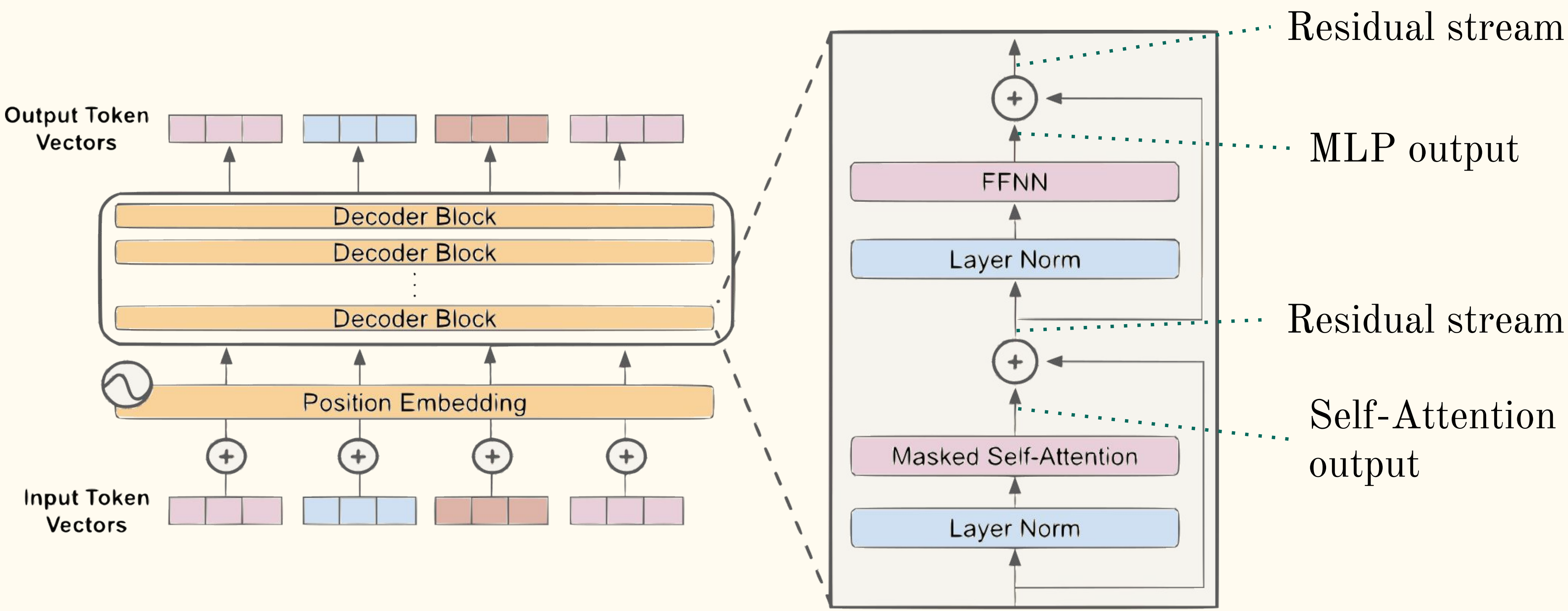
Modelling Trajectories of Language Models

Nicky Pochinkov, Einar Urdshals, Jasmina Nasufi,
Éloïse Benito-Rodriguez, Mikołaj Kniejski

