Memorizing Transformers

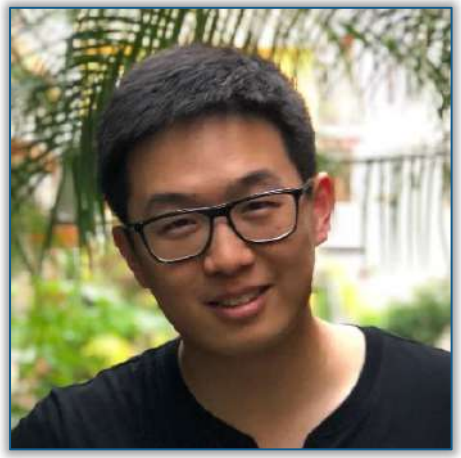
ICLR 2022 spotlight

Yuhuai Wu, Markus N. Rabe, DeLesley Hutchins, Christian Szegedy

Google

Reporter: Xiachong Feng

Authors



Yuhuai Wu
Postdoctoral Scholar at Stanford
Research Scientist at Google

Markus N. Rabe

DeLesley Hutchins

Christian Szegedy
Citation： 156032

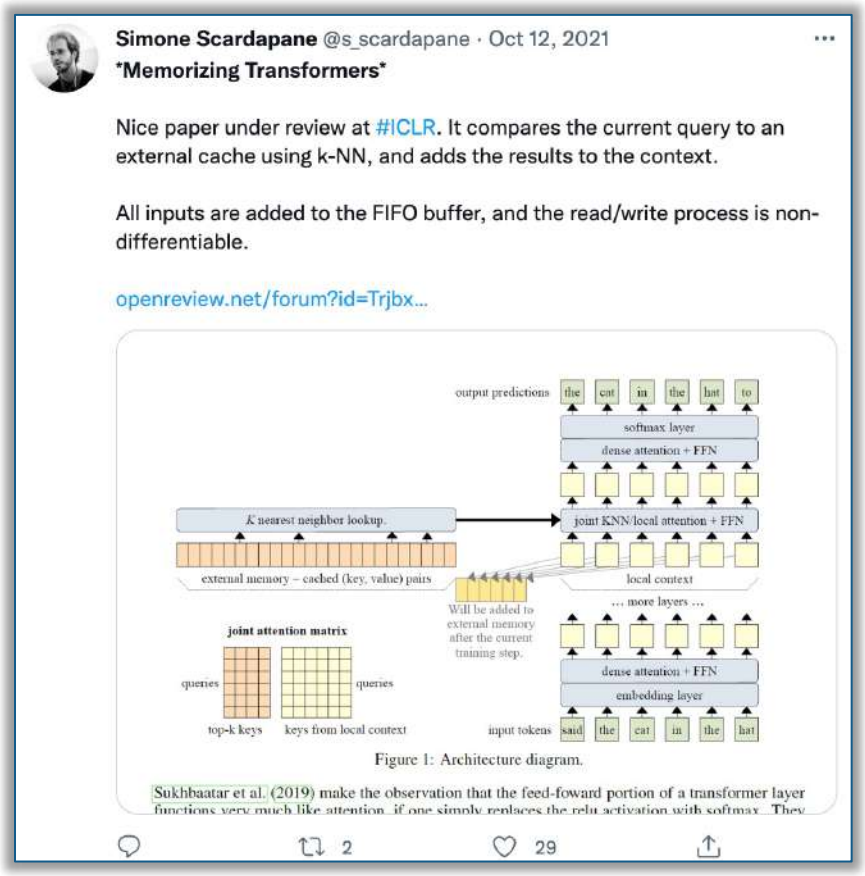
From Twitter

Jay Alammar @JayAlammar · Mar 16

This is really interesting work. I wrote about RETRO here:
jalammar.github.io/illustrated-re...

Can't wait to dive into **Memorizing Transformers**

> **Christian Szegedy** @ChrSzegedy · Mar 16
>
> To those who ask for comparison with (later) RETRO work: many differences:
>
> - MemT is a transparent, light-weight composable solution, RETRO is relatively heavyweight special-purpose system,
> - RETRO uses fixed (pretrained keys), we train retrieval E2E
>
> 1/2 twitter.com/Yuhu_ai_/statu...
>
> Show this thread



Simone Scardapane @s_scardapane · Oct 12, 2021

Memorizing Transformers

Nice paper under review at #ICLR. It compares the current query to an external cache using k-NN, and adds the results to the context.

All inputs are added to the FIFO buffer, and the read/write process is non-differentiable.

openreview.net/forum?id=Trjbx...

Figure 1: Architecture diagram.

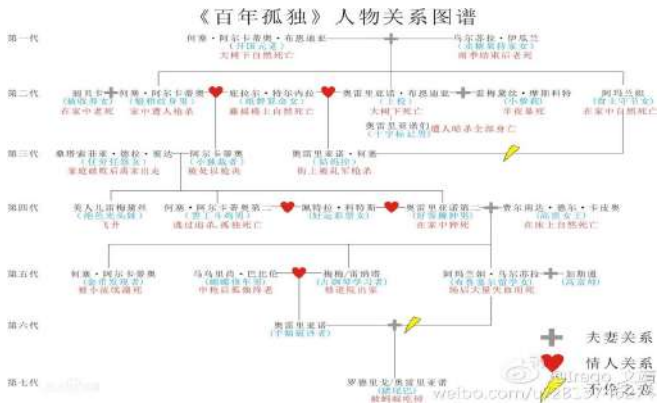
Sukhbaatar et al. (2019) make the observation that the feed-foward portion of a transformer layer functions very much like attention, if one simply replaces the relu activation with softmax. They

2 29

Transformers

- Transformer performance on many of these tasks is **limited by the context length** of attention.
- The ability to **attend to far-away tokens** is important in many situations.

Novels

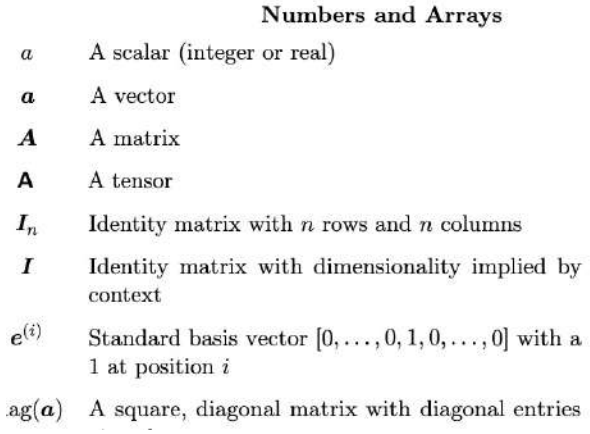


characters and events are referenced across multiple chapters

Source Code

references to classes and functions may occur quite far from the places

Theorem Proving



Numbers and Arrays

a	A scalar (integer or real)
$\boldsymbol{a}$	A vector
$\boldsymbol{A}$	A matrix
$\mathbf{A}$	A tensor
$\boldsymbol{I}_n$	Identity matrix with n rows and n columns
$\boldsymbol{I}$	Identity matrix with dimensionality implied by context
$\boldsymbol{e}^{(i)}$	Standard basis vector $[0, \dots, 0, 1, 0, \dots, 0]$ with a 1 at position i
$\text{ag}(\boldsymbol{a})$	A square, diagonal matrix with diagonal entries

proofs make use of previously defined lemmas

Why?

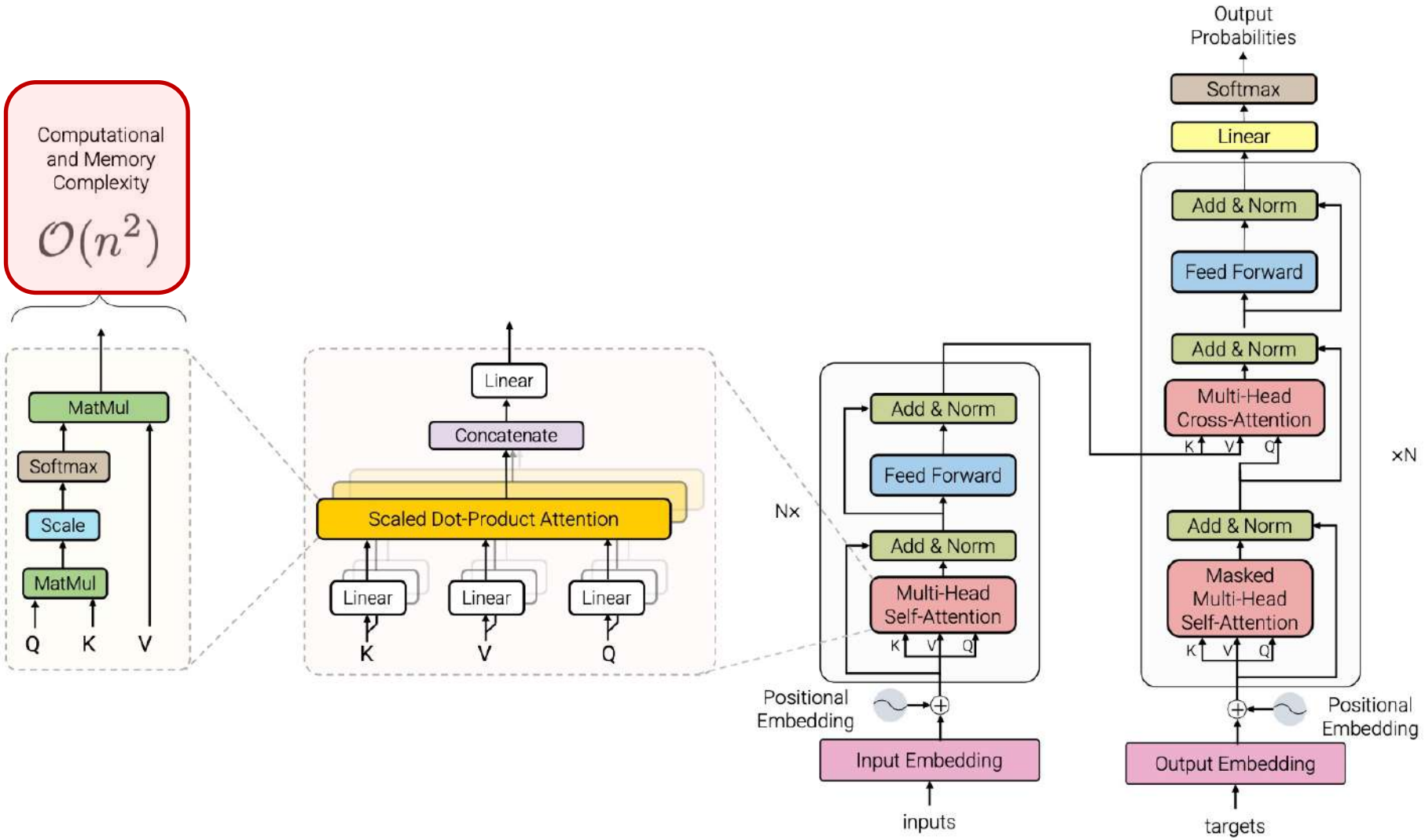
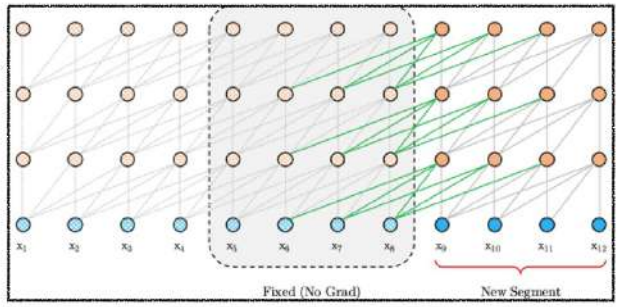


Figure 1: Architecture of the standard Transformer (Vaswani et al., 2017)

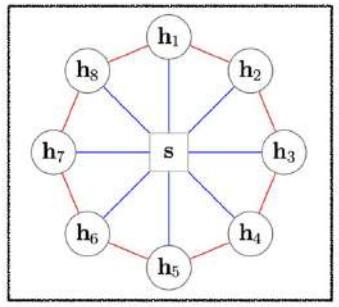
Efficient Transformers: A Survey 2022

Long-form Transformers

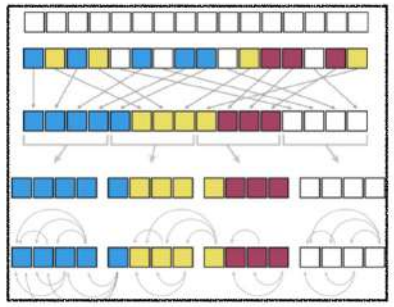
- Attention **over long sequences** is also useful as a form of rapid learning.



