a lot of data to stop looking at its previous neighbor.



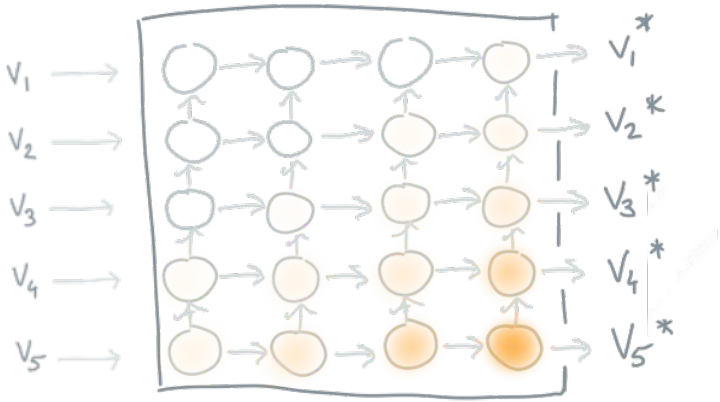
RNN

Starts out by looking at similarity of pretrained embeddings.



*More info blablabla

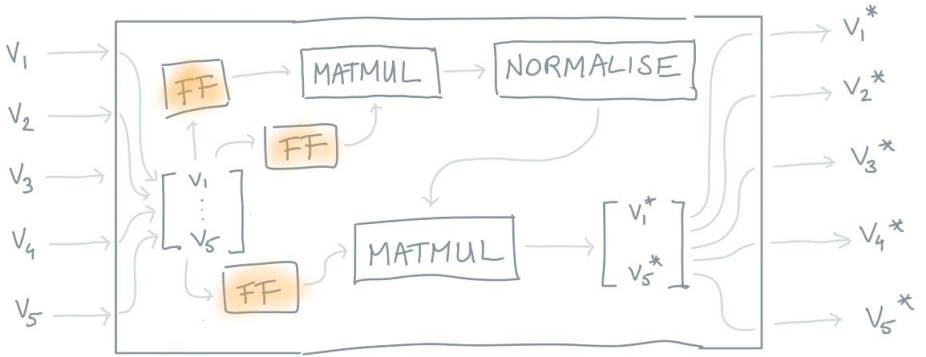
Needs a lot of data to stop looking at its previous neighbor.



RNN

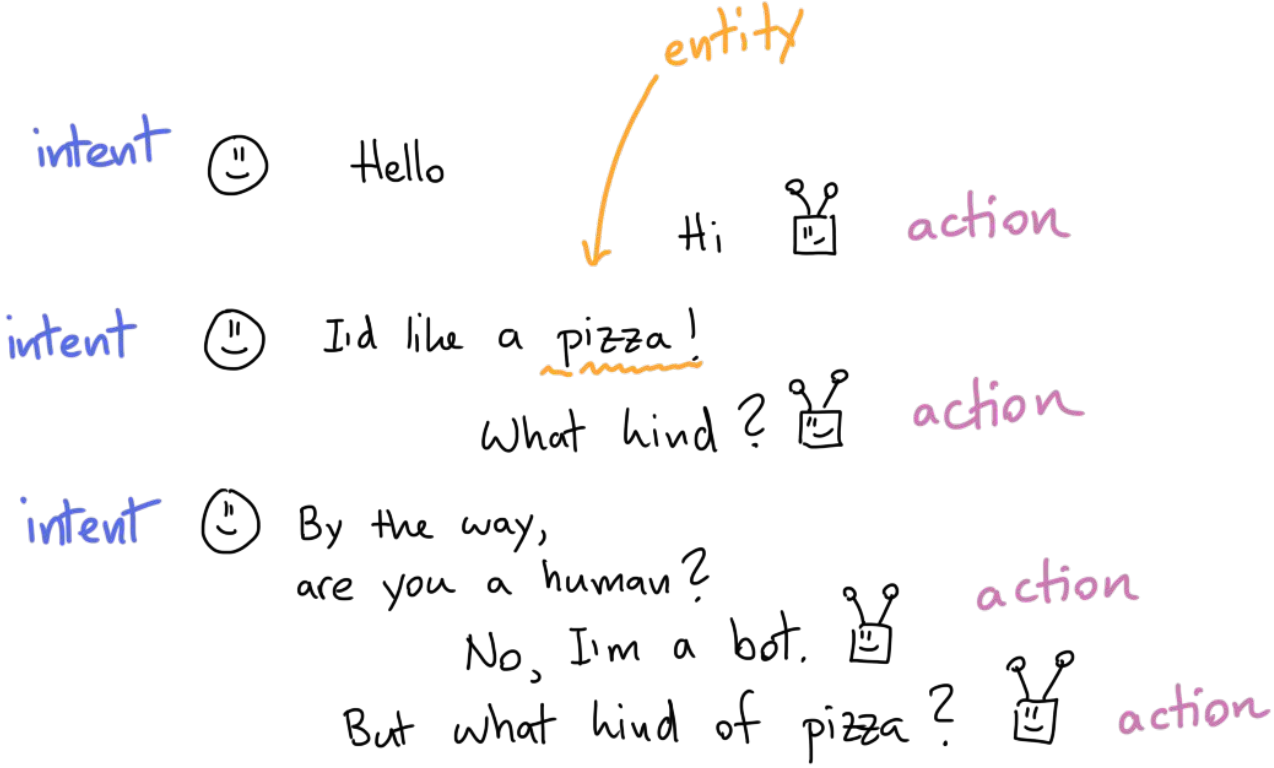
Hard to parallise.

Starts out by looking at similarity of pretrained embeddings.



More parallel options.

*More info blablabla



*More info blablabla

Q: So how does Rasa find these intents and entities?