We look at Transformer MLP Neurons



Trying to set up trajectory prediction

Title: Spaghetti alle Vongole - A Taste of the Italian Coast

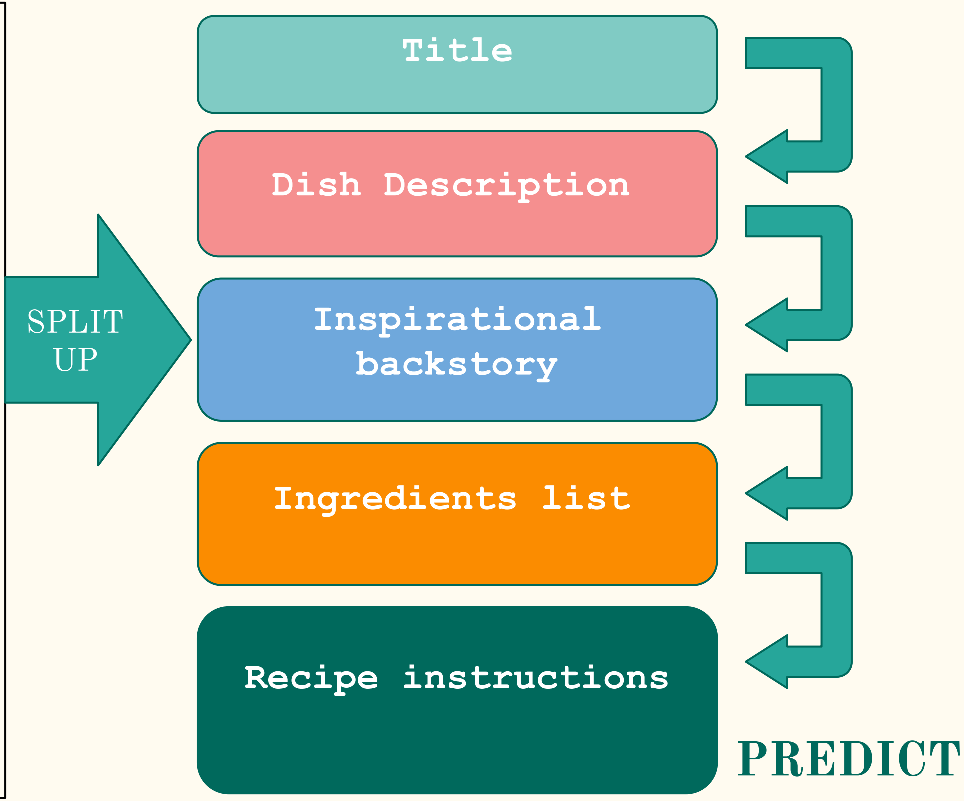
Spaghetti alle Vongole, or spaghetti clams, is a classic Italian…

My journey to Italy began on a crisp October morning, as I…

Recipe:
Ingredients: - 1 lb (454g) fresh spaghetti - 1/4 cup (55g) extra-virgin olive oil…

Directions: 1. Bring a large pot of salted water to a rolling boil. Cook the spaghetti according to the package instructions until al dente…