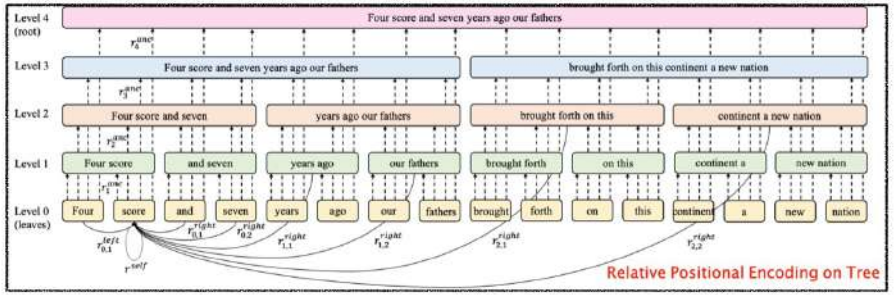
Transformer-XL



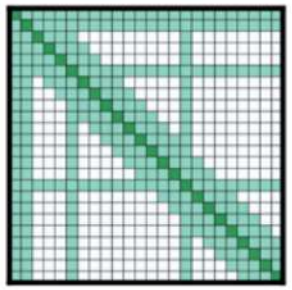
Star-Transformer



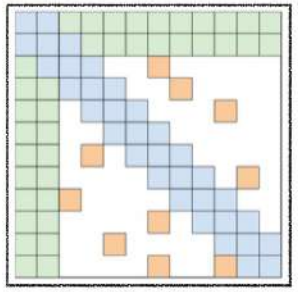
Reformer



BP-Transformer



Longformer



Big Bird

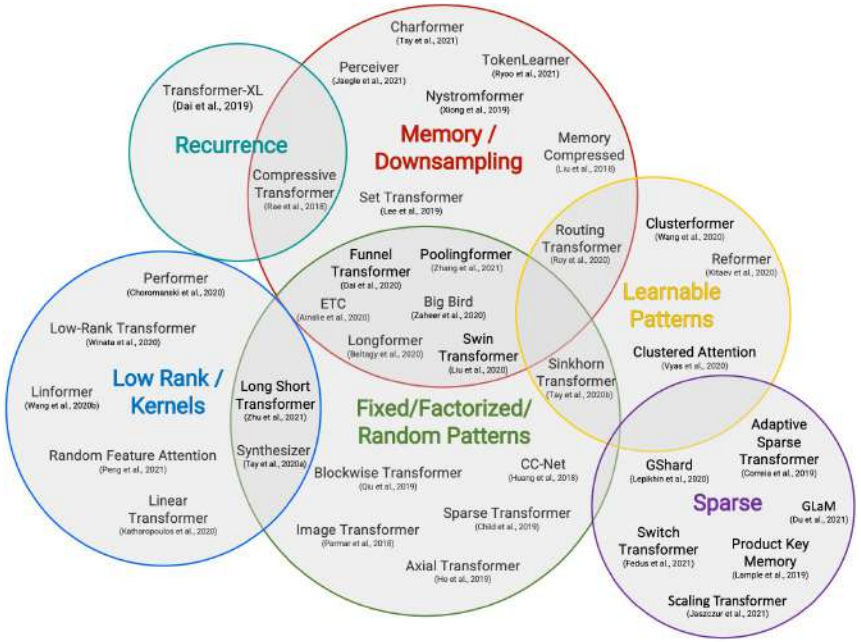


Figure 2: Taxonomy of Efficient Transformer Architectures.

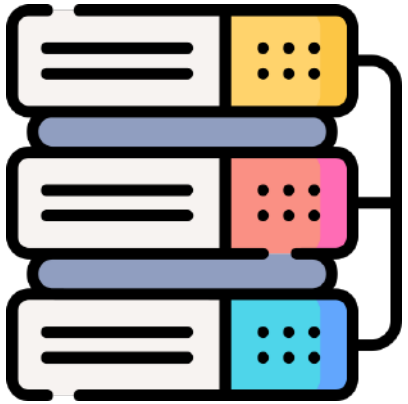
Averaging or summarization of tokens at long distances

Efficient Transformers: A Survey 2022

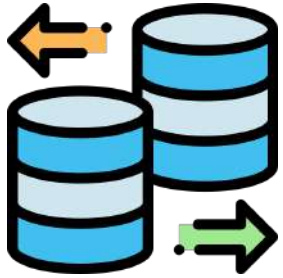
Motivation

Database

keys and values that were previously computed on prior training steps



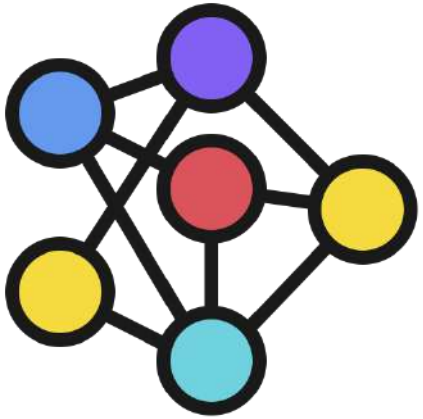
Store



Retrieve

Model

Retrieve exact values even from the distant context



kNN-augmented Transformer

kNN Augmented Attention Layer

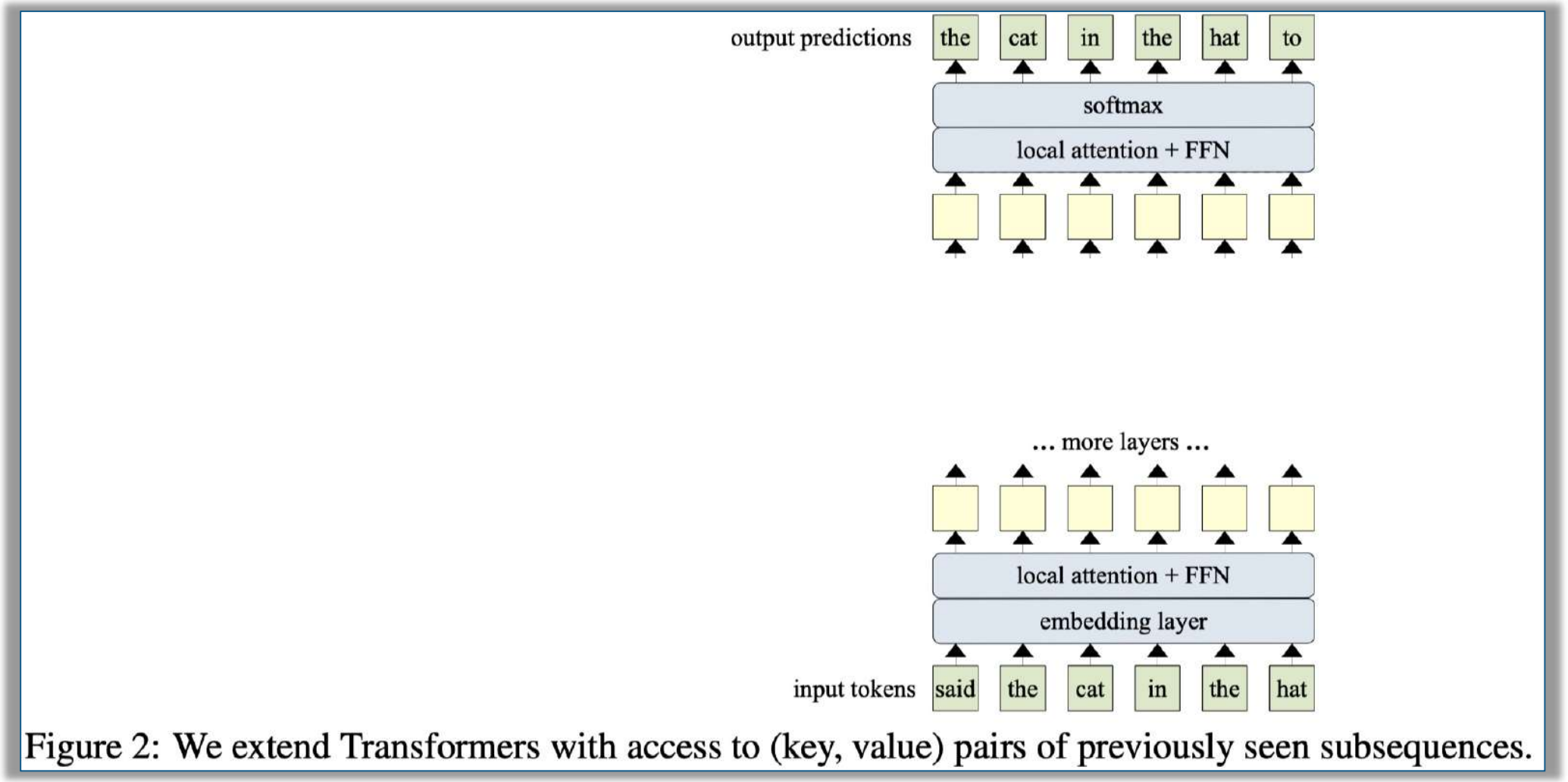


Figure 2: We extend Transformers with access to (key, value) pairs of previously seen subsequences.

kNN-augmented Transformer

kNN Augmented Attention Layer

① What is kNN attention?
② How the attention works?
③ Is the size of memory fixed?