A: DIET

Dual Intent & Entity Transformer

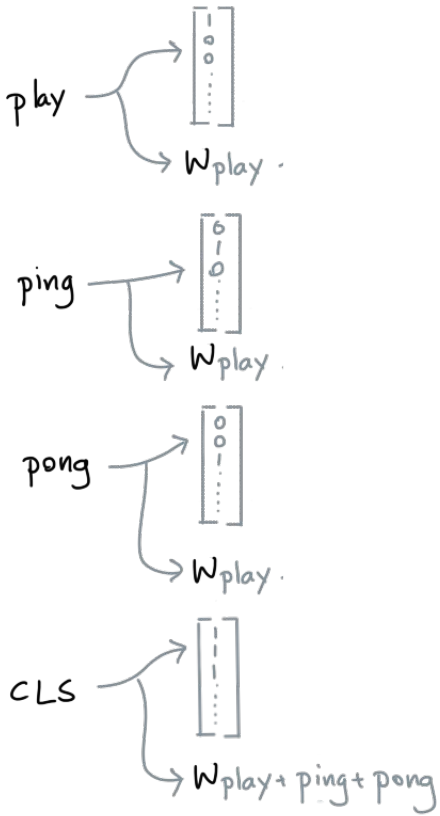
*More info blablabla

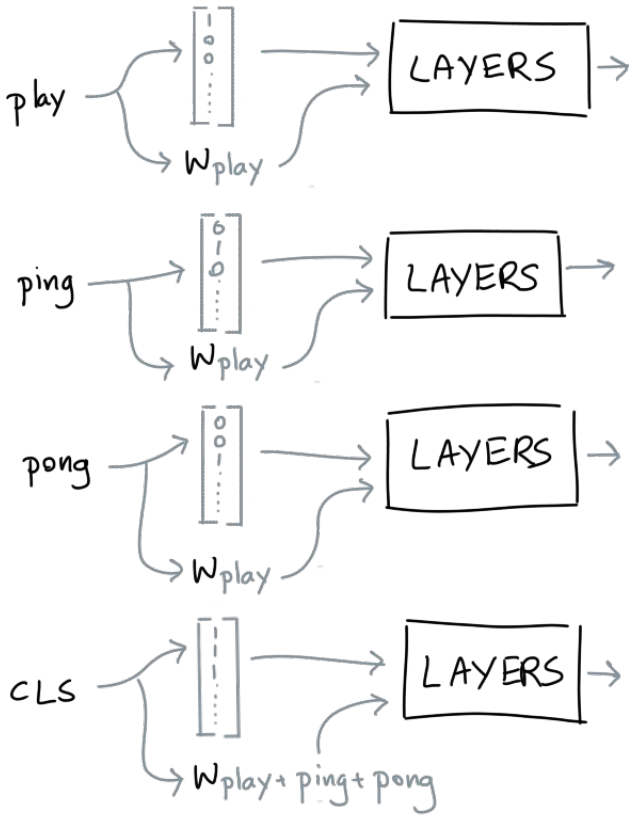
play –

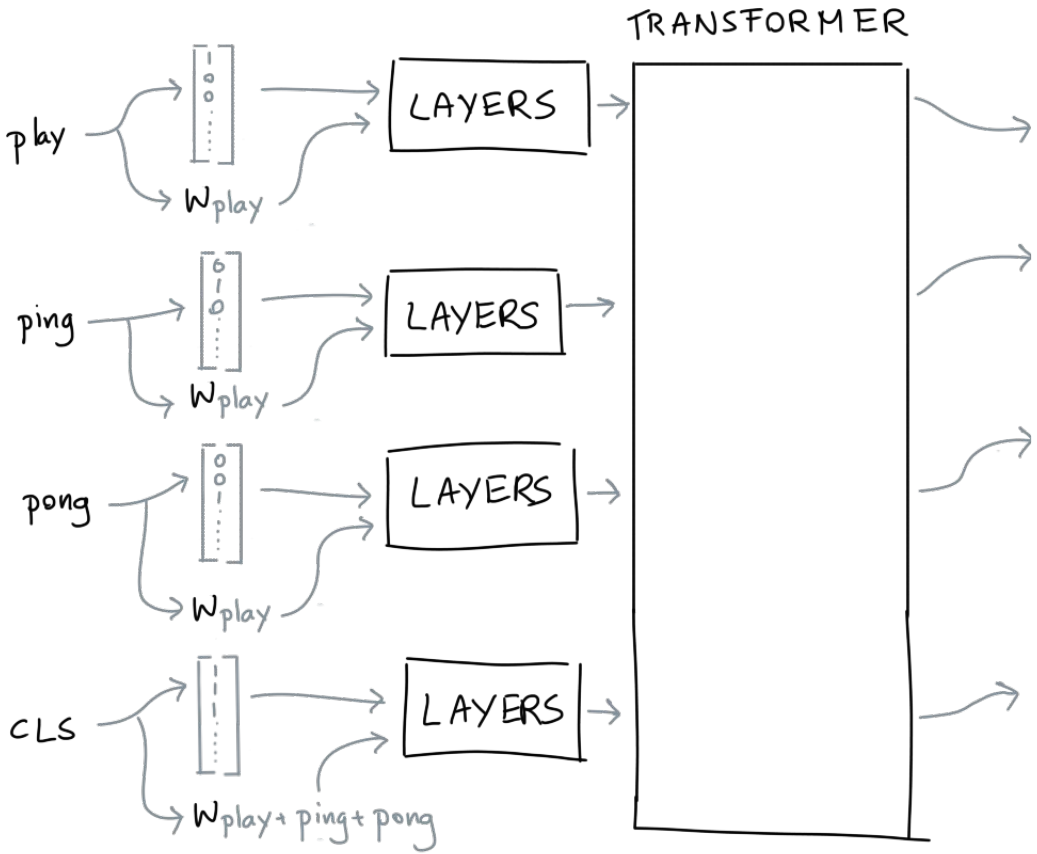
ping –

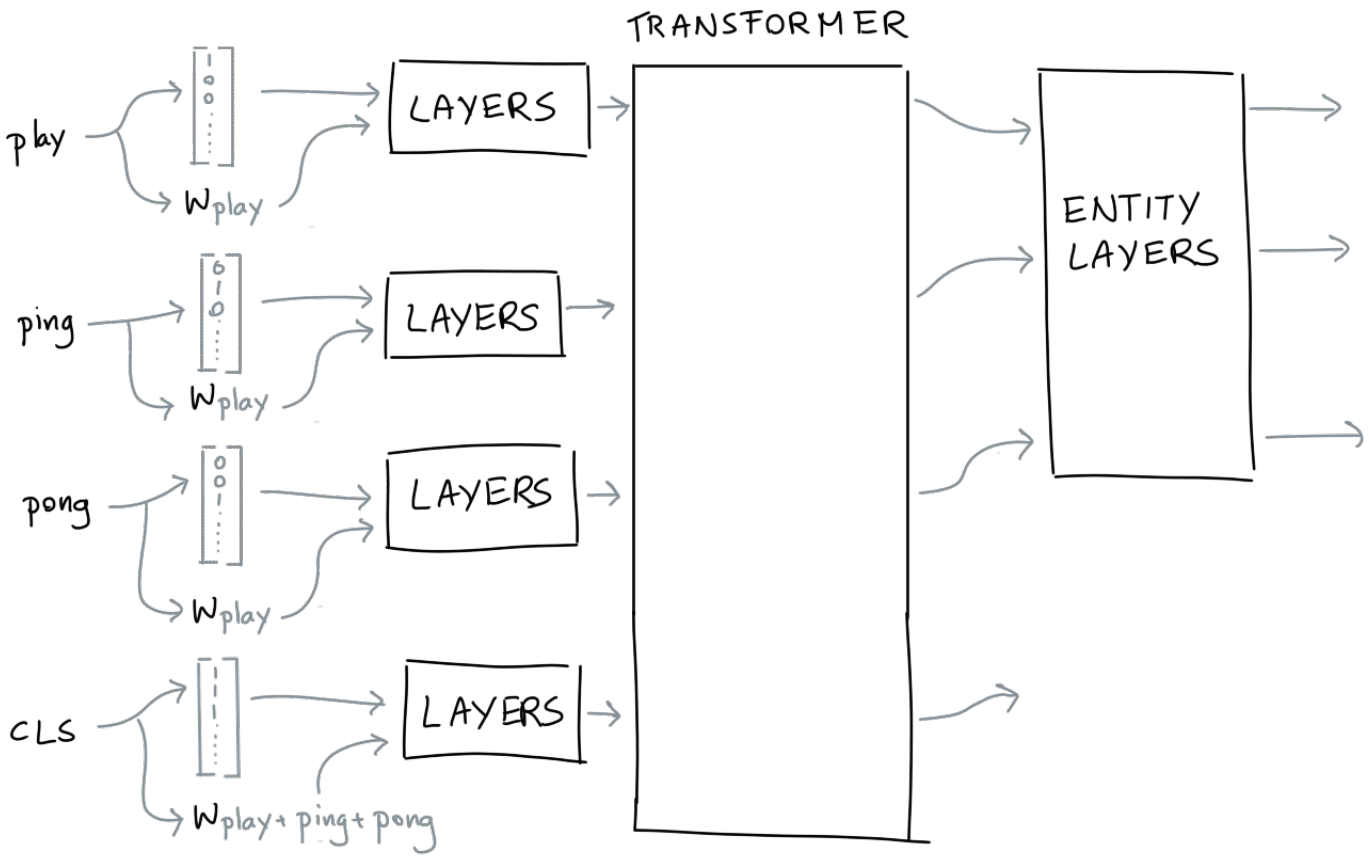
pong

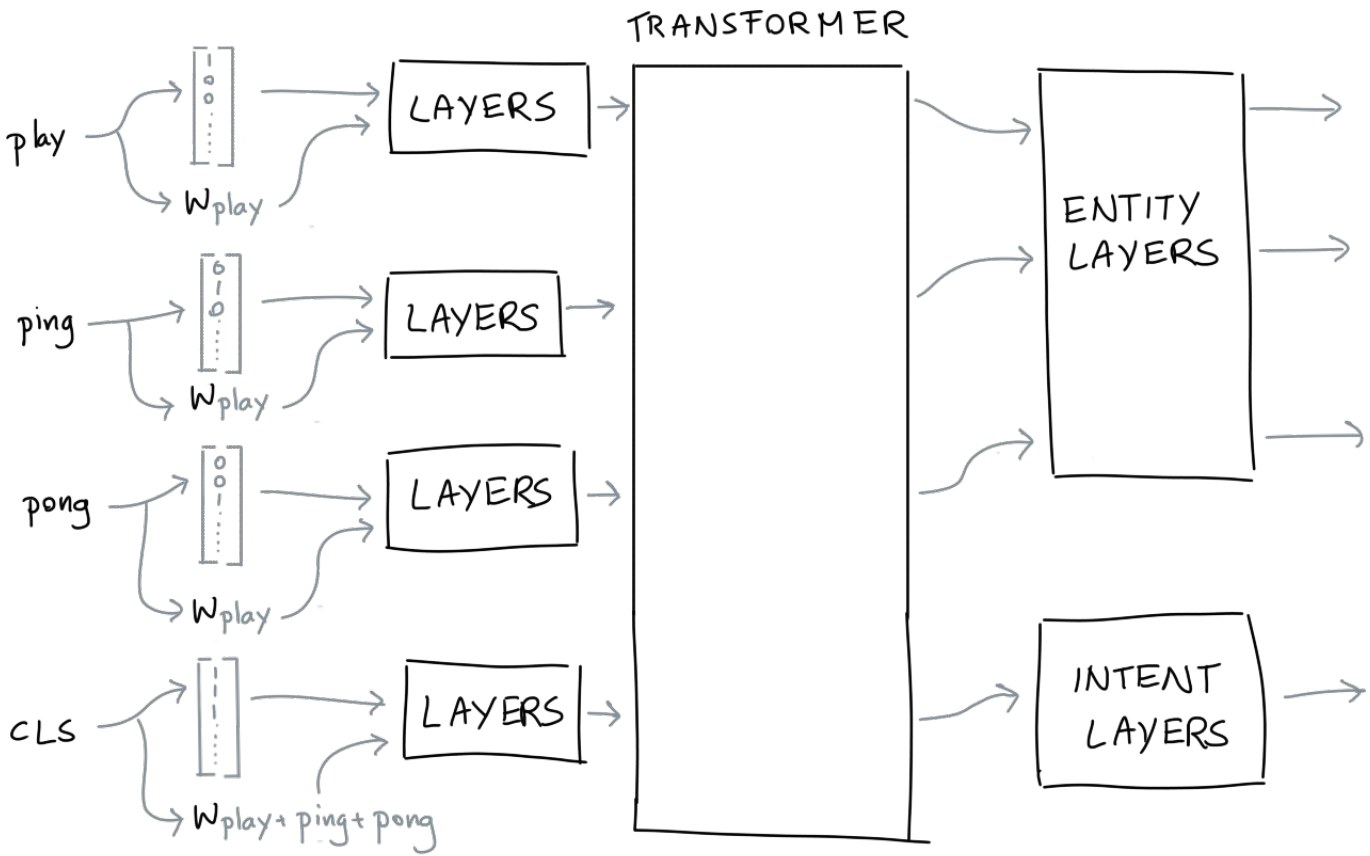
CLS –

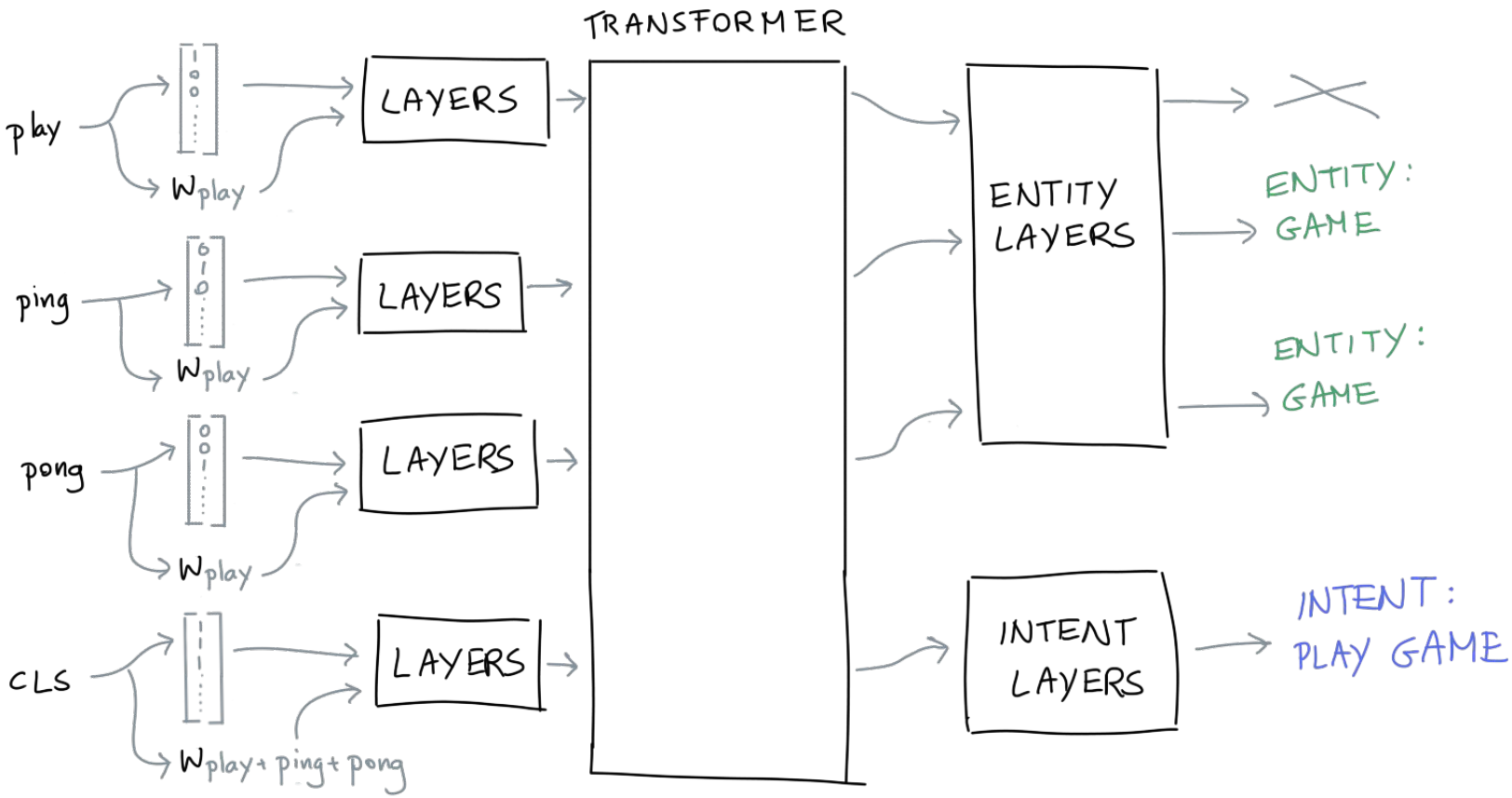

play
W_{play}
ping
W_{play}
pong
W_{play}
CLS
$W_{play} + ping + pong$

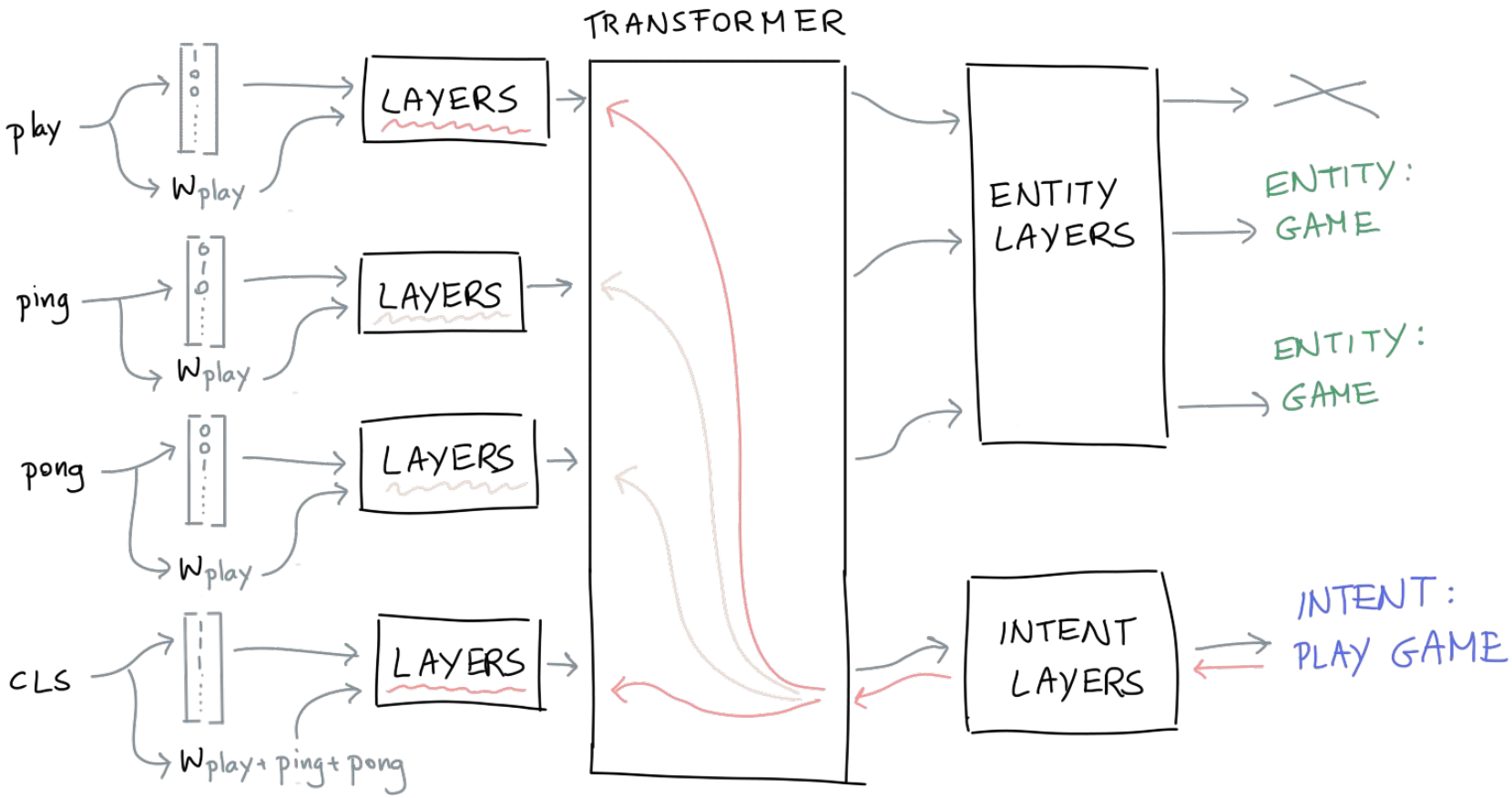

play
LAYERS
W_{play}
ping
LAYERS
W_{play}
pong
LAYERS
W_{play}
CLS
LAYERS
$W_{play} + ping + pong$


TRANSFORMER
play
ping
pong
CLS
LAYERS
LAYERS
LAYERS
LAYERS
Wplay
Wplay
Wplay
Wplay+ping+pong


TRANSFORMER
play