Generated Texts



SPLIT UP

Title

Dish Description

Inspirational backstory

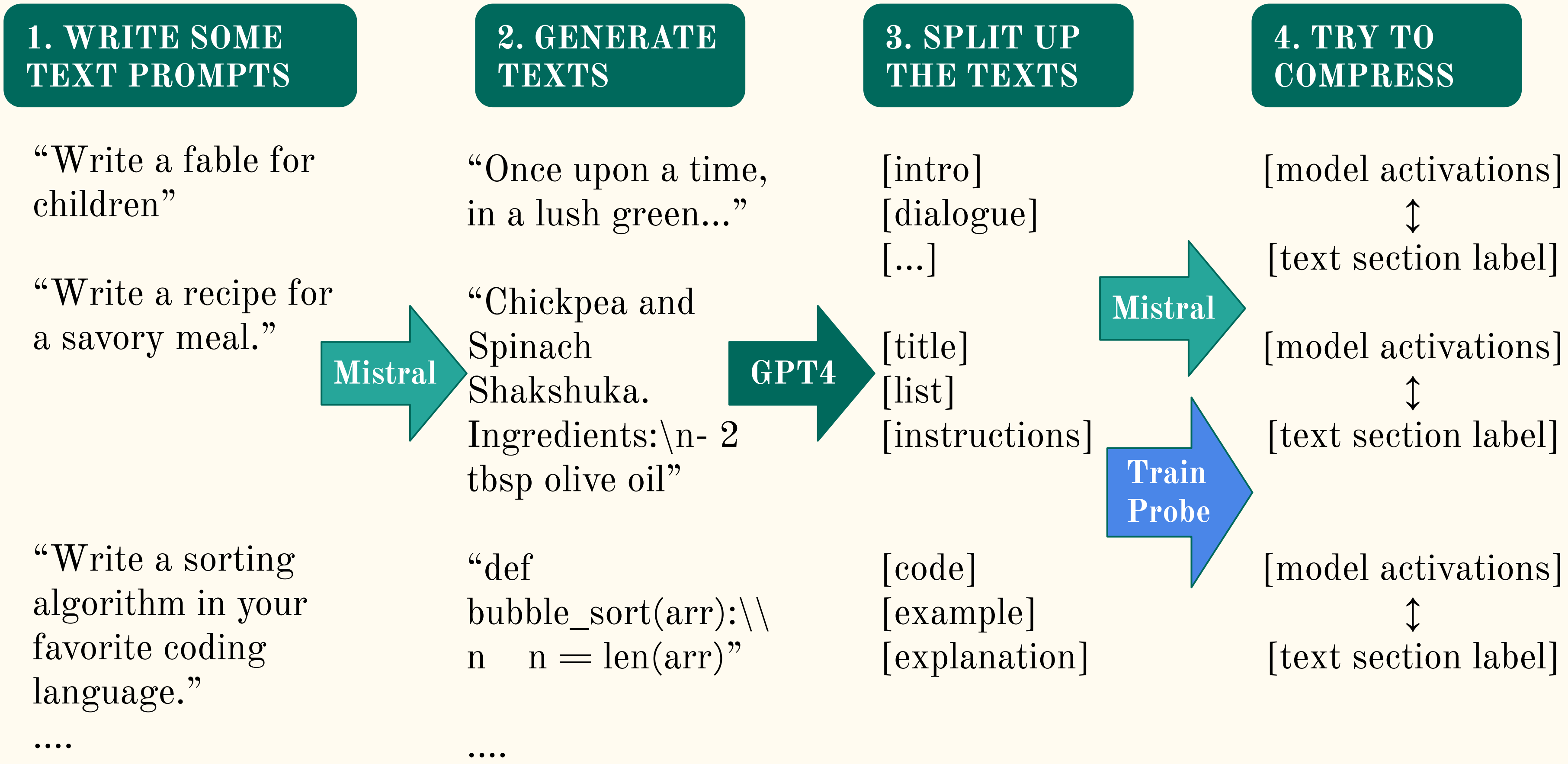
Ingredients list

Recipe instructions

PREDICT

Text Chunks

Our First Ideas - Generating Data



1. WRITE SOME TEXT PROMPTS

“Write a fable for children”

“Write a recipe for a savory meal.”

“Write a sorting algorithm in your favorite coding language.”

....

Mistral

2. GENERATE TEXTS

“Once upon a time, in a lush green...”

“Chickpea and Spinach Shakshuka. Ingredients:\n- 2 tbsp olive oil”

“def bubble_sort(arr):\\ n n = len(arr)”

....

GPT4

3. SPLIT UP THE TEXTS

[intro]
[dialogue]
[...]

[title]
[list]
[instructions]

[code]
[example]
[explanation]

Mistral

Train Probe

4. TRY TO COMPRESS

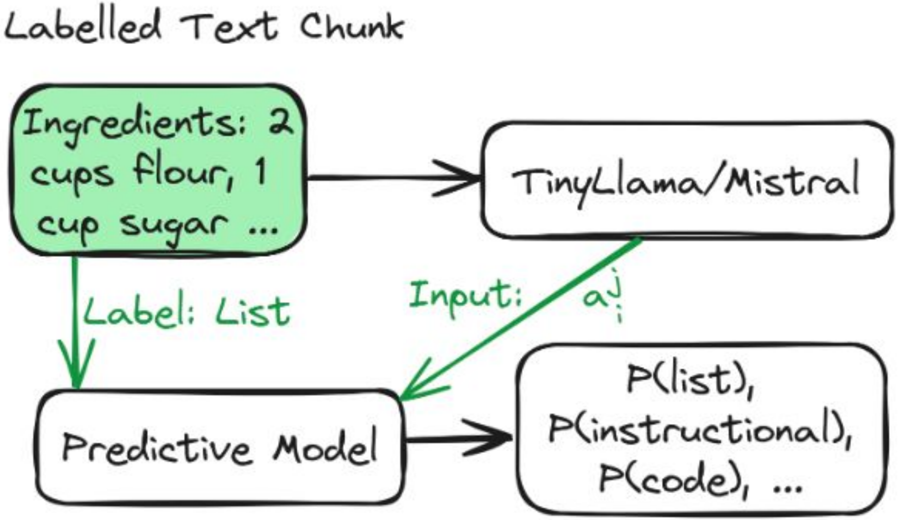
[model activations] ↕ [text section label]