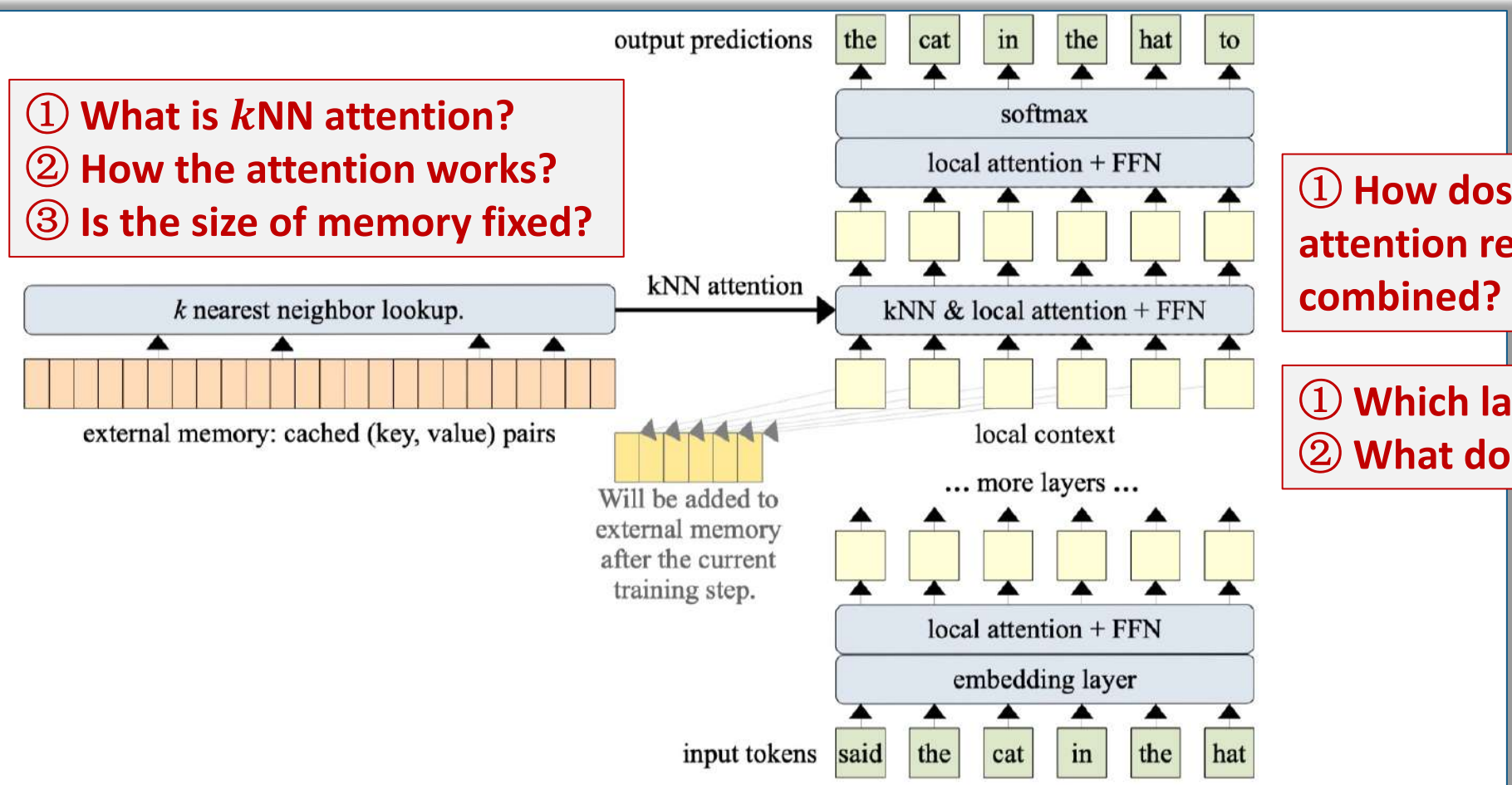


Figure 2: We extend Transformers with access to (key, value) pairs of previously seen subsequences.

① How dose both attention results combined?

① Which layer?
② What does it involve?

kNN-augmented Transformer

kNN Augmented Attention Layer

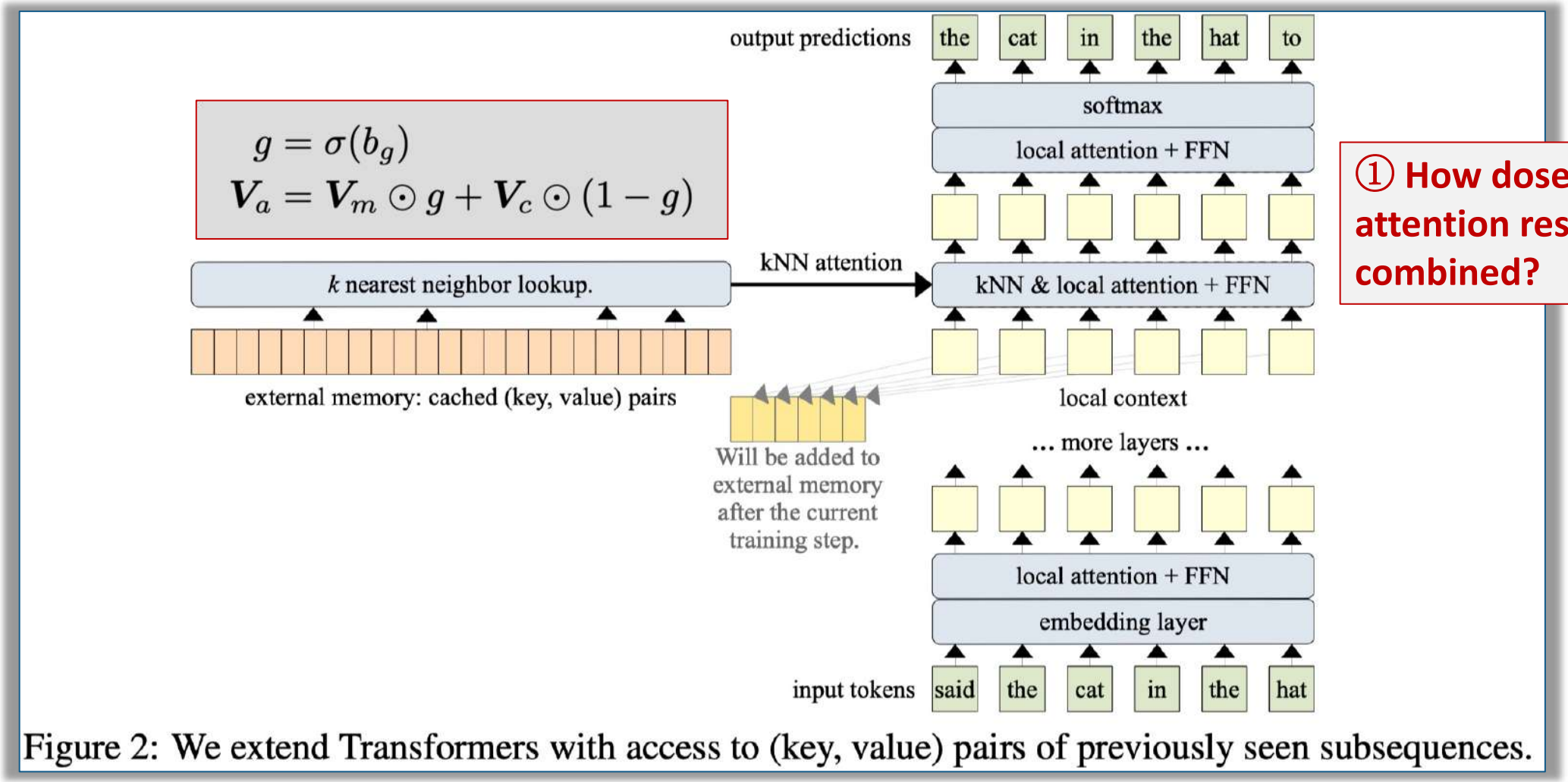


Figure 2: We extend Transformers with access to (key, value) pairs of previously seen subsequences.

① How dose both attention results combined?

Notes: Position Bias

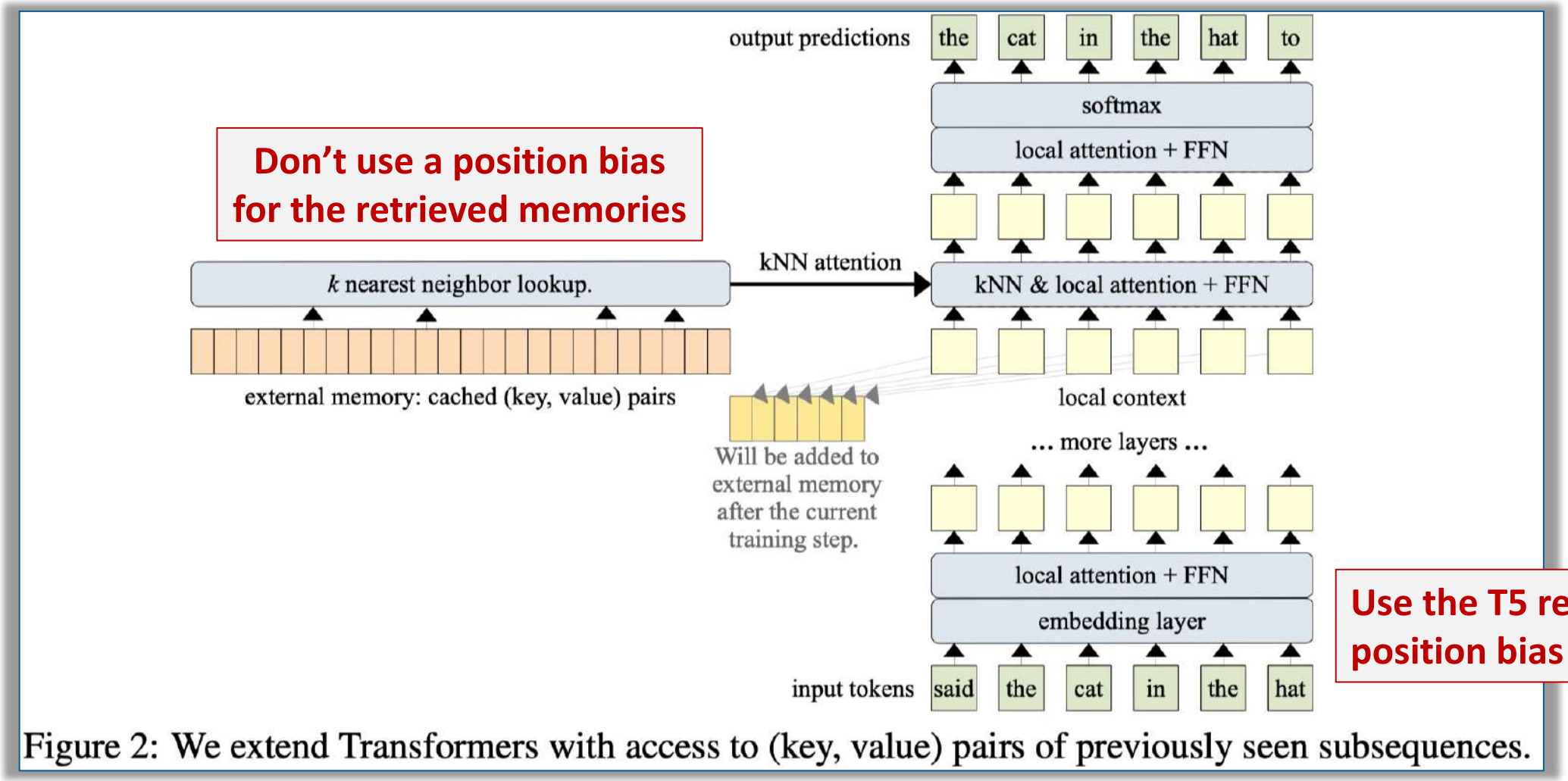


Figure 2: We extend Transformers with access to (key, value) pairs of previously seen subsequences.