ping
pong
CLS
LAYERS
LAYERS
LAYERS
LAYERS
Wplay
Wplay
Wplay
Wplay+ ping+ pong
ENTITY
LAYERS


TRANSFORMER
play
ping
pong
CLS
W play
W play
W play
W play + ping + pong
LAYERS
LAYERS
LAYERS
LAYERS
ENTITY LAYERS
INTENT LAYERS

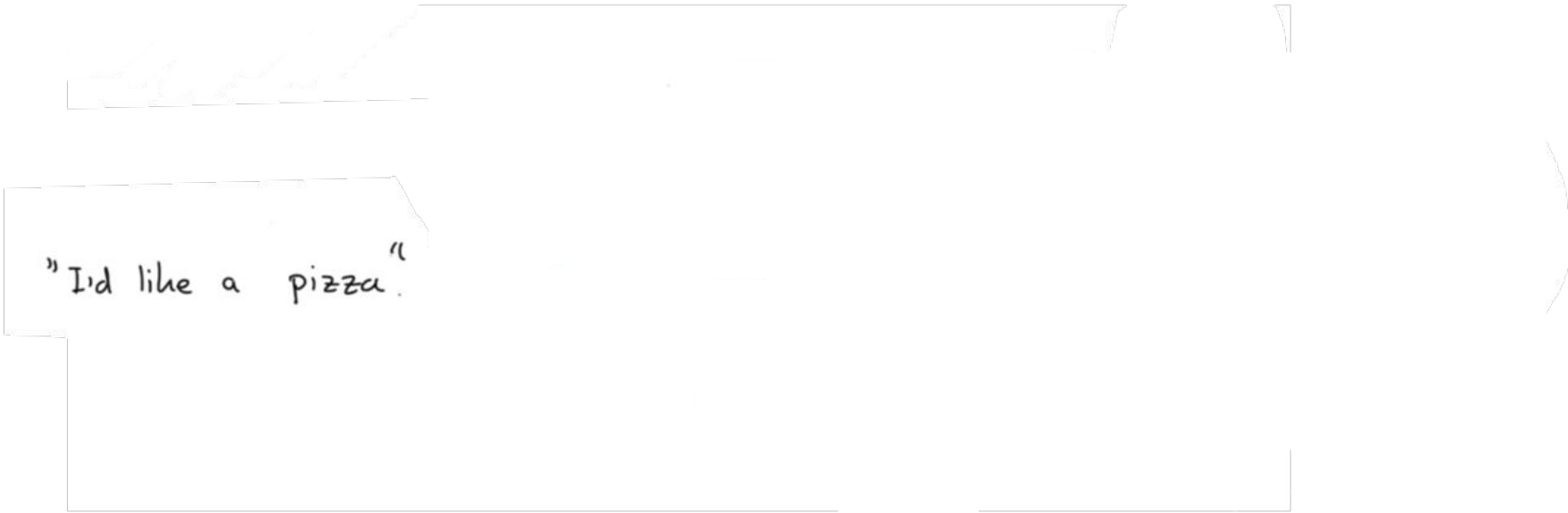

TRANSFORMER
play
ping
pong
CLS
LAYERS
LAYERS
LAYERS
LAYERS
Wplay
Wplay
Wplay
Wplay + ping + pong
ENTITY LAYERS
INTENT LAYERS
ENTITY: GAME
ENTITY: GAME
INTENT: PLAY GAME


TRANSFORMER
play
ping
pong
CLS
W play
W play
W play
W play + ping + pong
LAYERS
LAYERS
LAYERS
LAYERS
ENTITY LAYERS
INTENT LAYERS
ENTITY: GAME
ENTITY: GAME
INTENT: PLAY GAME

Q: So how does Rasa determine what actions to take?