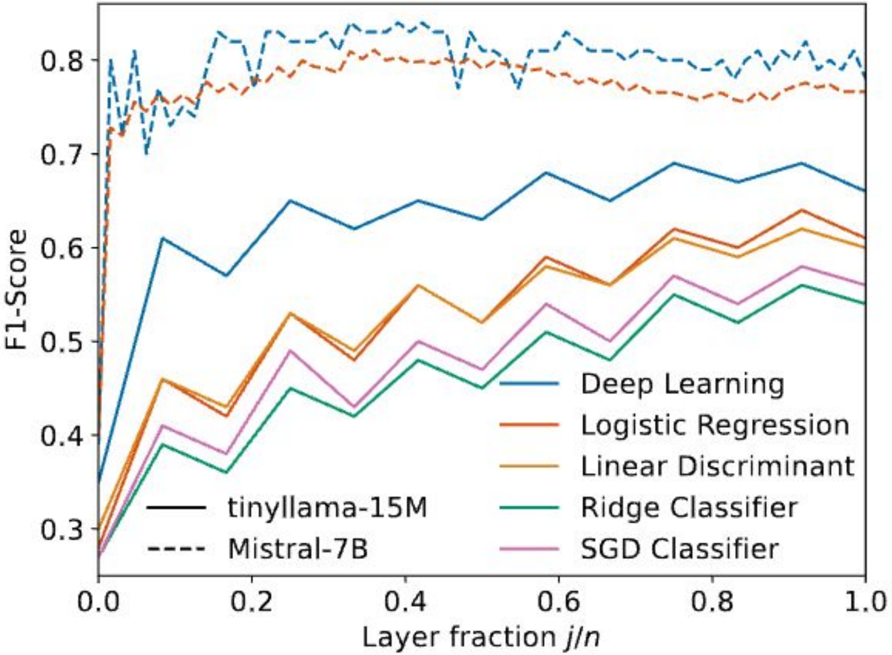
[model activations] ↕ [text section label]

[model activations] ↕ [text section label]

We hoped it would be easy to make a simple dataset

We thought we had some successes probing...


Labelled Text Chunk
Ingredients: 2 cups flour, 1 cup sugar ...
TinyLlama/Mistral
Label: List
Input: a_i^j
Predictive Model
P(list), P(instructional), P(code), ...



However the picture is a bit more nuanced

Correction of probing experiment

We should have a control model to see if probes can have high performances even on model with random weights.