Notes: Gradient

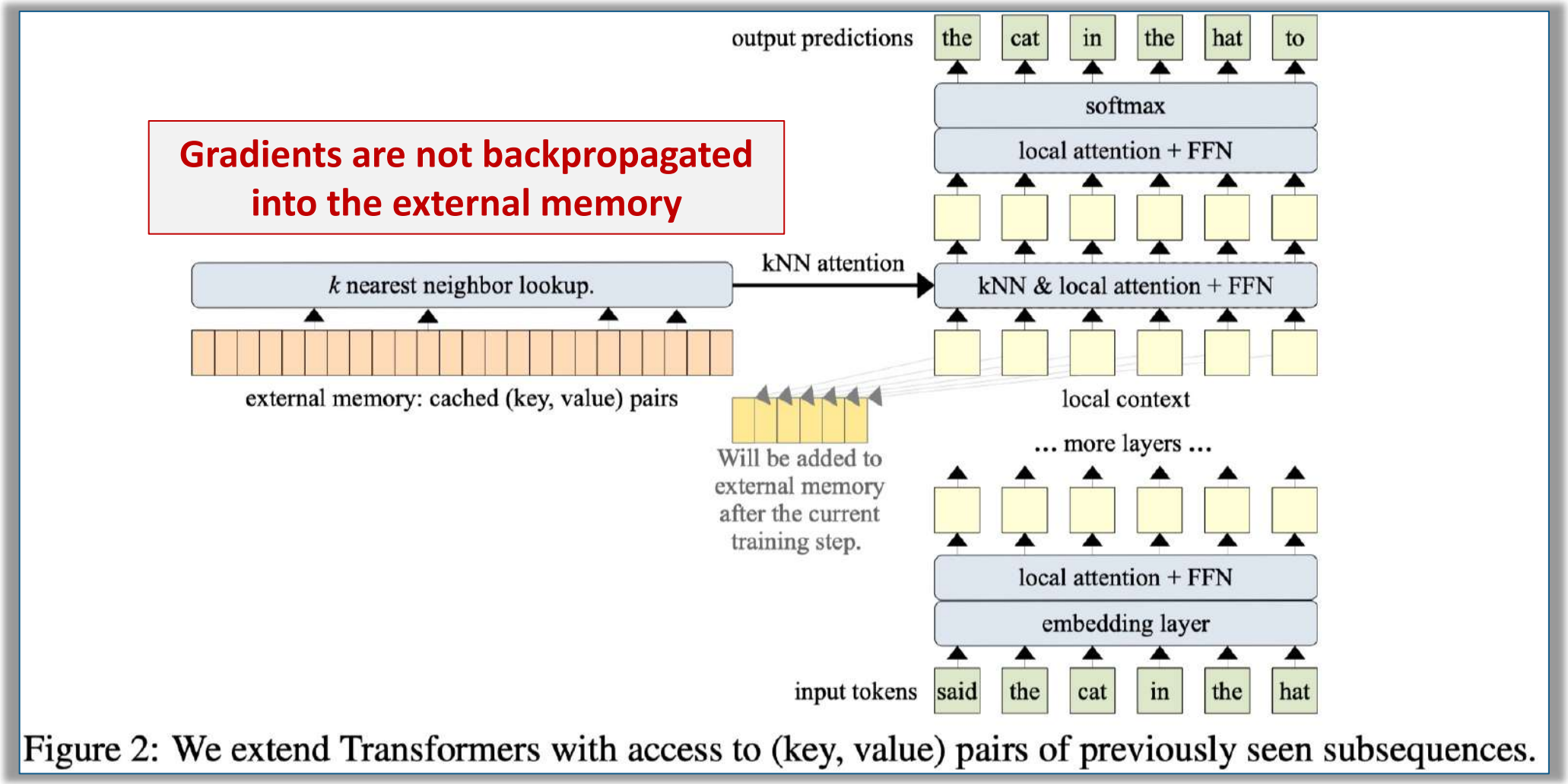


Figure 2: We extend Transformers with access to (key, value) pairs of previously seen subsequences.

Note: Batching

- Memory is **cleared** at the start of each new document.

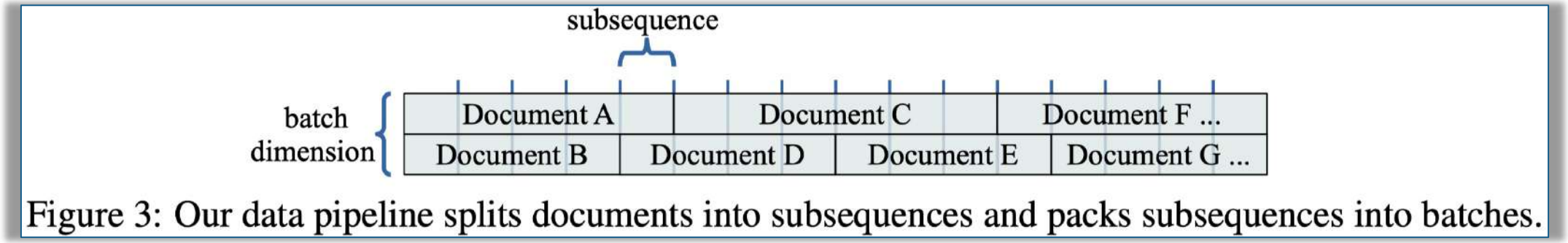


Figure 3: Our data pipeline splits documents into subsequences and packs subsequences into batches.

Note: Distributional Shift

- **Problem: staleness**
- Methods: normalize keys and queries

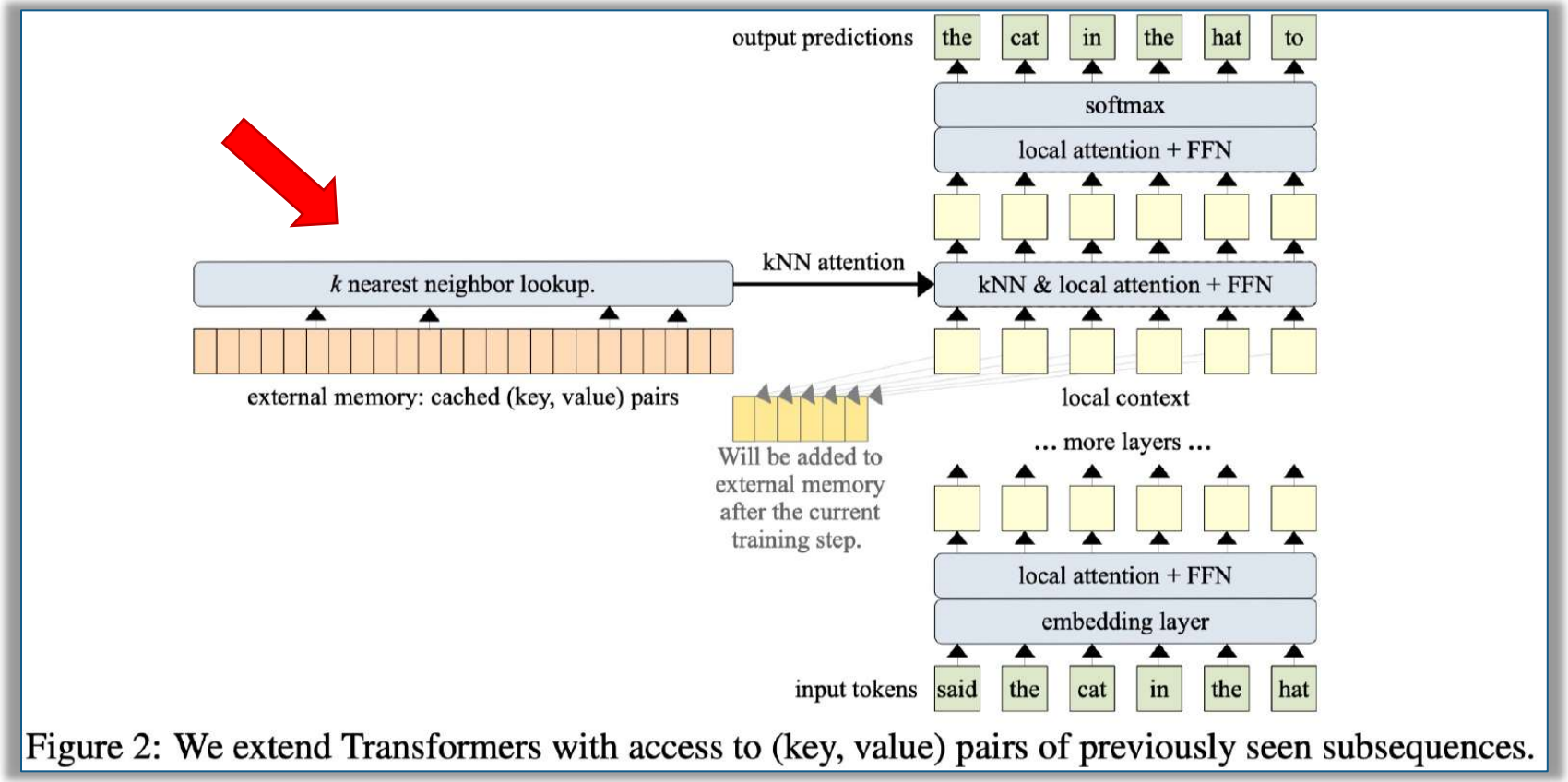


Figure 2: We extend Transformers with access to (key, value) pairs of previously seen subsequences.

Note: Approximate kNN

a) Hashing: Hashing based algorithms transform every data point to a low-dimensional representation, so each point can be represented by a shortcode called hash code. There are two main hashing sub-categories: Locality Sensitive Hashing (LSH) [22] and Learning to Hash (L2H) [23]. Locality sensitive hashing is a data-independent hashing approach. The LSH methods rely on a family of locality-sensitive hash functions that map similar input data points to the same hash codes with higher probability than different points. Random linear projections are commonly used by hash functions to generate the hash code.

b) Partition Based: Methods in this category that encapsulates Trees can be seen as dividing the entire high dimensional space into multiple disjoint subspaces. If a query "q" is located in a region Rq, then its nearest neighbors should belong in Rq or regions close to Rq. The partition process carries out recursively, building a tree structure. There are types of partition based methods: pivoting, hyperplane, and compact partitioning schemes. Pivoting methods divide the points relying on the distances from the points to some pivots. A representative example is the Ball Tree algorithm [24]. Hyperplane partitioning methods recursively divide the space by a hyperplane usually defined by a random direction. Representative examples of this category are the Annoy [25], the Random-Projection Forest [26] and the Randomized KD-trees [27] algorithms. Compact partitioning algorithms either divide the data into clusters or create approximate Voronoi partitions [28] to exploit locality.

c) Graph Based: Graph-based methods construct a proximity graph where each data point corresponds to a node, while edges connecting some of these nodes define the neighbor relationship. The core concept is that a neighbor's neighbor is also likely to be a neighbor. The search can be efficiently performed by iteratively expanding neighbors' neighbors in a best-first search strategy following the edges. Differences between the graph structures define different variations of Graph-based methods.