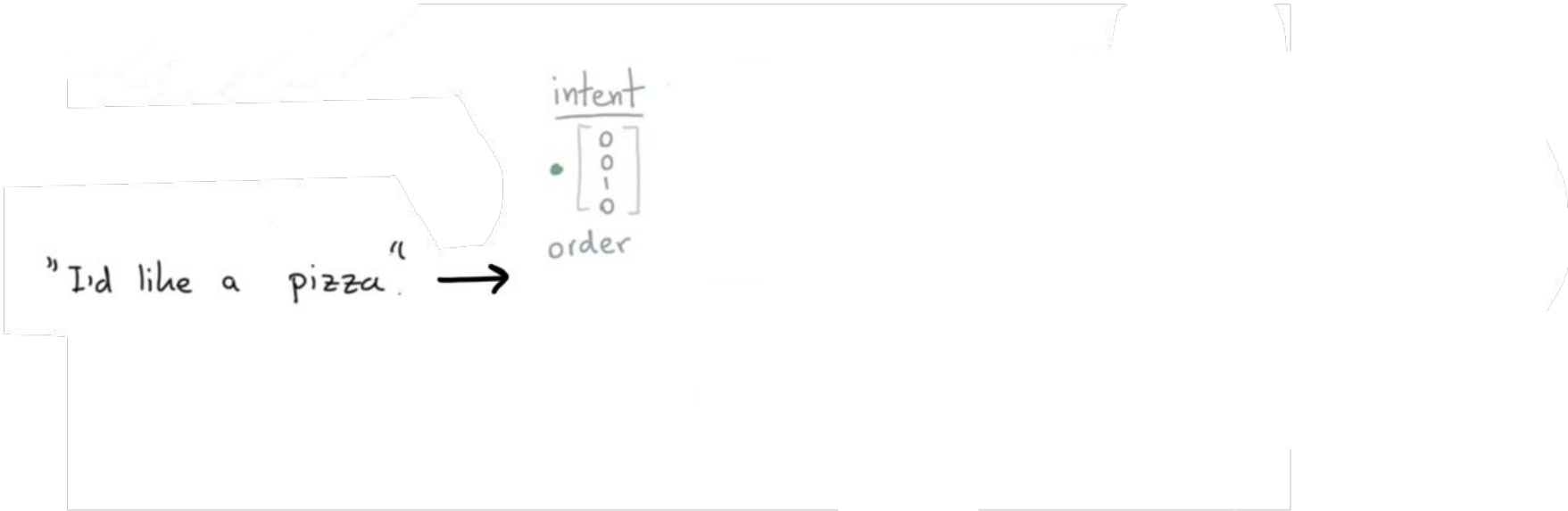
A: TED
Transformer Embedding Dialogue

TED policy



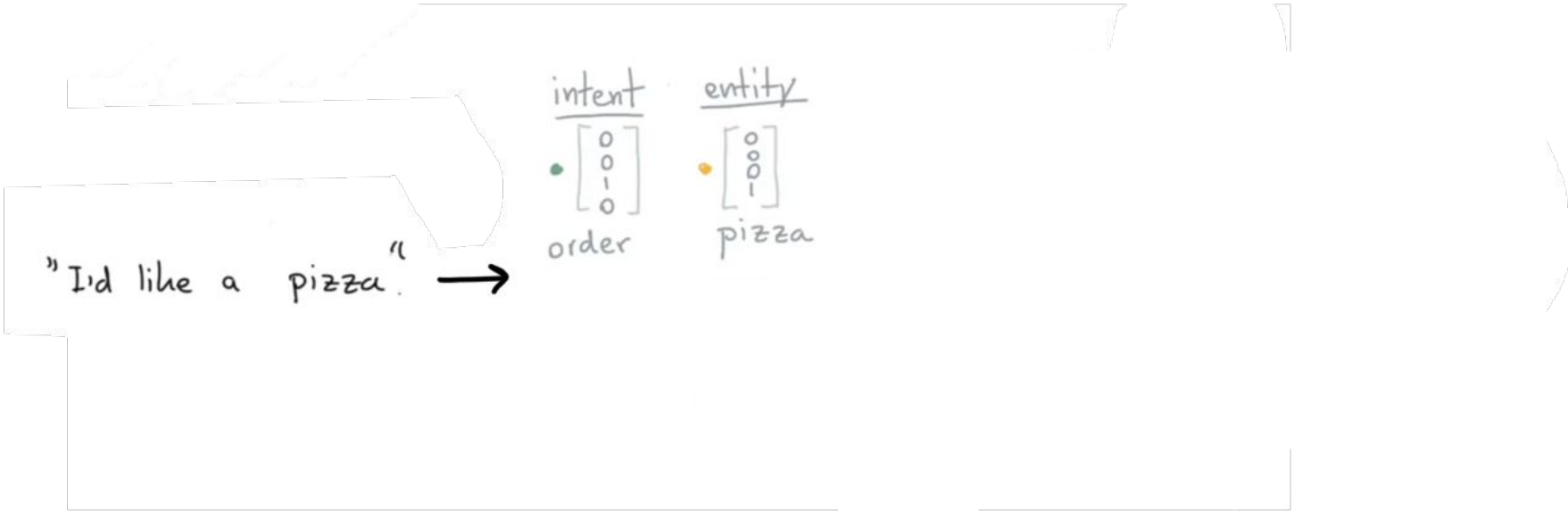
*More info blablabla

TED policy



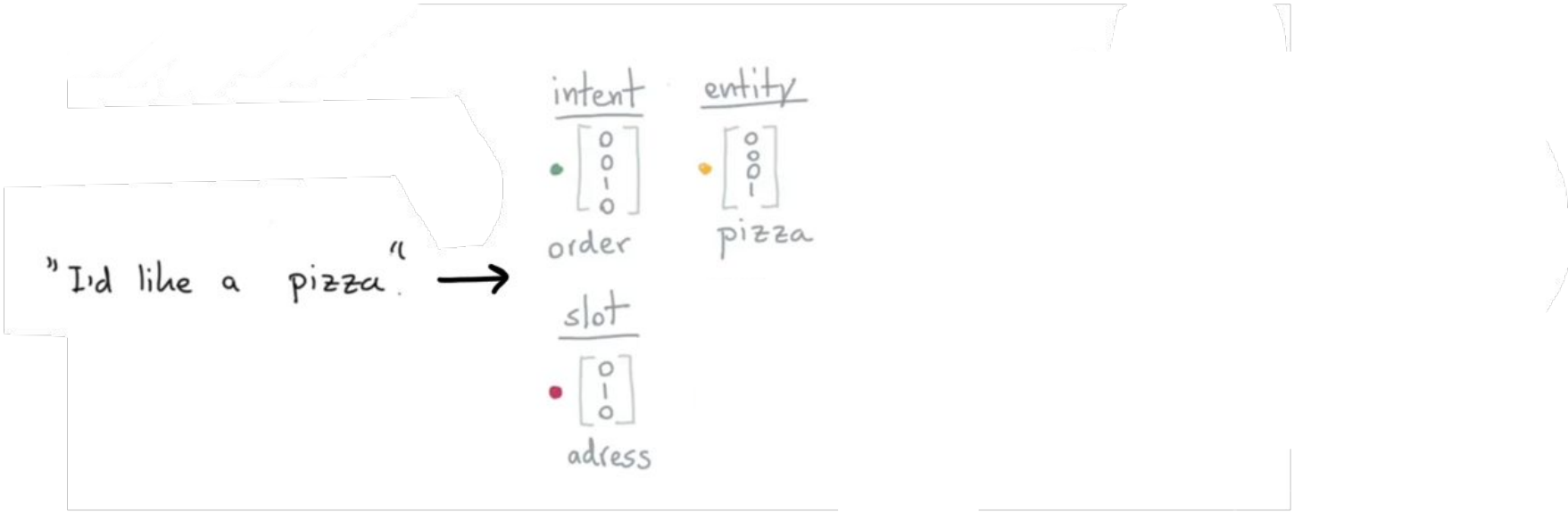
*More info blablabla

TED policy



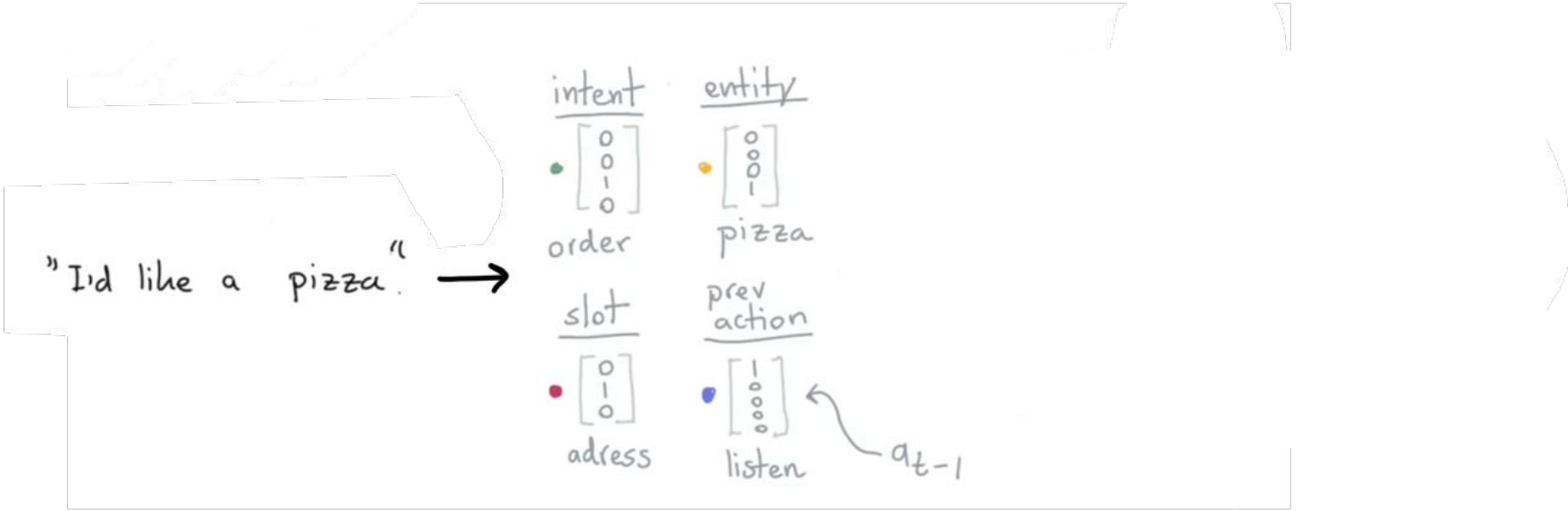
*More info blablabla

TED policy



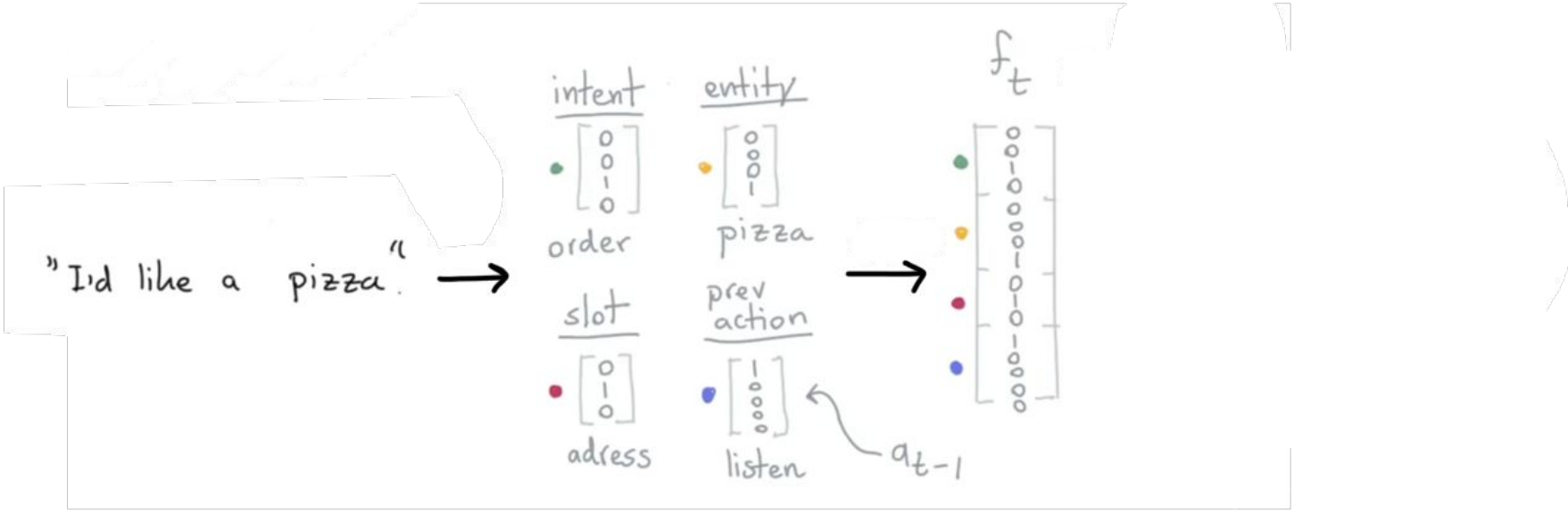
*More info blablabla

TED policy



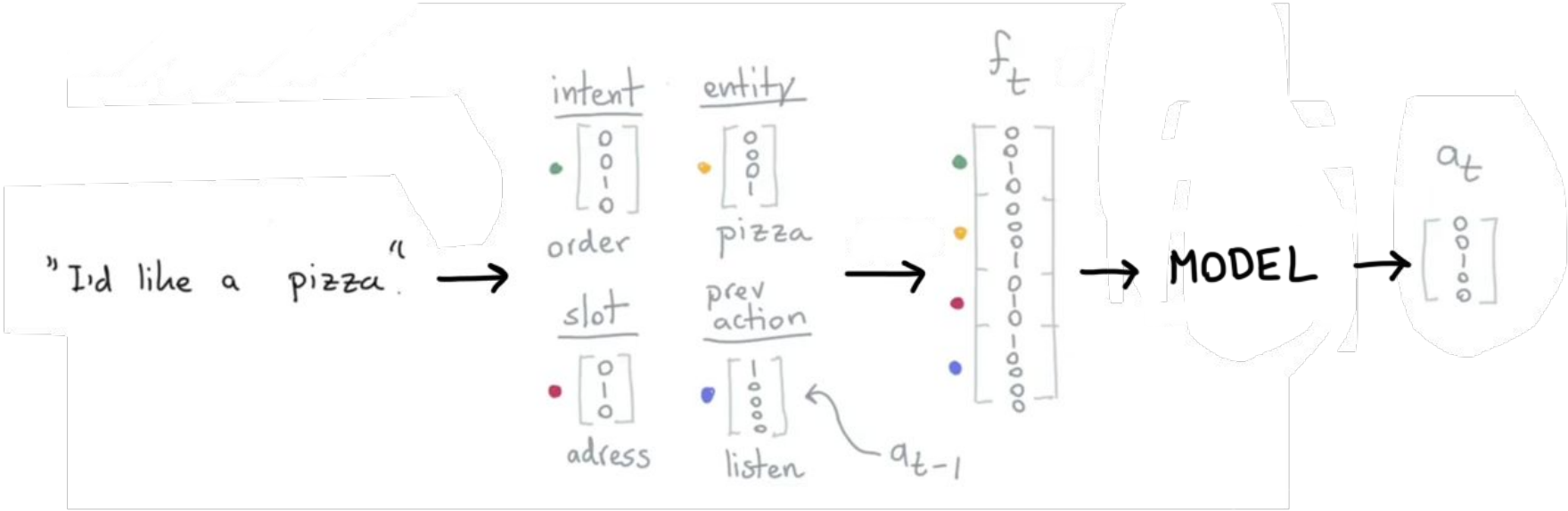
*More info blablabla

TED policy



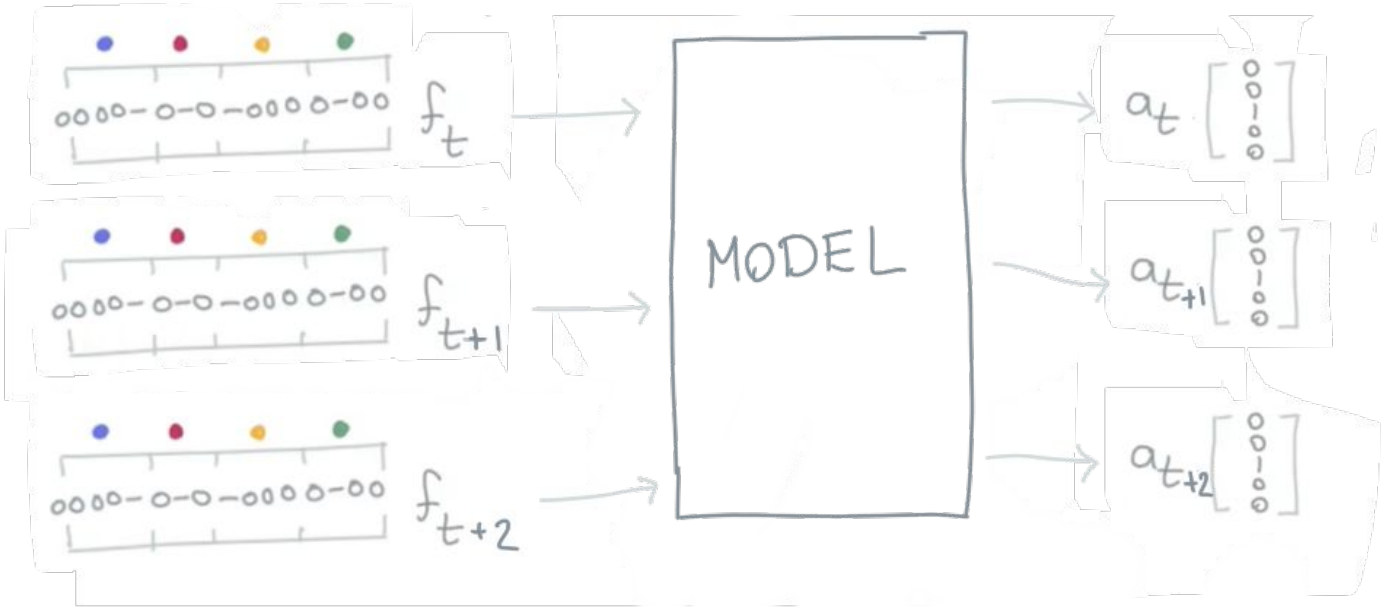
*More info blablabla

TED policy

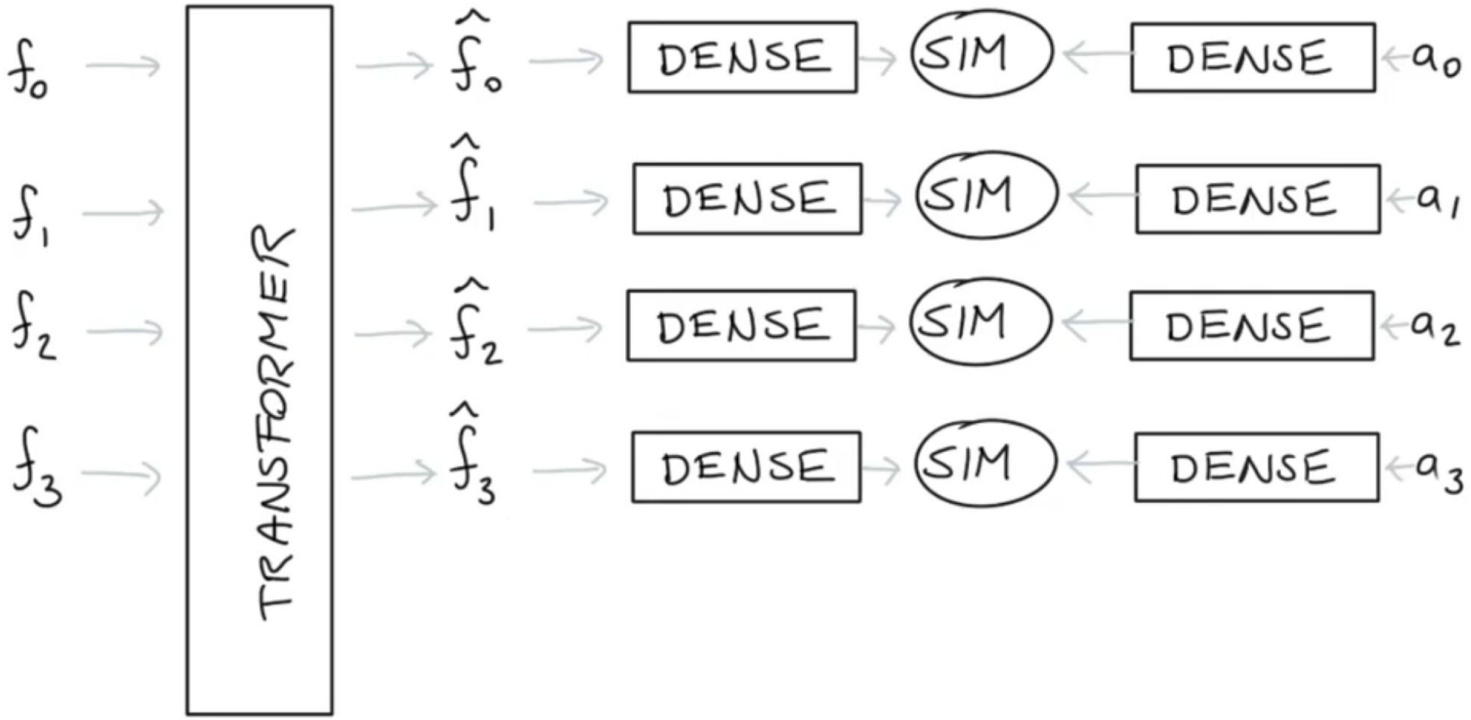


*More info blablabla

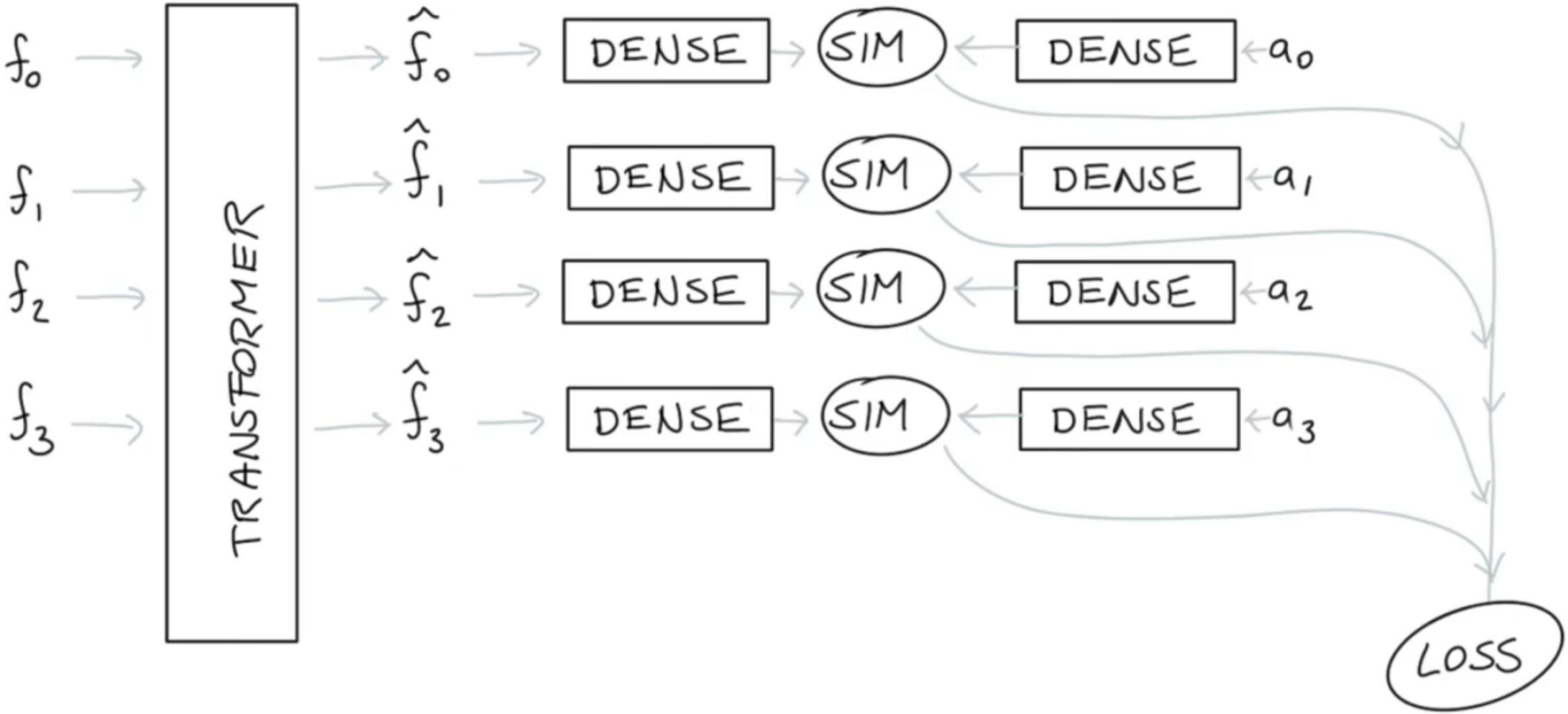
TED policy



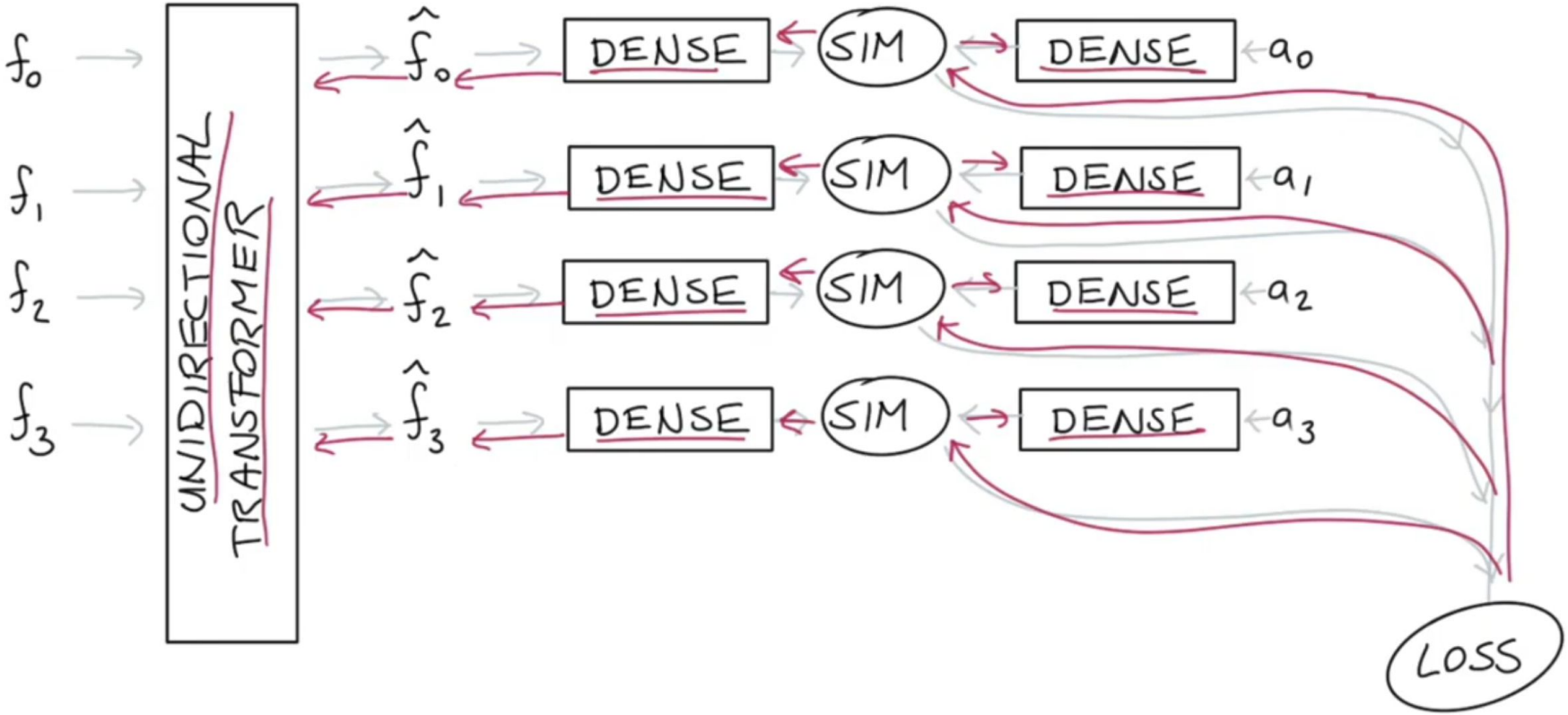
*More info blablabla



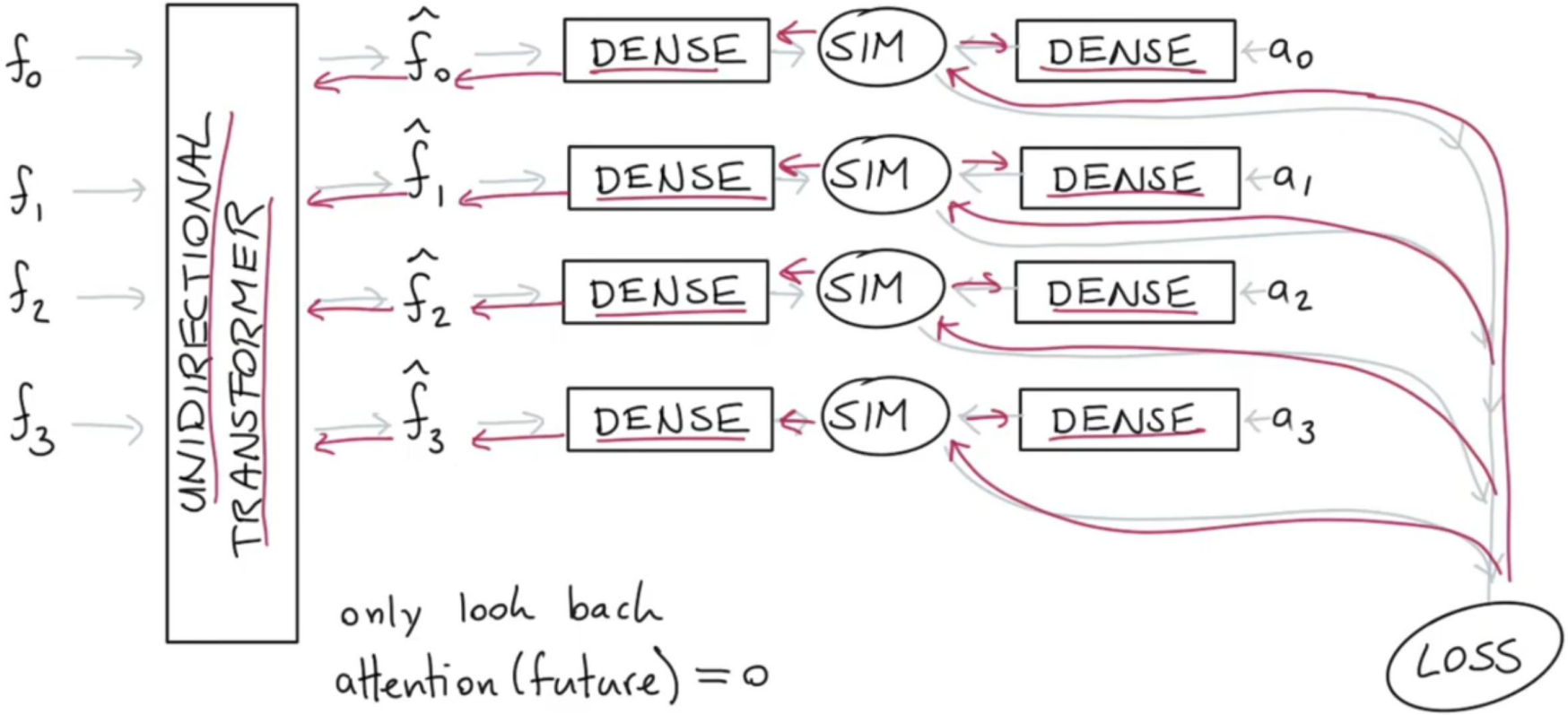
*More info blablabla



*More info blablabla



*More info blablabla



*More info blablabla

Demotime

*More info blablabla

config.yml

```
policies:
  - name: MemoizationPolicy
  - name: TEDPolicy
    epochs: 200
    max_history: 1
  - name: MappingPolicy
```

```
Your input ->  start counting
    Countdown! ETA 10 🎵
Your input ->  ok
    Countdown! ETA 5 🎵
Your input ->  ok
    Countdown! ETA 5 🎵
Your input ->  ok
    Countdown! ETA 5 🎵
Your input ->  ok
    Countdown! ETA 5 🎵
```