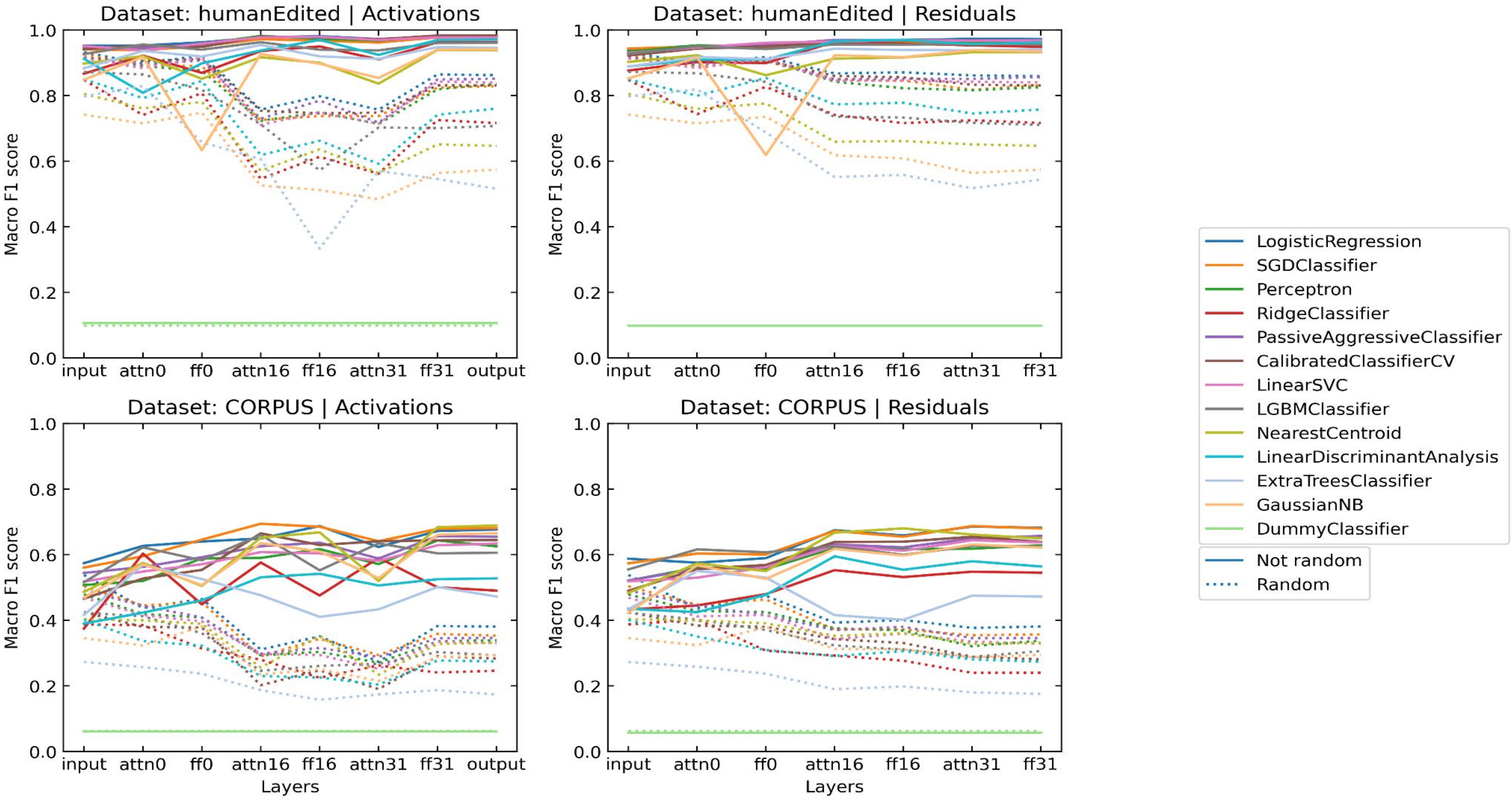
We also:

- recreate our dataset “humanEdited” by increasing diversity by generating more prompts, we have 5 labels of text category: Narrative, List, Speech, Code, Explanation

- took an already existing dataset: Corpus CORE.

- took the mean of activations and residuals stream in a chunk, instead of individual tokens