Recall

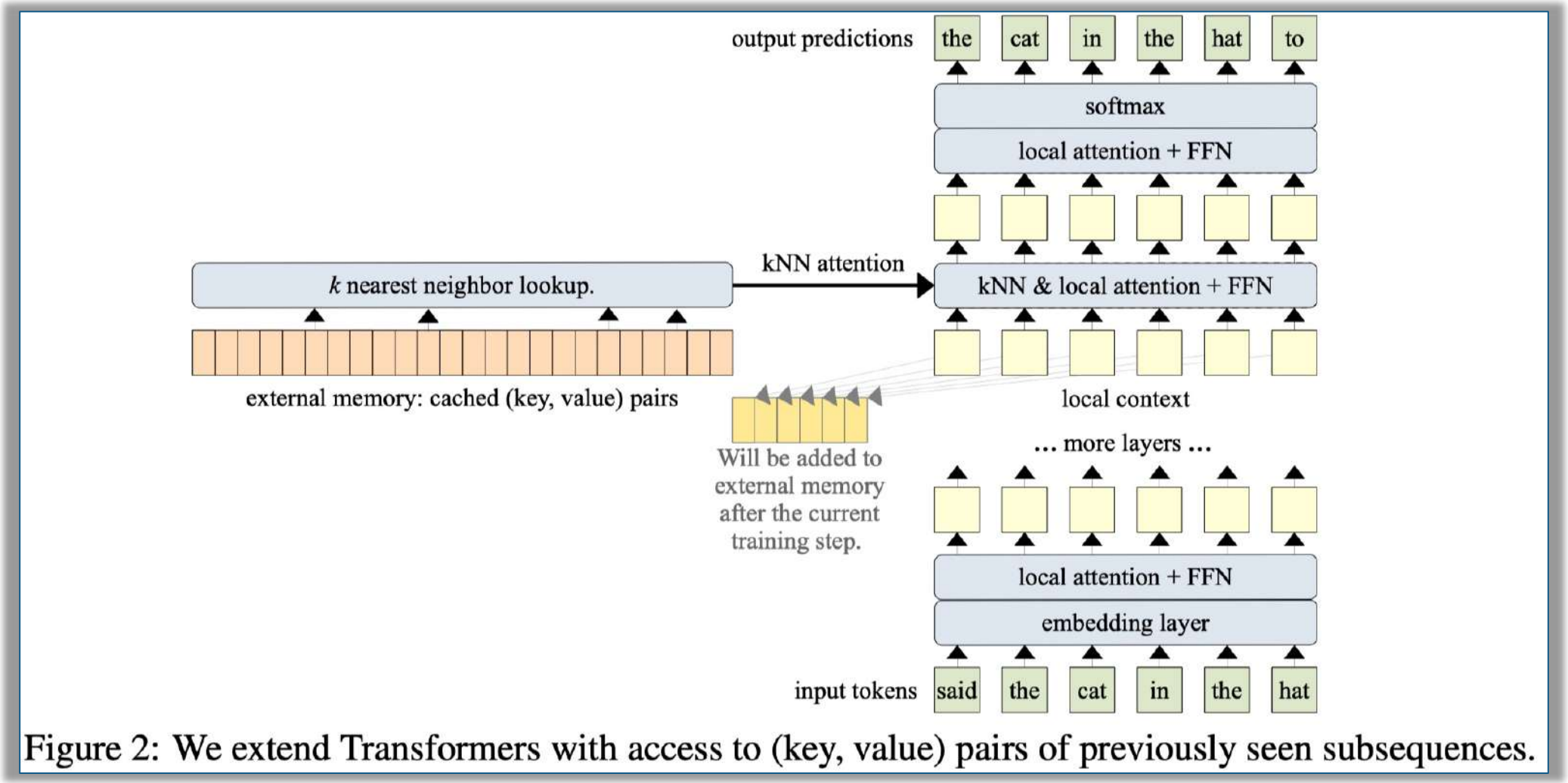


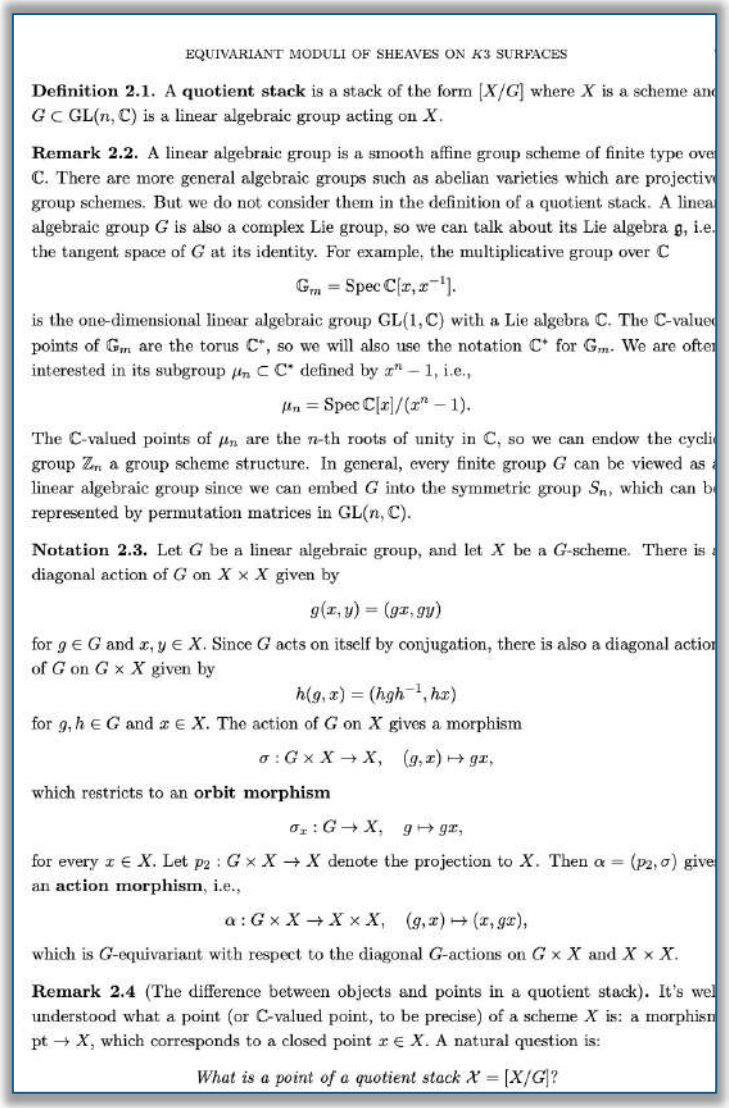
Figure 2: We extend Transformers with access to (key, value) pairs of previously seen subsequences.

Experiments

- Five **language modeling** tasks.
  - Technical math papers (arXiv Math)
  - Source code (Github)
  - Formal theorems (Isabelle)
  - Long web articles (C4)
  - English language books (PG-19)

Datasets: arXiv Math

- Downloading papers via the [arXiv Bulk Data Access]
- Include papers labeled as **"Mathematics"**
- LATEX source was available.



EQUIVARIANT MODULI OF SHEAVES ON $K3$ SURFACES

Definition 2.1. A **quotient stack** is a stack of the form $[X/G]$ where X is a scheme and $G \subset \mathrm{GL}(n, \mathbb{C})$ is a linear algebraic group acting on X.

Remark 2.2. A linear algebraic group is a smooth affine group scheme of finite type over $\mathbb{C}$. There are more general algebraic groups such as abelian varieties which are projective group schemes. But we do not consider them in the definition of a quotient stack. A linear algebraic group G is also a complex Lie group, so we can talk about its Lie algebra $\mathfrak{g}$, i.e. the tangent space of G at its identity. For example, the multiplicative group over $\mathbb{C}$

$$\mathbb{G}_m = \operatorname{Spec} \mathbb{C}[x, x^{-1}].$$

is the one-dimensional linear algebraic group $\mathrm{GL}(1, \mathbb{C})$ with a Lie algebra $\mathbb{C}$. The $\mathbb{C}$-valued points of $\mathbb{G}_m$ are the torus $\mathbb{C}^*$, so we will also use the notation $\mathbb{C}^*$ for $\mathbb{G}_m$. We are often interested in its subgroup $\mu_n \subset \mathbb{C}^*$ defined by $x^n - 1$, i.e.,

$$\mu_n = \operatorname{Spec} \mathbb{C}[x]/(x^n - 1).$$

The $\mathbb{C}$-valued points of μ_n are the n-th roots of unity in $\mathbb{C}$, so we can endow the cyclic group $\mathbb{Z}_n$ a group scheme structure. In general, every finite group G can be viewed as a linear algebraic group since we can embed G into the symmetric group S_n, which can be represented by permutation matrices in $\mathrm{GL}(n, \mathbb{C})$.

Notation 2.3. Let G be a linear algebraic group, and let X be a G-scheme. There is a diagonal action of G on $X \times X$ given by

$$g(x, y) = (gx, gy)$$

for $g \in G$ and $x, y \in X$. Since G acts on itself by conjugation, there is also a diagonal action of G on $G \times X$ given by

$$h(g, x) = (hgh^{-1}, hx)$$

for $g, h \in G$ and $x \in X$. The action of G on X gives a morphism

$$\sigma : G \times X \to X, \quad (g, x) \mapsto gx,$$

which restricts to an **orbit morphism**

$$\sigma_x : G \to X, \quad g \mapsto gx,$$

for every $x \in X$. Let $p_2 : G \times X \to X$ denote the projection to X. Then $\alpha = (p_2, \sigma)$ gives an **action morphism**, i.e.,

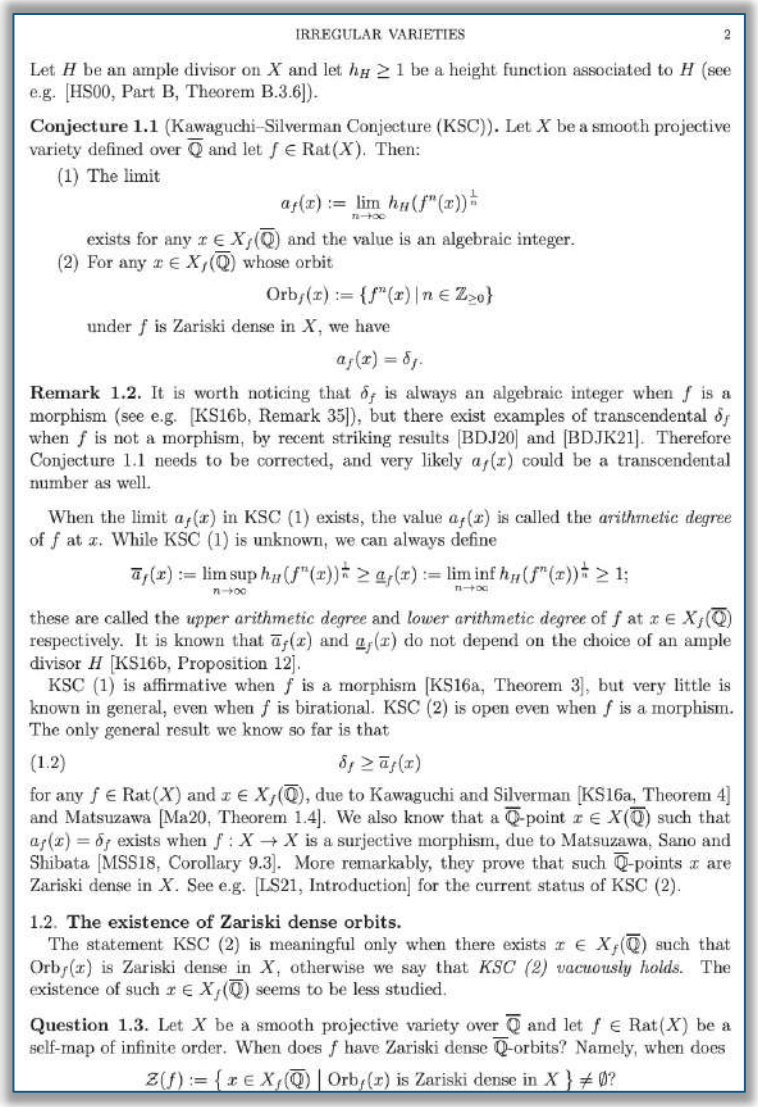
$$\alpha : G \times X \to X \times X, \quad (g, x) \mapsto (x, gx),$$

which is G-equivariant with respect to the diagonal G-actions on $G \times X$ and $X \times X$.