*More info blablabla

config.yml

```
policies:
  - name: MemoizationPolicy
  - name: TEDPolicy
    epochs: 200
    max_history: 3
  - name: MappingPolicy
```

```
Your input ->  count
    Countdown! ETA 10 🎵
Your input ->  ok
    Countdown! ETA 9 🎵
Your input ->  ok
    Countdown! ETA 8 🎵
Your input ->  ok
    Countdown! ETA 7 🎵
Your input ->  are you a bot?
    I am a bot, not a human, powered by Rasa.
    Countdown! ETA 6 🎵
Your input ->  ok
    Countdown! ETA 5 🎵
Your input ->  are you a bot?
    I am a bot, not a human, powered by Rasa.
Your input ->
    Countdown! ETA 10 🎵
```


*More info blablabla

config.yml

```
policies:
  - name: MemoizationPolicy
  - name: TEDPolicy
    epochs: 200
    max_history: 10
  - name: MappingPolicy
```

```
Your input -> count
    Countdown! ETA 10 🎵
Your input -> ok
    Countdown! ETA 9 🎵
Your input -> ok
    Countdown! ETA 8 🎵
Your input -> are you a bot?
    I am a bot, not a human, powered by Rasa.
    Countdown! ETA 7 🎵
Your input -> ok
    Countdown! ETA 6 🎵
Your input -> are you a bot?
    I am a bot, not a human, powered by Rasa.
    Countdown! ETA 5 🎵
Your input -> ok
    Countdown! ETA 4 🎵
Your input -> are you a bot?
    I am a bot, not a human, powered by Rasa.
    Countdown! ETA 3 🎵
Your input -> ok