Our results today with Mistral-7B



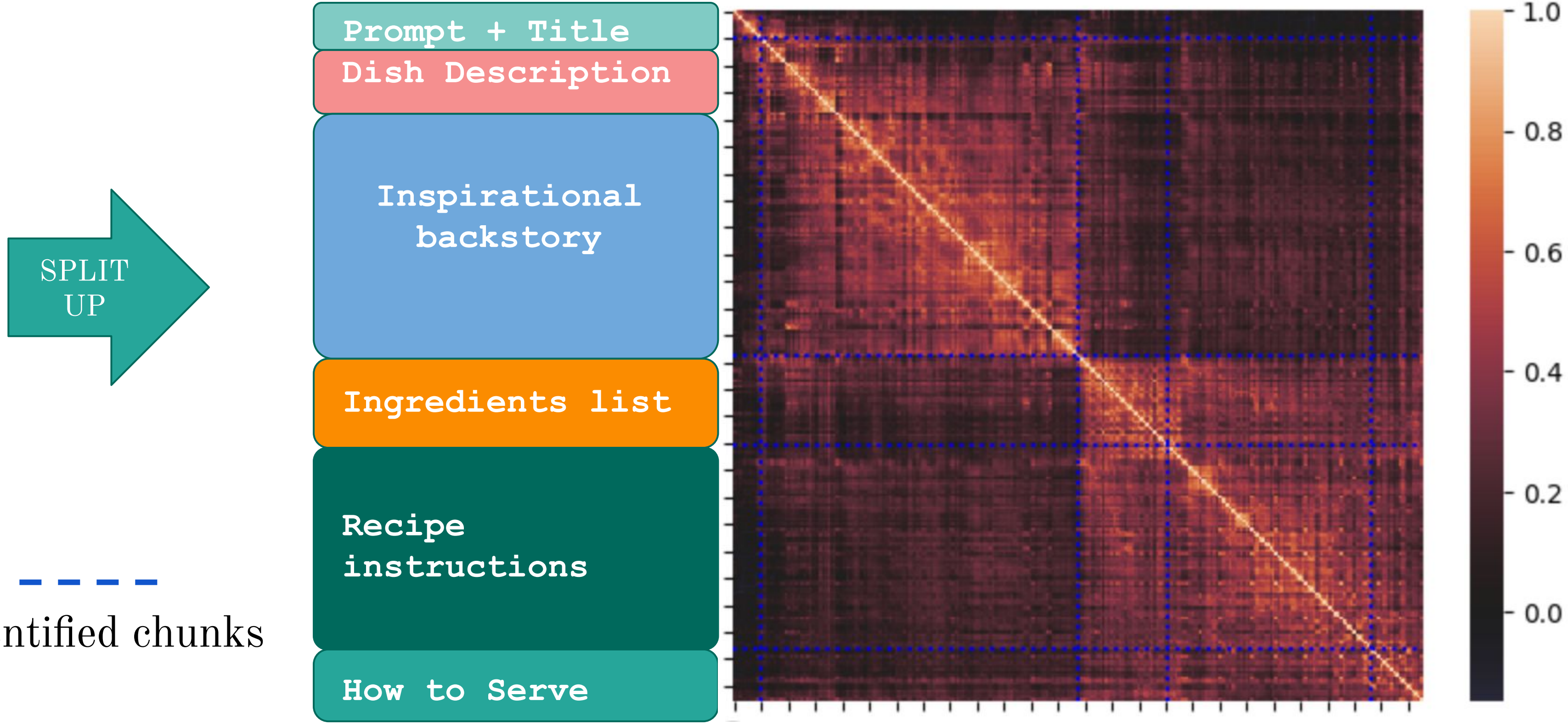
We have found splitting into “text chunks” seems easy

Simple “Chunking” Algorithm

```
curr_chunk = [];
For token in tokens:
    If cosine_sim(token, curr_chunk.mean()) < threshold:
        # Start new chunk
    Else:
        curr_chunk.append(token)
```