Remark 2.4 (The difference between objects and points in a quotient stack)**.** It's well understood what a point (or $\mathbb{C}$-valued point, to be precise) of a scheme X is: a morphism $\mathrm{pt} \to X$, which corresponds to a closed point $x \in X$. A natural question is:

What is a point of a quotient stack $\mathcal{X} = [X/G]$?

https://arxiv.org/pdf/2204.09824.pdf



IRREGULAR VARIETIES 2

Let H be an ample divisor on X and let $h_H \geq 1$ be a height function associated to H (see e.g. [HS00, Part B, Theorem B.3.6]).

Conjecture 1.1 (Kawaguchi–Silverman Conjecture (KSC))**.** Let X be a smooth projective variety defined over $\overline{\mathbb{Q}}$ and let $f \in \mathrm{Rat}(X)$. Then:

(1) The limit

$$a_f(x) := \lim_{n \to \infty} h_H(f^n(x))^{\frac{1}{n}}$$

exists for any $x \in X_f(\overline{\mathbb{Q}})$ and the value is an algebraic integer.

(2) For any $x \in X_f(\overline{\mathbb{Q}})$ whose orbit

$$\mathrm{Orb}_f(x) := \{f^n(x) \mid n \in \mathbb{Z}_{\geq 0}\}$$

under f is Zariski dense in X, we have

$$a_f(x) = \delta_f.$$

Remark 1.2. It is worth noticing that δ_f is always an algebraic integer when f is a morphism (see e.g. [KS16b, Remark 35]), but there exist examples of transcendental δ_f when f is not a morphism, by recent striking results [BDJ20] and [BDJK21]. Therefore Conjecture 1.1 needs to be corrected, and very likely $a_f(x)$ could be a transcendental number as well.

When the limit $a_f(x)$ in KSC (1) exists, the value $a_f(x)$ is called the *arithmetic degree* of f at x. While KSC (1) is unknown, we can always define

$$\overline{a}_f(x) := \limsup_{n \to \infty} h_H(f^n(x))^{\frac{1}{n}} \geq \underline{a}_f(x) := \liminf_{n \to \infty} h_H(f^n(x))^{\frac{1}{n}} \geq 1;$$

these are called the *upper arithmetic degree* and *lower arithmetic degree* of f at $x \in X_f(\overline{\mathbb{Q}})$ respectively. It is known that $\overline{a}_f(x)$ and $\underline{a}_f(x)$ do not depend on the choice of an ample divisor H [KS16b, Proposition 12].

KSC (1) is affirmative when f is a morphism [KS16a, Theorem 3], but very little is known in general, even when f is birational. KSC (2) is open even when f is a morphism. The only general result we know so far is that

$$\delta_f \geq \overline{a}_f(x) \tag{1.2}$$

for any $f \in \mathrm{Rat}(X)$ and $x \in X_f(\overline{\mathbb{Q}})$, due to Kawaguchi and Silverman [KS16a, Theorem 4] and Matsuzawa [Ma20, Theorem 1.4]. We also know that a $\overline{\mathbb{Q}}$-point $x \in X(\overline{\mathbb{Q}})$ such that $a_f(x) = \delta_f$ exists when $f : X \to X$ is a surjective morphism, due to Matsuzawa, Sano and Shibata [MSS18, Corollary 9.3]. More remarkably, they prove that such $\overline{\mathbb{Q}}$-points x are Zariski dense in X. See e.g. [LS21, Introduction] for the current status of KSC (2).

1.2. **The existence of Zariski dense orbits.**

The statement KSC (2) is meaningful only when there exists $x \in X_f(\overline{\mathbb{Q}})$ such that $\mathrm{Orb}_f(x)$ is Zariski dense in X, otherwise we say that *KSC (2) vacuously holds*. The existence of such $x \in X_f(\overline{\mathbb{Q}})$ seems to be less studied.

Question 1.3. Let X be a smooth projective variety over $\overline{\mathbb{Q}}$ and let $f \in \mathrm{Rat}(X)$ be a self-map of infinite order. When does f have Zariski dense $\overline{\mathbb{Q}}$-orbits? Namely, when does

$$\mathcal{Z}(f) := \{\, x \in X_f(\overline{\mathbb{Q}}) \mid \mathrm{Orb}_f(x) \text{ is Zariski dense in } X \,\} \neq \emptyset?$$

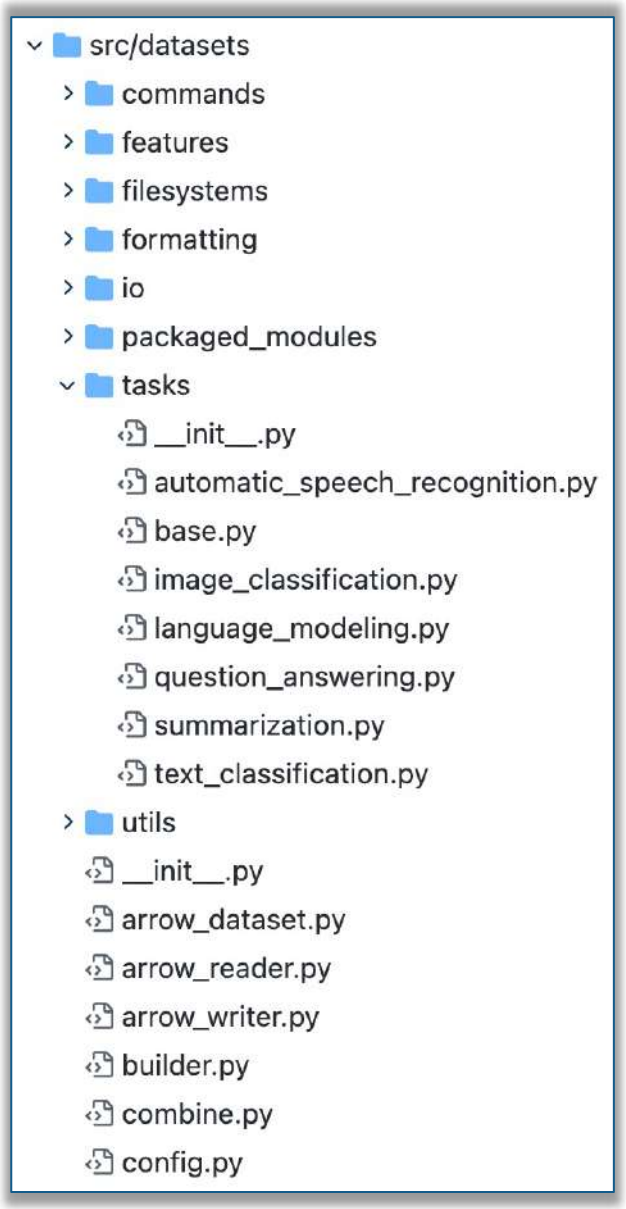
https://arxiv.org/pdf/2204.09845.pdf

Datasets: Github

- Github repositories that are published with open-source licenses.
- Use file endings to filter for files in the languages *C, C++, Java, Python, Go, and TypeScript*.
- Traversing the directory tree to create long document for each Github repository.

```
if not isinstance(self.repo_info, DatasetInfo):
    raise NotImplementedError(f"Open is only implemented for dataset repositories, but got {self.repo_info}")
url = hf_hub_url(self.repo_info.id, path, revision=self.repo_info.sha)
return fsspec.open(
    url,
    mode=mode,
    headers=get_authentication_headers_for_url(url, use_auth_token=self.token),
).open()
```

https://github.com/huggingface/datasets

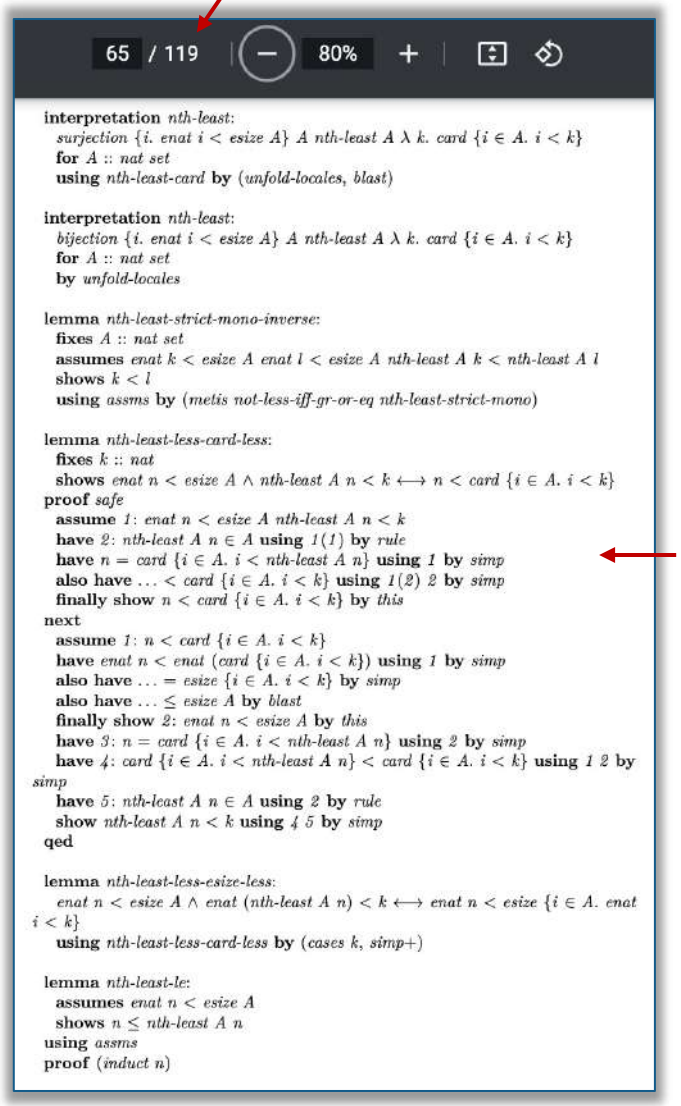


https://github.com/huggingface/datasets

Datasets: Formal Math – Isabelle

- Consists of formal mathematical proofs of theories
  - **627** theories available on The Archive of Formal Proofs
  - **57** theories from the Isabelle standard library
- Foundational logic, advanced analysis, algebra, or cryptography, and consists of multiple files containing proofs
- All files that make up a theory are concatenated together into one **long document**

119 pages in total, long document!