    Countdown! ETA 2 🎵
Your input -> are you a bot?
    I am a bot, not a human, powered by Rasa.
    Countdown! ETA 1 🎵
Your input -> ok
    End of     countdown!
```


*More info blablabla

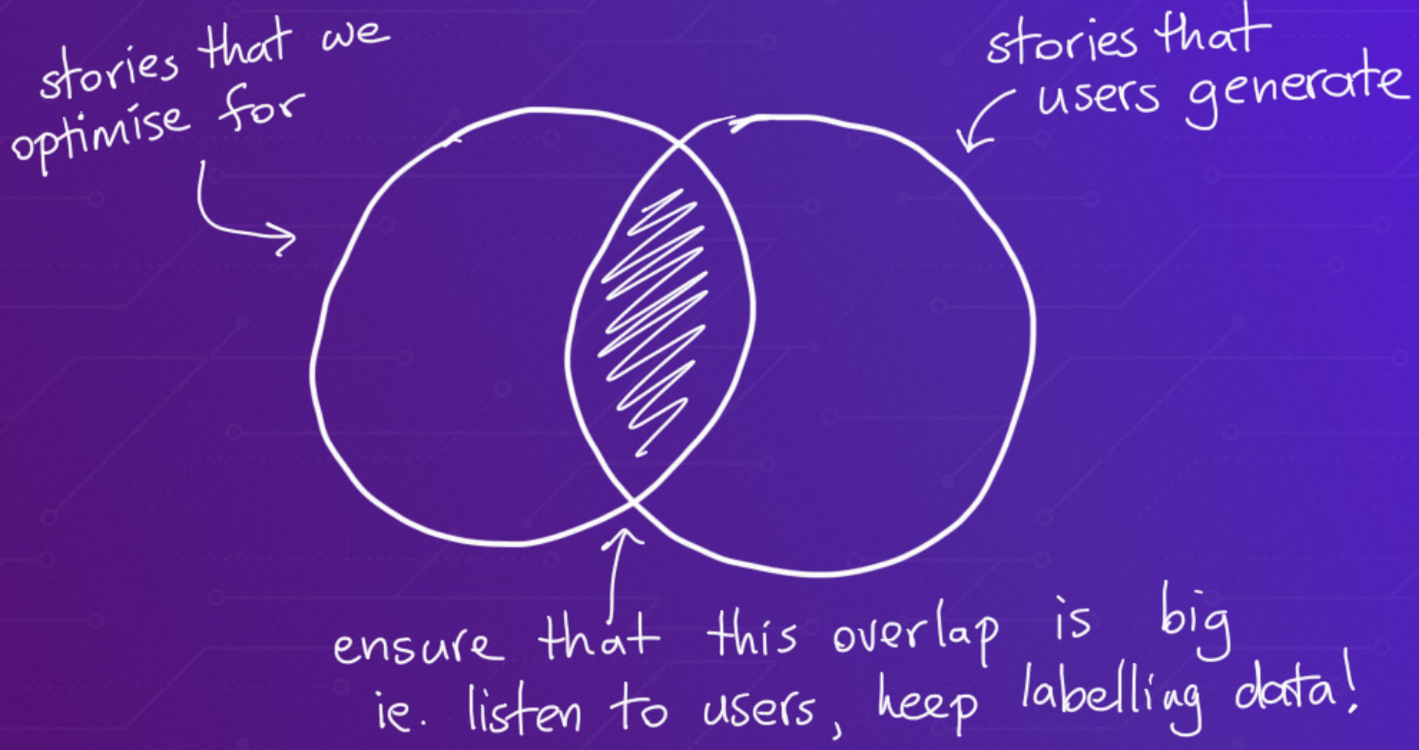
config.yml
Using LSTM

```
policies:
  - name: MemoizationPolicy
  - name: KerasPolicy
    epochs: 200
  - name: MappingPolicy
```

```
Your input ->  count
Countdown! ETA 10 🎵
Your input ->  ok
Countdown! ETA 9 🎵
Your input ->  ok
Countdown! ETA 8 🎵
Your input ->  are you a bot?
I am a bot, not a human, powered by Rasa.
Countdown! ETA 7 🎵
Your input ->  are you a bot
I am a bot, not a human, powered by Rasa.
Countdown! ETA 6 🎵
Your input ->  are you a bot?
Countdown! ETA 5 🎵
Your input ->  ok
Countdown! ETA 4 🎵
Your input ->  are you a bot?
I am a bot, not a human, powered by Rasa.
Countdown! ETA 3 🎵
Your input ->  are you a bot?
I am a bot, not a human, powered by Rasa.
```


*More info blablabla

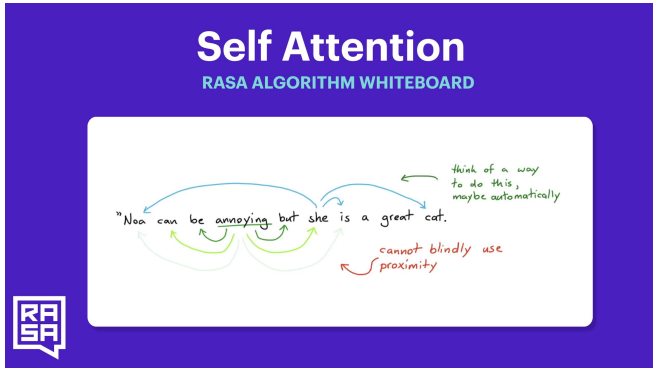
The real problem though.

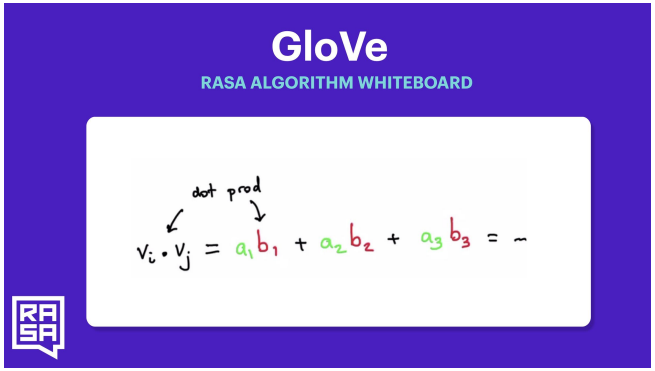


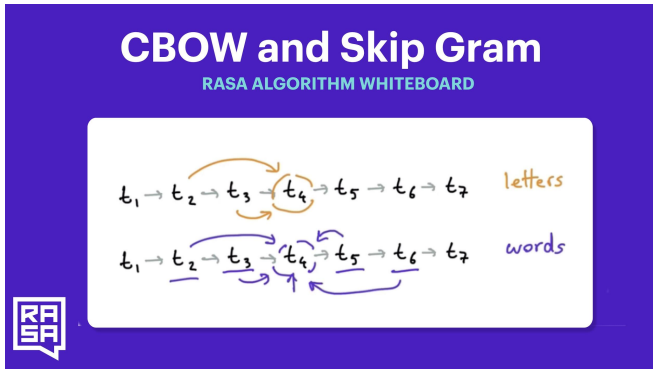
RASA

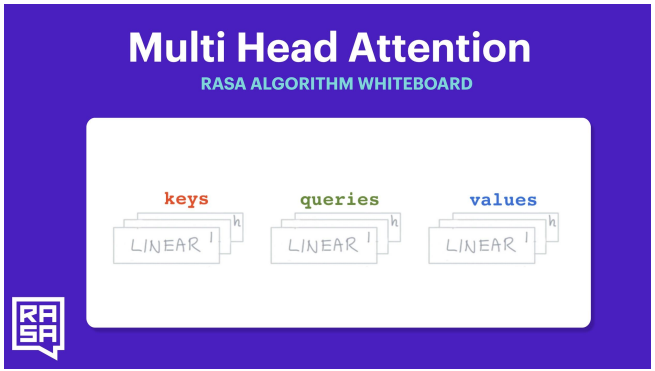
While I have you here: check the Rasa Algorithm Whiteboard!

Our algorithms are explained in more detail! I also take requests for content.