Text Chunking Success

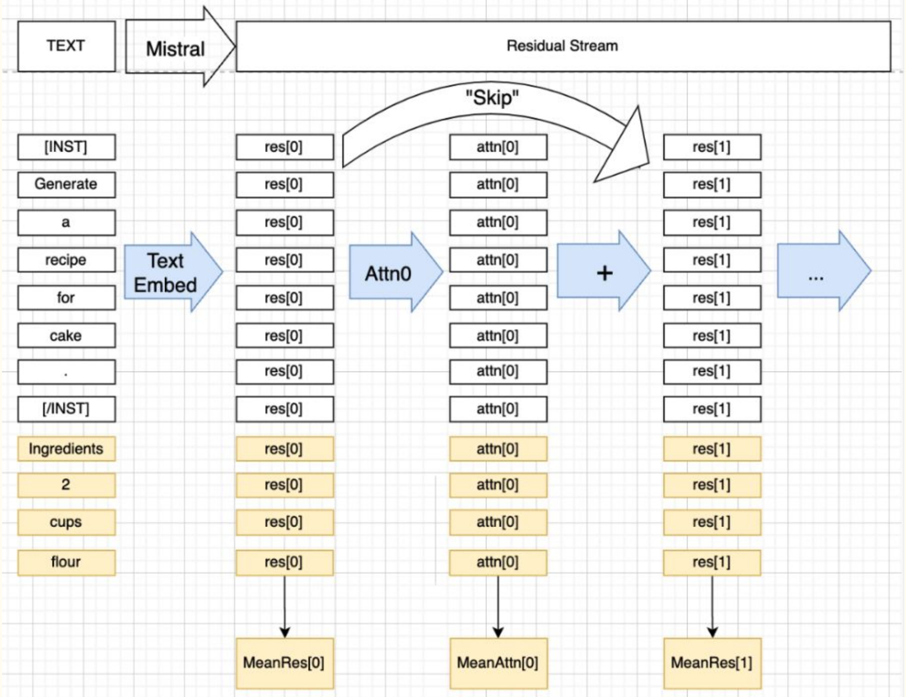
Cosine similarity between tokens in layer 15 of Mistral



Identified chunks

Trajectory Predicting has been somewhat difficult

LoRA Fine-tuned “Trajectory Predictor”



TEXT

Mistral

Residual Stream

"Skip"

Text Embed

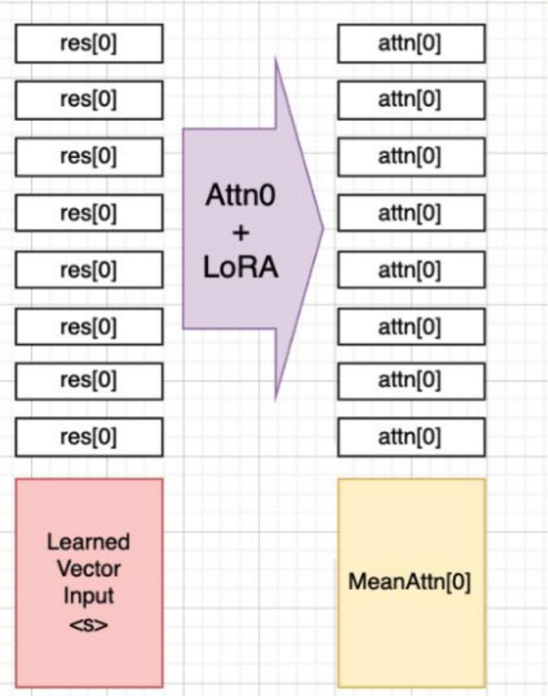
Attn0

+

...

[INST]	res[0]	attn[0]	res[1]
Generate	res[0]	attn[0]	res[1]
a	res[0]	attn[0]	res[1]
recipe	res[0]	attn[0]	res[1]
for	res[0]	attn[0]	res[1]
cake	res[0]	attn[0]	res[1]
.	res[0]	attn[0]	res[1]
[/INST]	res[0]	attn[0]	res[1]
Ingredients	res[0]	attn[0]	res[1]
2	res[0]	attn[0]	res[1]
cups	res[0]	attn[0]	res[1]
flour	res[0]	attn[0]	res[1]
	MeanRes[0]	MeanAttn[0]	MeanRes[1]

Train Attention Predictor



res[0]	Attn0 + LoRA	attn[0]
res[0]		attn[0]
res[0]		attn[0]
res[0]		attn[0]
res[0]		attn[0]
res[0]		attn[0]
res[0]		attn[0]
res[0]		attn[0]
Learned Vector Input <s>		MeanAttn[0]

Collect Activations

Train a Predictor

Mediocre Results

We try to train a predictor, with suboptimal results.