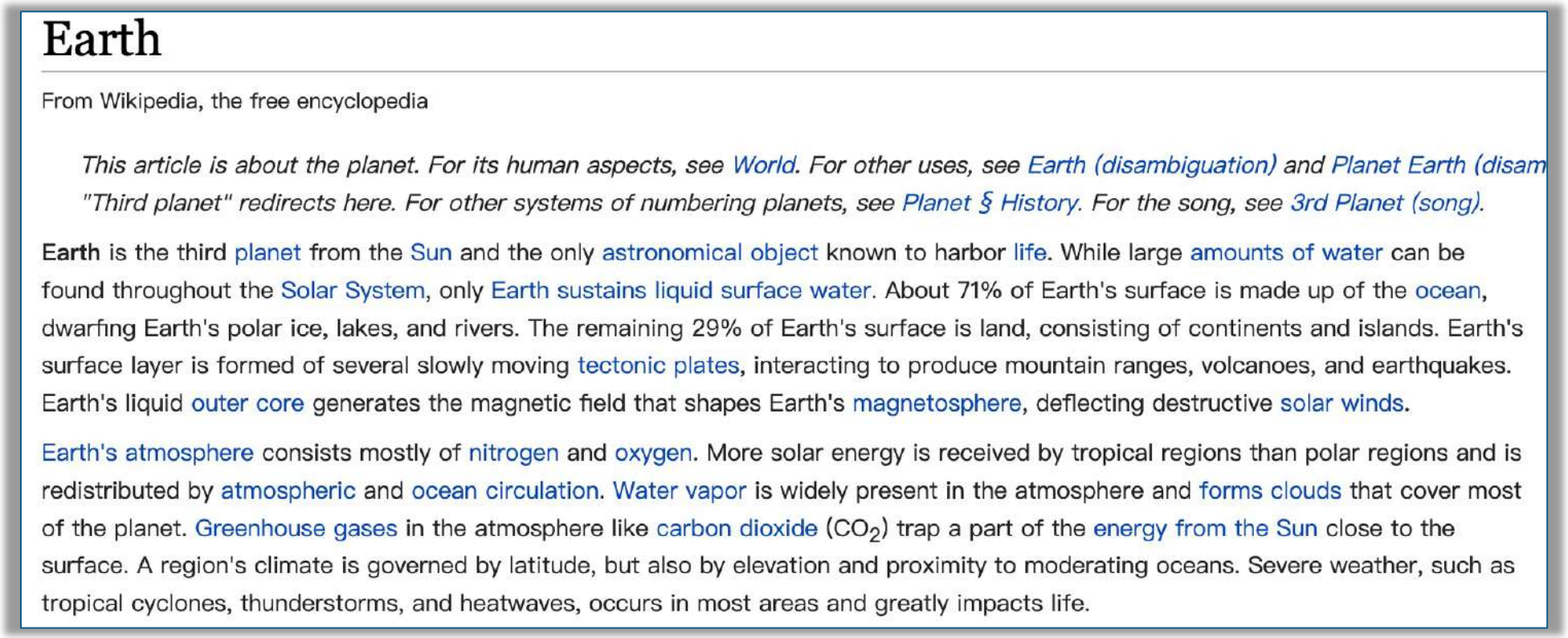


Math proofs!

https://www.isa-afp.org/browser_info/current/AFP/Partial_Order_Reduction/document.pdf

Dataset: C4(4K+)

- Cleaned common crawl.
- Filtered out all documents that have less than 4096 tokens.



Earth

From Wikipedia, the free encyclopedia

This article is about the planet. For its human aspects, see World. For other uses, see Earth (disambiguation) and Planet Earth (disam
"Third planet" redirects here. For other systems of numbering planets, see Planet § History. For the song, see 3rd Planet (song).

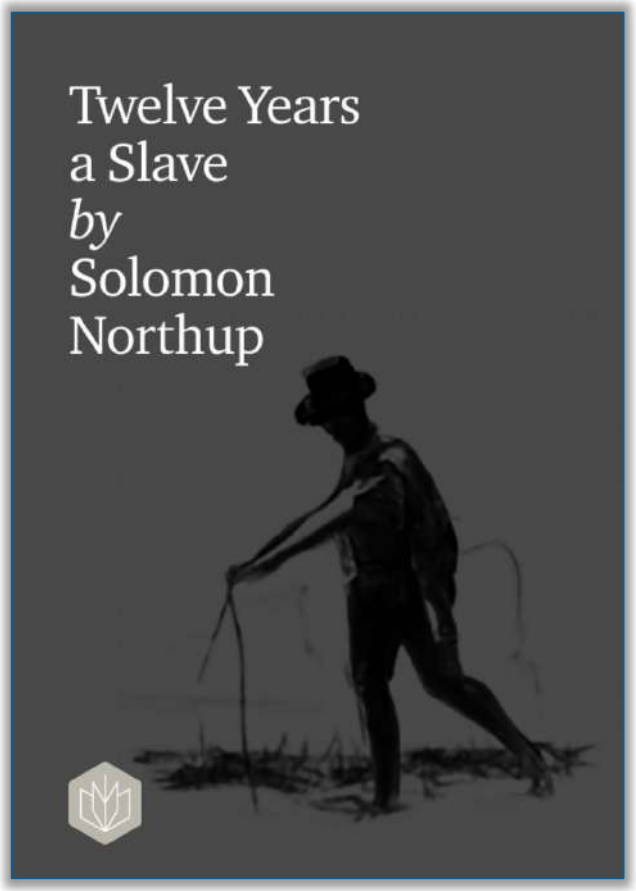
Earth is the third planet from the Sun and the only astronomical object known to harbor life. While large amounts of water can be found throughout the Solar System, only Earth sustains liquid surface water. About 71% of Earth's surface is made up of the ocean, dwarfing Earth's polar ice, lakes, and rivers. The remaining 29% of Earth's surface is land, consisting of continents and islands. Earth's surface layer is formed of several slowly moving tectonic plates, interacting to produce mountain ranges, volcanoes, and earthquakes. Earth's liquid outer core generates the magnetic field that shapes Earth's magnetosphere, deflecting destructive solar winds.

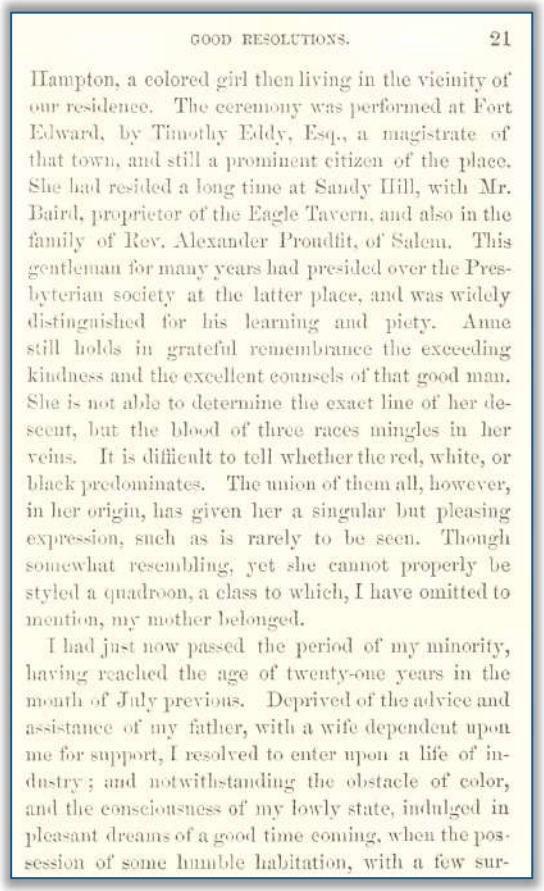
Earth's atmosphere consists mostly of nitrogen and oxygen. More solar energy is received by tropical regions than polar regions and is redistributed by atmospheric and ocean circulation. Water vapor is widely present in the atmosphere and forms clouds that cover most of the planet. Greenhouse gases in the atmosphere like carbon dioxide (CO_2) trap a part of the energy from the Sun close to the surface. A region's climate is governed by latitude, but also by elevation and proximity to moderating oceans. Severe weather, such as tropical cyclones, thunderstorms, and heatwaves, occurs in most areas and greatly impacts life.

https://en.wikipedia.org/wiki/Earth

Dataset: PG-19

- English-language books





GOOD RESOLUTIONS. 21

Hampton, a colored girl then living in the vicinity of our residence. The ceremony was performed at Fort Edward, by Timothy Eddy, Esq., a magistrate of that town, and still a prominent citizen of the place. She had resided a long time at Sandy Hill, with Mr. Baird, proprietor of the Eagle Tavern, and also in the family of Rev. Alexander Proudfit, of Salem. This gentleman for many years had presided over the Presbyterian society at the latter place, and was widely distinguished for his learning and piety. Anne still holds in grateful remembrance the exceeding kindness and the excellent counsels of that good man. She is not able to determine the exact line of her descent, but the blood of three races mingles in her veins. It is difficult to tell whether the red, white, or black predominates. The union of them all, however, in her origin, has given her a singular but pleasing expression, such as is rarely to be seen. Though somewhat resembling, yet she cannot properly be styled a quadroon, a class to which, I have omitted to mention, my mother belonged.

I had just now passed the period of my minority, having reached the age of twenty-one years in the month of July previous. Deprived of the advice and assistance of my father, with a wife dependent upon me for support, I resolved to enter upon a life of industry; and notwithstanding the obstacle of color, and the consciousness of my lowly state, indulged in pleasant dreams of a good time coming, when the possession of some humble habitation, with a few sur-

https://english.hku.hk/staff/kjohnson/PDF/Northup12YEARS1853.pdf
https://openai.com/blog/summarizing-books/

Model

- **12-layer decoder-only** transformer (with and without Transformer-XL cache)
- Embedding size of 1024, 8 attention heads of dimension 128, and an FFN hidden layer of size 4096
- $k = 32$
- **9th layer as the kNN augmented attention layer**
- Experiments on 32 TPU cores.
- Always 2^{17} **tokens in a batch**
  - a model with a context length of 512 has a batch size of 256
  - the 2048 model has a batch size of 64.
- For a model with around 200M trainable parameters the step time increased from **0.2s to 0.25s** when we added a memory of size 8K, and to **0.6s** when we added a memory of size 65K (measured on TPUv3).

Effect of External Memory

- **Adding external memory** results in substantial gains across datasets and architectures
- **Increasing the size** of the memory increases the benefit of the memory.

Context	Memory	XL cache	arXiv	PG19	C4(4K+)	GitHub	Isabelle
512	None	None	3.29	13.71	17.20	3.05	3.09
2048	None	None	2.69	12.37	14.81	2.22	2.39
512	None	512	2.67	12.34	15.38	2.26	2.46
2048	None	2048	2.42	11.88	14.03	2.10	2.16
512	1536	None	2.61	12.50	14.97	2.20	2.33
512	8192	None	2.49	12.29	14.42	2.09	2.19
512	8192	512	2.37	11.93	14.04	2.03	2.08
512	65K	512	2.31	11.62	14.04	1.87	2.06
2048	8192	2048	2.33	11.84	13.80	1.98	2.06
2048	65K	2048	**2.26**	**11.37**	**13.64**	**1.80**	**1.99**

Table 4: Average token-level perplexities of each model when trained for 500k steps.