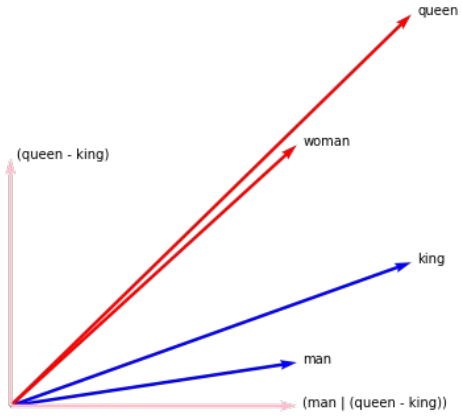


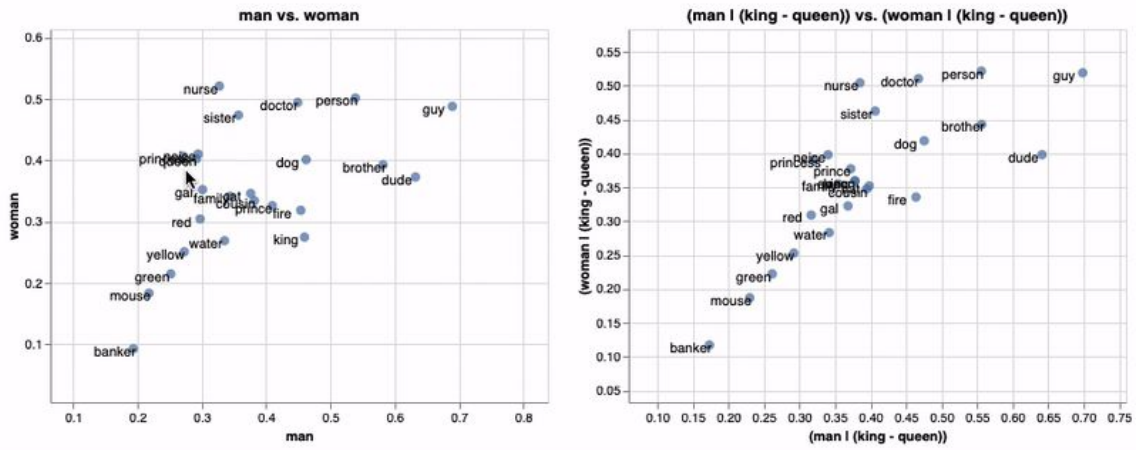




While I have you here: check out WHATLIES

```
orig_chart = emb.plot_interactive('man', 'woman')
new_ts     = emb | (emb['king'] - emb['queen'])
new_chart  = new_ts.plot_interactive('man', 'woman')
```





It's an open source package for visualising word embeddings.

Soon: features for detecting bias. Feedback is appreciated!

RASA

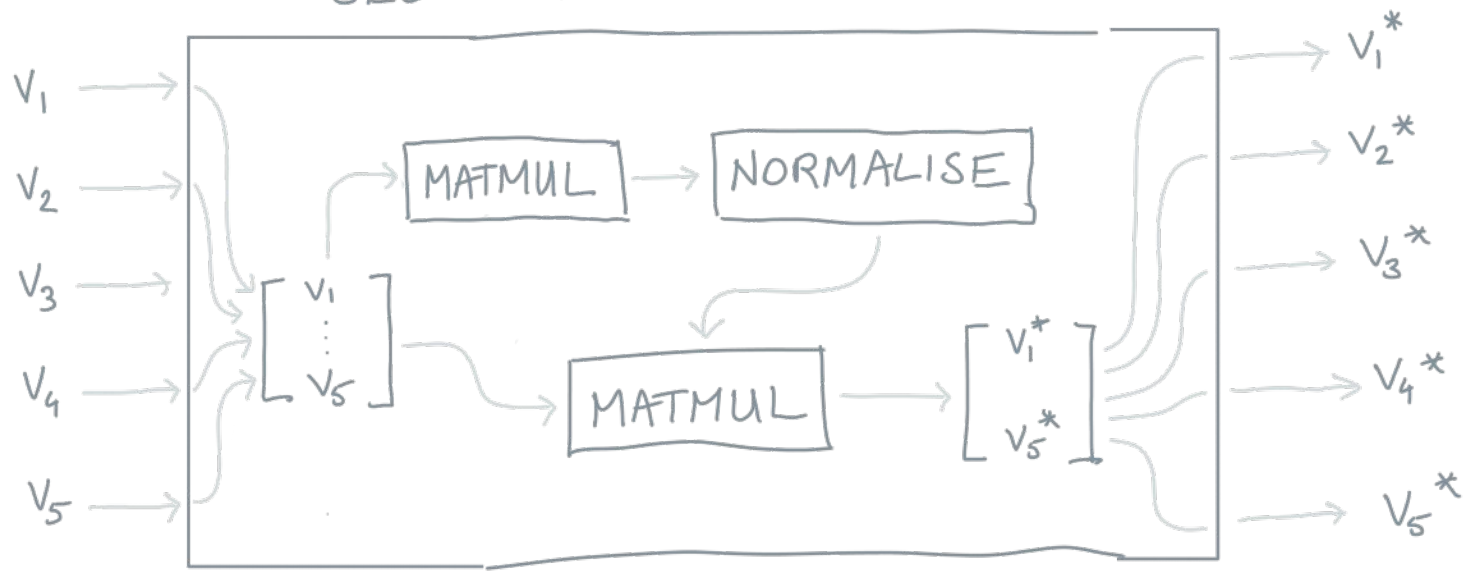
Get in touch!

Vincent D. Warmerdam
Research Advocate
v.warmerdam@rasa.com

Appendix

Vincent D. Warmerdam
Research Advocate
v.warmerdam@rasa.com

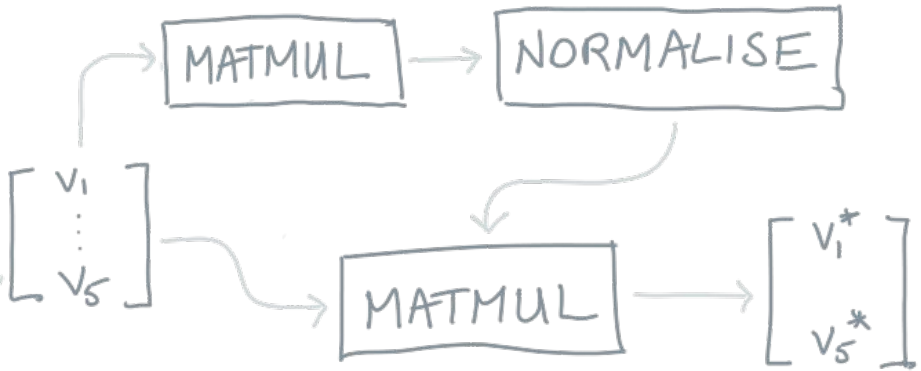
SELF ATTENTION BLOCK



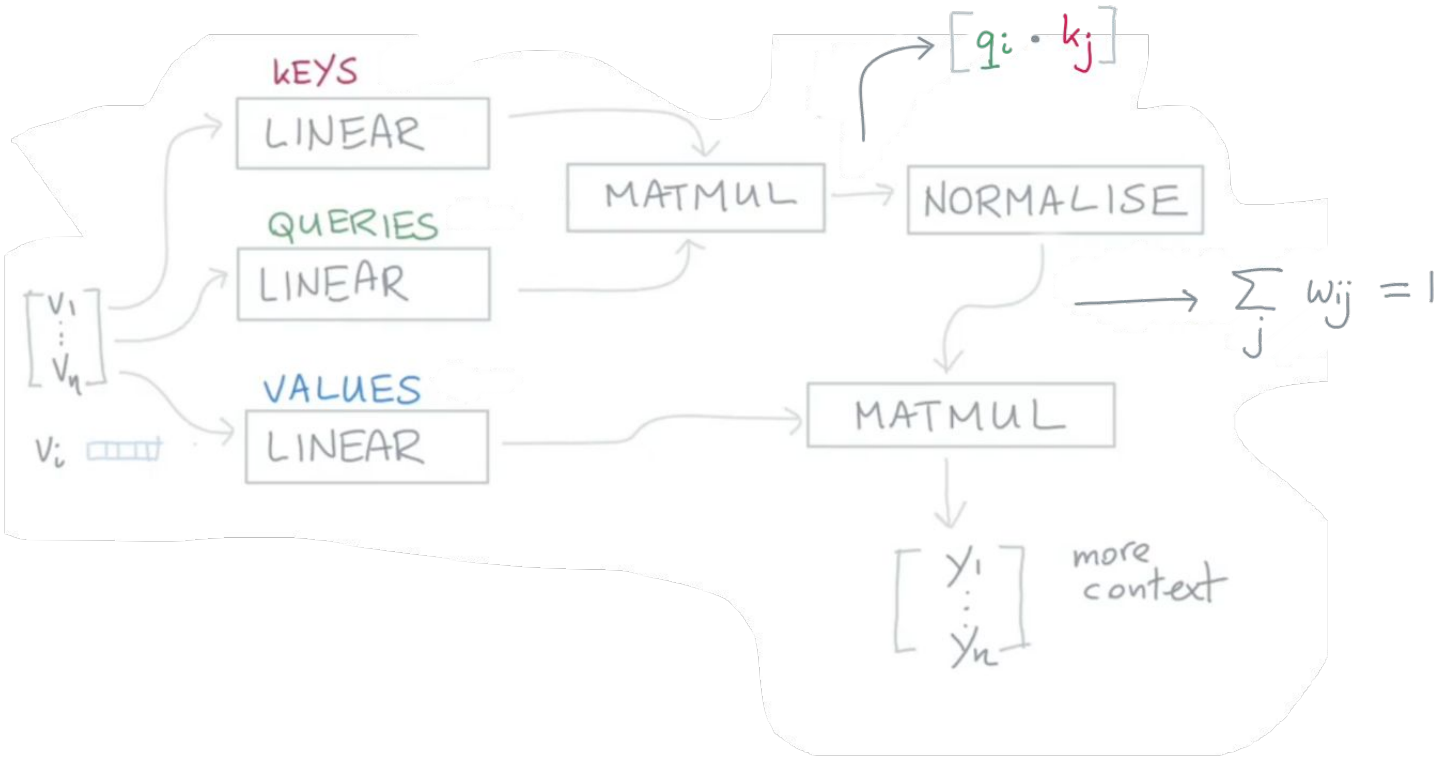
- ORDER DOES NOT MATTER
- NO TRAINABLE WEIGHTS (YET)
- DEPENDS ON PRETRAINED QUALITY
- TASK INDEPENDANT

*More info blablabla

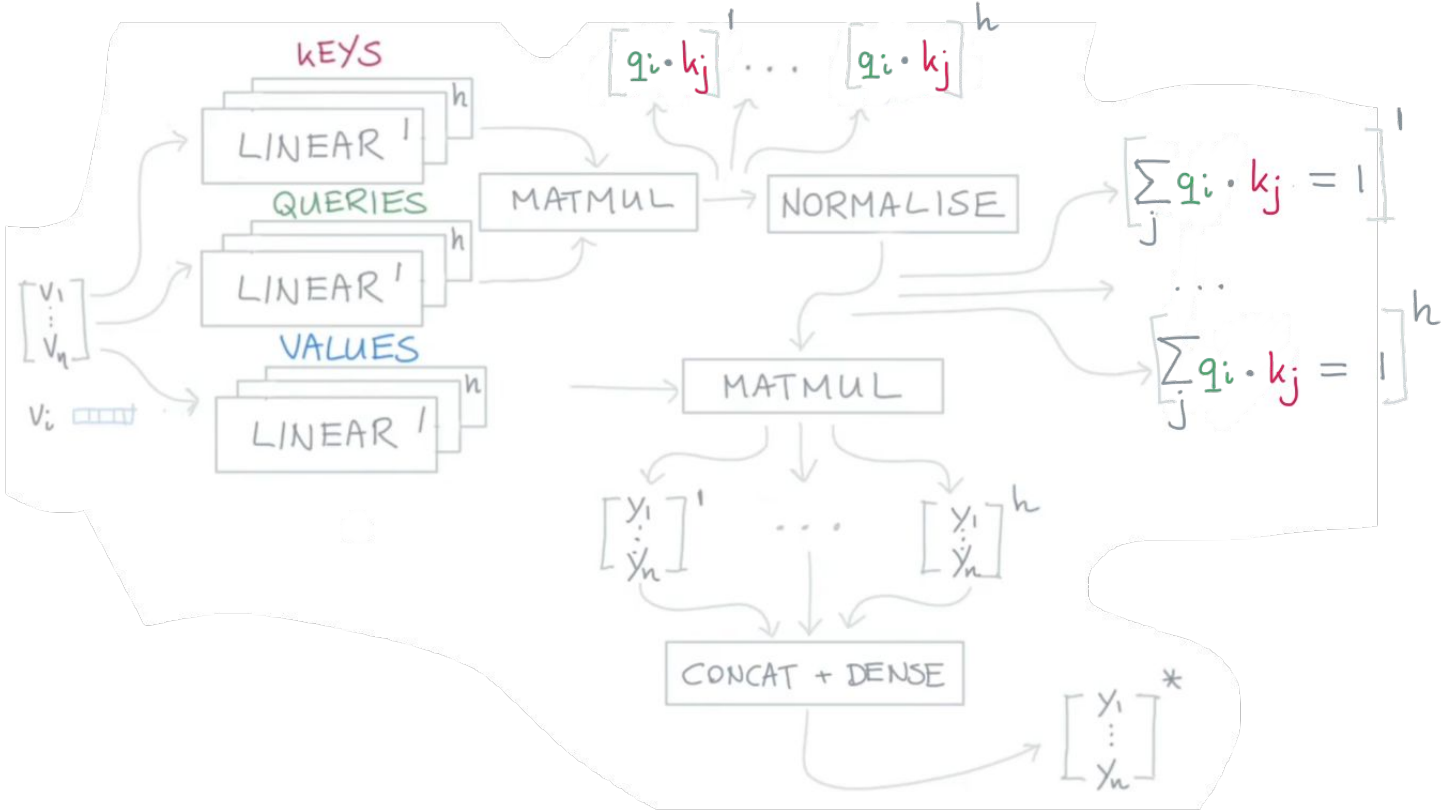
How's about we add us some trainable weights ?