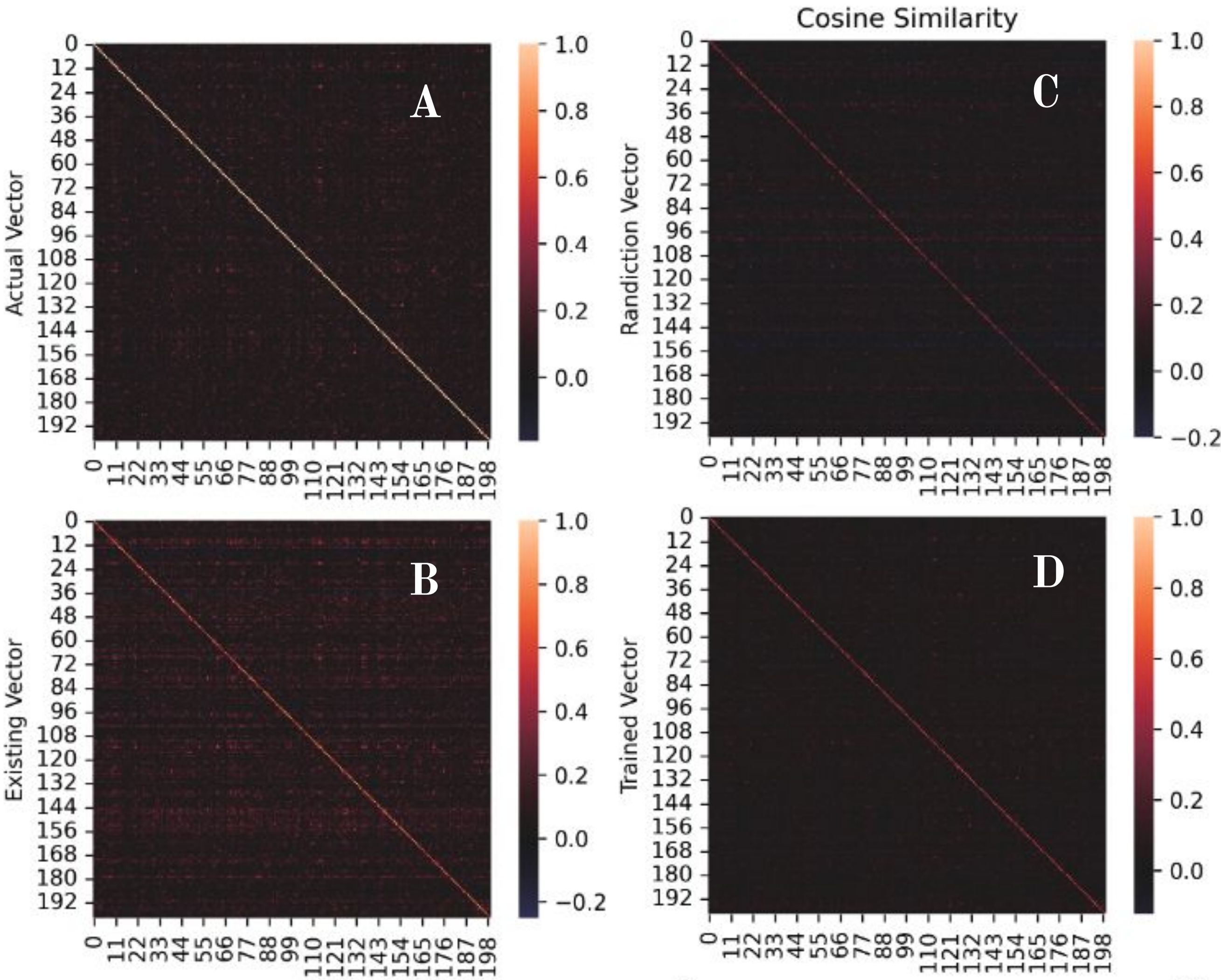
Cosine Similarity from **different text** outputs Expected[100:110] vs:

A) Expected (self)

B) Mean [0:100]

C) Baseline Attn

D) Fine-Tuned Attn



We hope to improve our trajectory predictor .

Questions?