Effect of External Memory

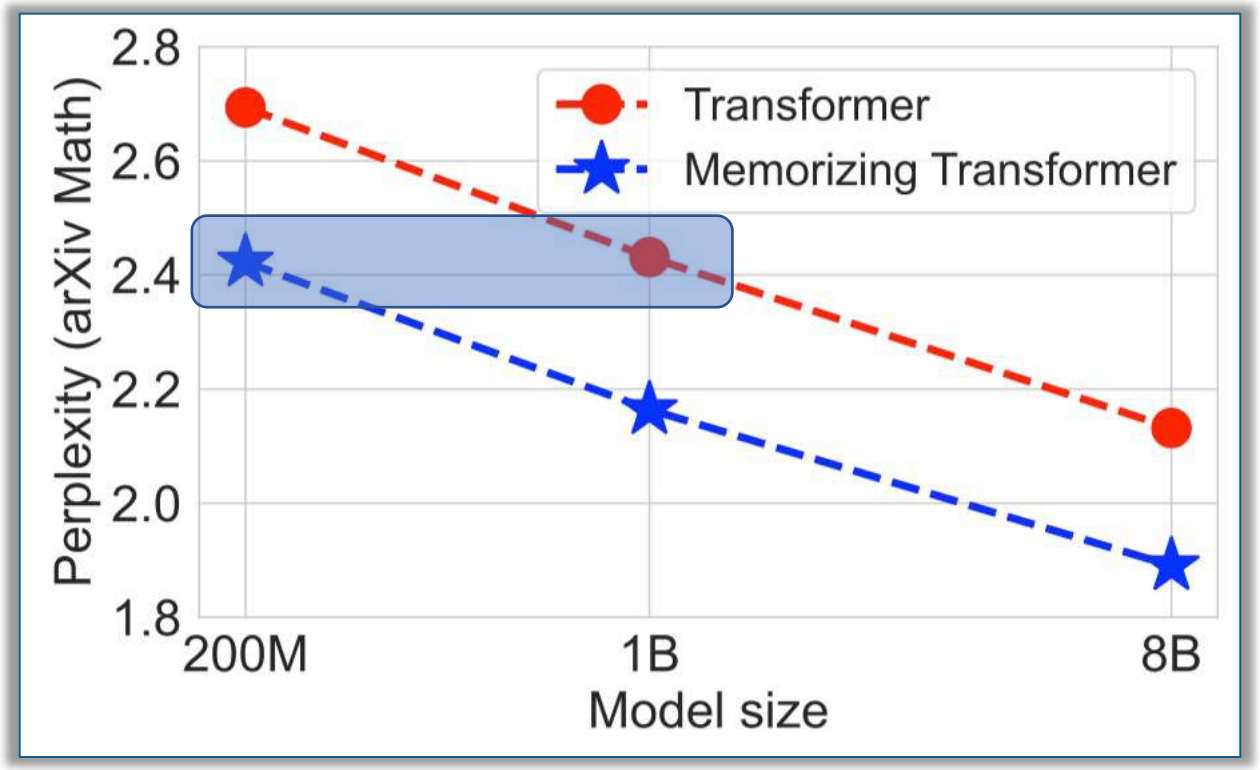
- The lower layers of a Transformer don't necessarily need long-range context, and having a differentiable memory is not as important as one might suspect.

Context	Memory	XL cache	arXiv	PG19	C4(4K+)	GitHub	Isabelle
512	None	None	3.29	13.71	17.20	3.05	3.09
2048	None	None	2.69	12.37	14.81	2.22	2.39
512	None	512	2.67	12.34	15.38	2.26	2.46
2048	None	2048	2.42	11.88	14.03	2.10	2.16
512	1536	None	2.61	12.50	14.97	2.20	2.33
512	8192	None	2.49	12.29	14.42	2.09	2.19
512	8192	512	2.37	11.93	14.04	2.03	2.08
512	65K	512	2.31	11.62	14.04	1.87	2.06
2048	8192	2048	2.33	11.84	13.80	1.98	2.06
2048	65K	2048	**2.26**	**11.37**	**13.64**	**1.80**	**1.99**

Table 4: Average token-level perplexities of each model when trained for 500k steps.

Scaling to Larger Models

- the smaller Memorizing Transformer with just 8k tokens in memory can match the perplexity of a larger vanilla Transformer which has 5X more trainable parameters.



Adding a memory of 8K tokens improves perplexity across different model sizes.

Finetuning on Larger Memories

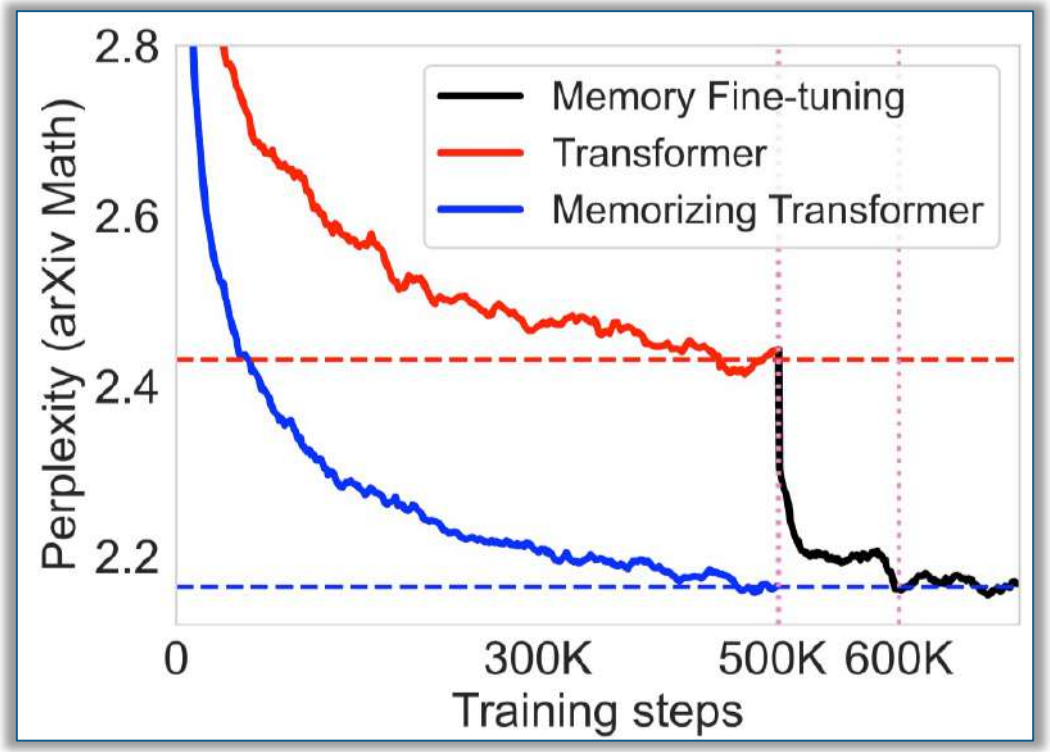
- **Pretrain**:the model with a memory size of 8192 or 65K for 500K steps,
- **Finetune:** with the larger memory for an additional 20K steps.

Context	Pretrain	Fine-tune	Perplexity
512	8192	None	2.37
512	65K	None	2.31
512	8192	65K	2.32
512	8192	131K	2.30
512	8192	262K	2.26
2048	8192	None	2.33
2048	65K	None	2.26
2048	65K	131K	2.23
2048	65K	262K	**2.21**

Table 5: Finetuning for 20K steps to make use of a larger memory on the arXiv data set.

Finetuning a Non-memory Model to Use Memory

- Within 20K steps (4% of the pre-training time) the fine-tuned model has already closed 85% of the gap between it and the 1B Memorizing Transformer, and after 100k steps it has closed the gap entirely.



Finetuning a 1B vanilla Transformer model to use external memory of size 65K

Information Retrieval Patterns

- **Which tokens show a benefit from memory?**
  - After token 8193, we can see that the larger memory helps
  - The improvement in perplexity seems to be mainly driven by a small percentage of tokens that obtain a large improvement in cross-entropy loss when using the larger memory.

$$\Delta_i = \text{cross} - \text{entropy}_{\mathbf{8192}}\ (x_i) - \text{cross} - \text{entropy}_{\mathbf{32K}}\ (x_i)$$

Positive values show an improvement in loss.

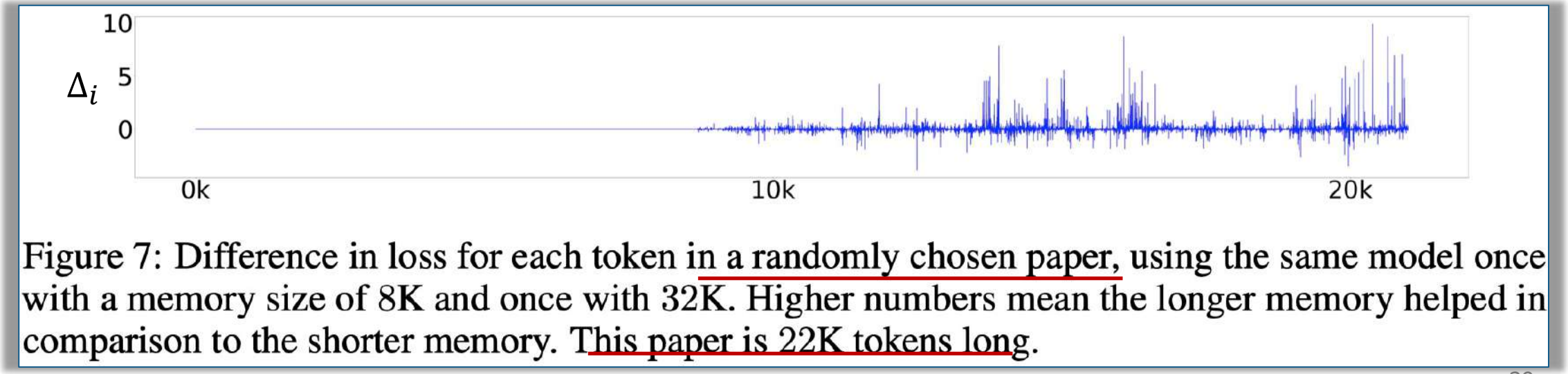


Figure 7: Difference in loss for each token in a randomly chosen paper, using the same model once with a memory size of 8K and once with 32K. Higher numbers mean the longer memory helped in comparison to the shorter memory. This paper is 22K tokens long.

Information Retrieval Patterns

- **What information is being looked up?**
  - For arXiv math and Github, the model retrieved function and variable names.
  - Our case study on the Isabelle corpus provides one of the clearest illustrations of how a model can make good use of external memory.

Query index	Input	Target	Surrounding context	Retrieved index	Retrieved surrounding context
29721	mark	ov	rule prob_space. markov_inequality	8088	M. t \<le> X a} \<le> expectation X / t"
40919	_	th	= (subgraph_threshold H n / p n)	27219	threshold H n = n powr (-(1 / max_density'
49699	S	w	assumes " orthonormal_system S w"	28050	definition orthonormal_system :: "

Table 8: Examples of memory retrieval in the Isabelle dataset. The model is able to find the definition of a lemma from a reference to it. The retrieved surrounding context (highlighted) is the definition body of the mathematical object highlighted in the querying context.

Conclusions && Thoughts

- kNN-augmented attention for Transformer
- Good Performance
- **Potentially able to leverage vast knowledge bases or code repositories**

[−] **Paper Decision**

ICLR 2022 Conference Program Chairs

21 Jan 2022 ICLR 2022 Conference Paper4131 Decision Readers: Everyone

Decision: Accept (Spotlight)

Comment: This paper studies the problem of dealing with long contexts within a Transformer architecture.
The key contribution is a kNN memory module that works in concert with a Transformer by integrating upper layers with additional retrieved context.

The idea is simple but the execution is good. While the idea is reminiscent of other recent work on this topic, and novelty is somewhat borderline, it is practically useful.
Overall, though ambivalent, my recommendation is that the paper should probably be accepted

Scores: 8 6 6 5

Thanks~