*More info blablabla

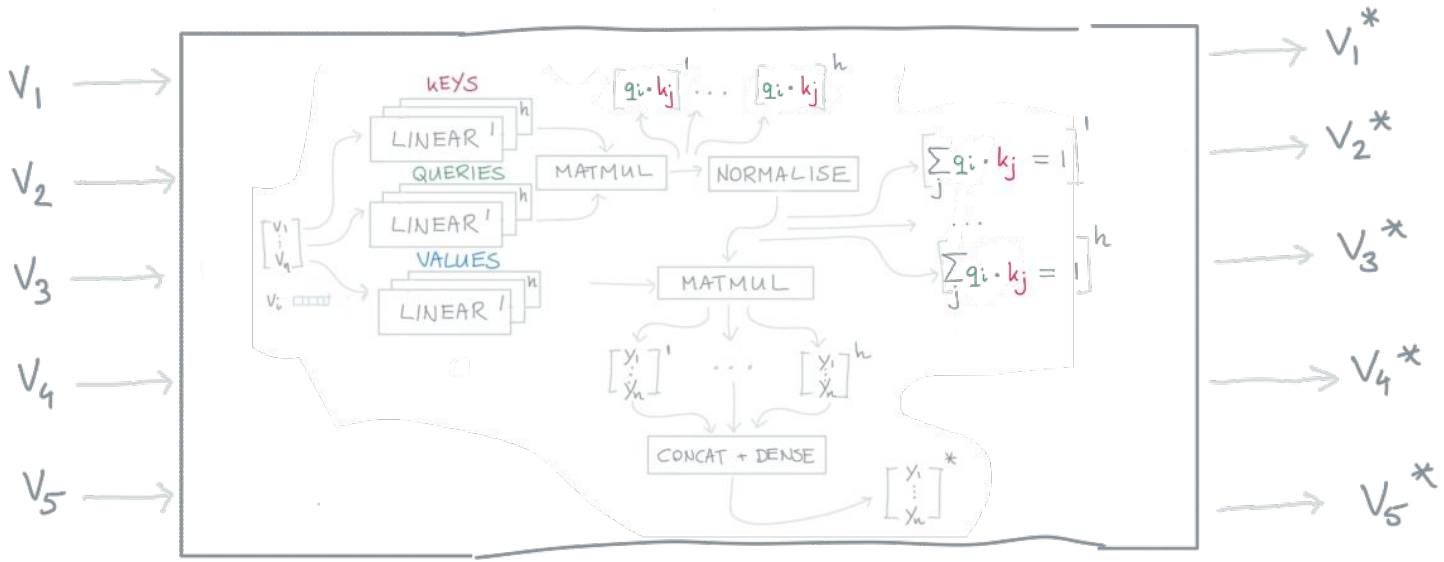


*More info blablabla

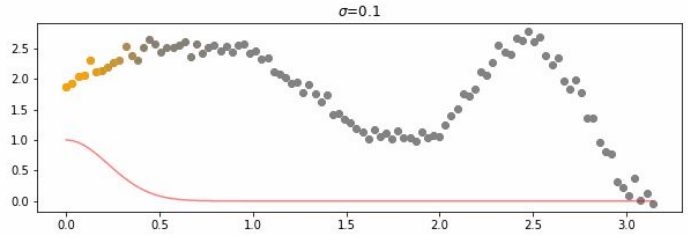


*More info blablabla

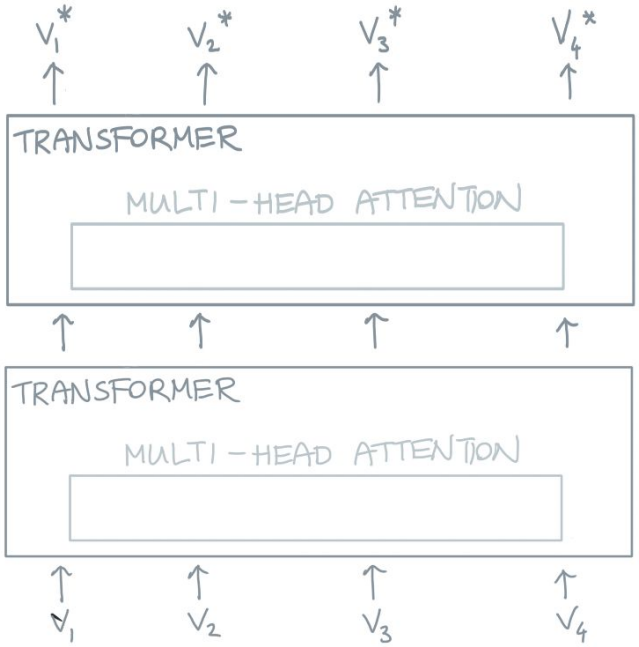
MULTI HEAD ATTENTION



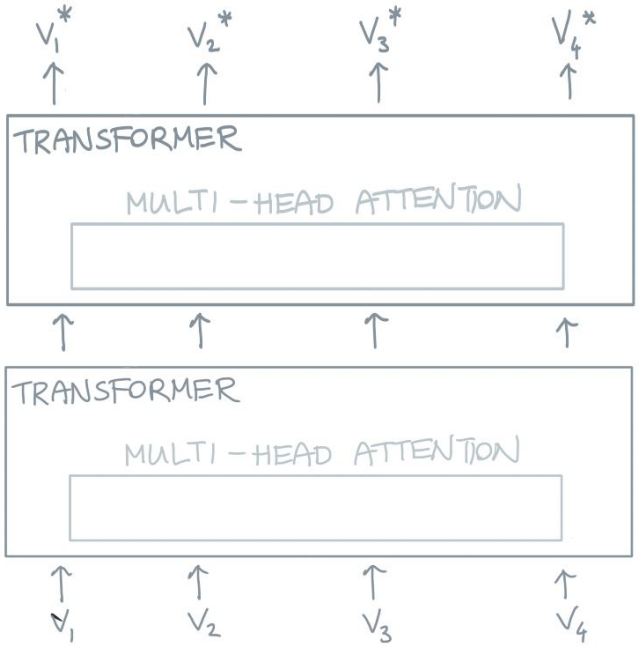
It's a more elaborate way to do stuff like this →



*More info blablabla



*More info blablabla



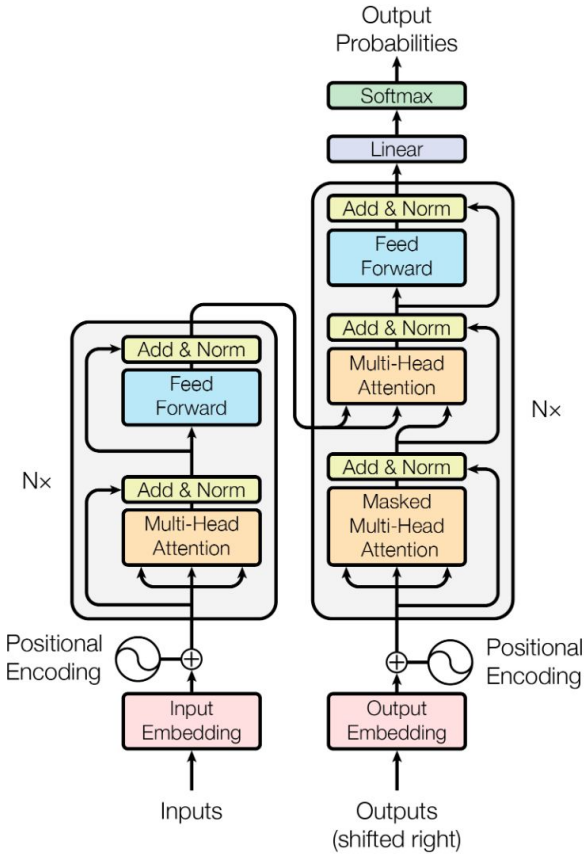
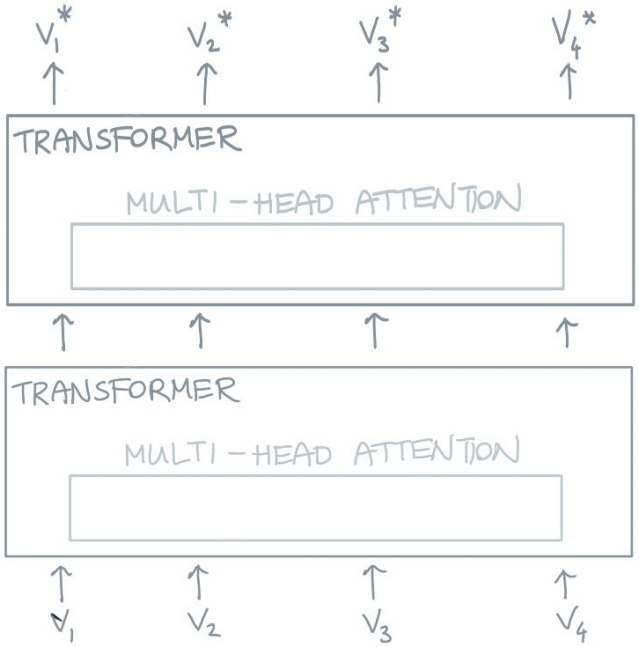


Figure 1: The Transformer - model architecture.

*More info blablabla



We use this `encoder` bit.

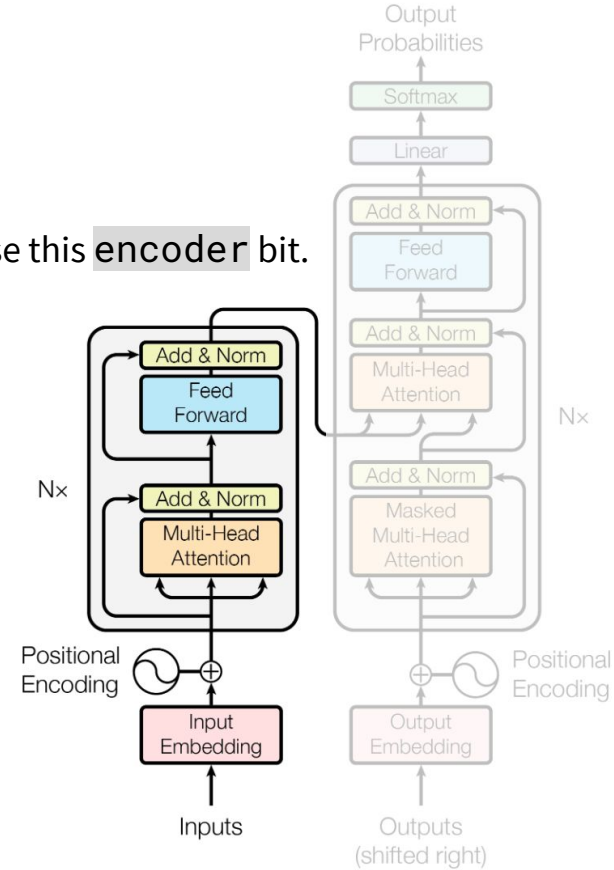


Figure 1: The Transformer - model architecture.

*More info blablabla