Memorization in Models

Yufang Liu

East China Normal University

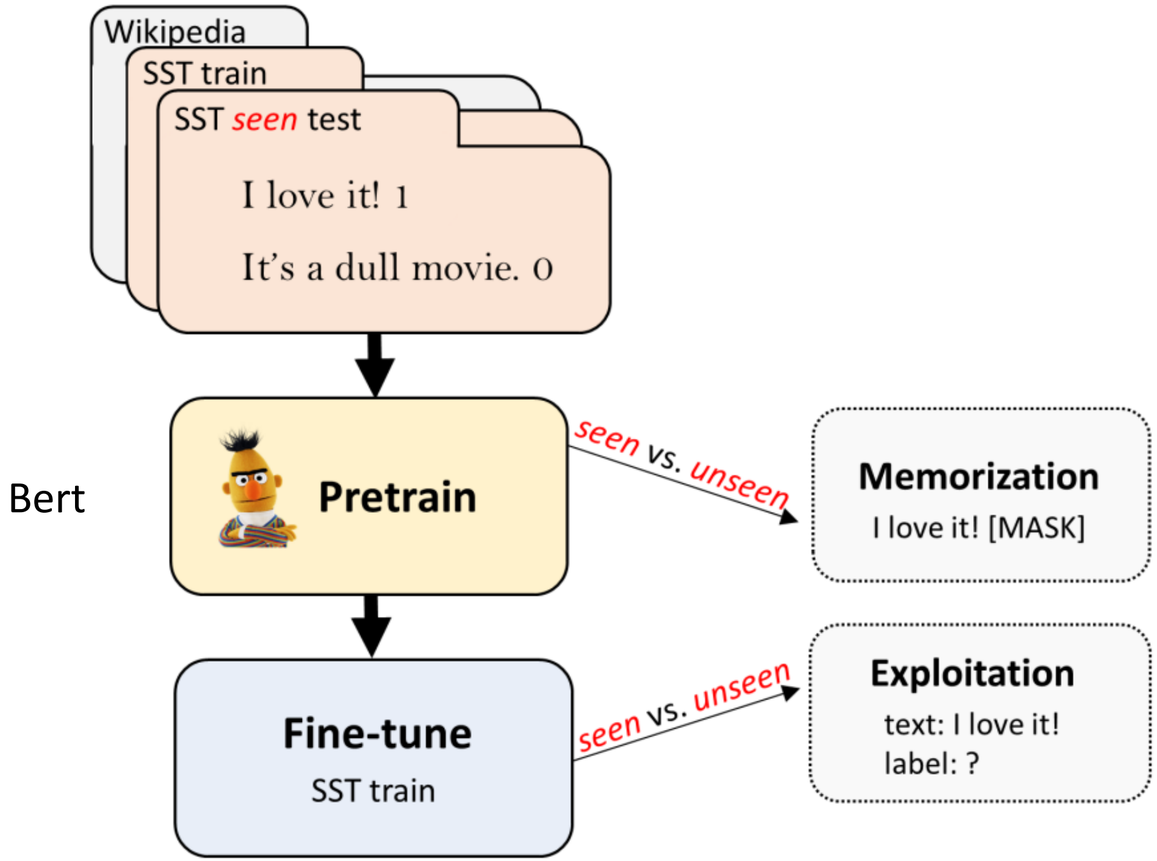
Data Contamination: From Memorization to Exploitation

Inbal Magar **Roy Schwartz**
School of Computer Science and Engineering, The Hebrew University of Jerusalem, Israel
{inbal.magar,roy.schwartz1}@mail.huji.ac.il

ACL 2022 short

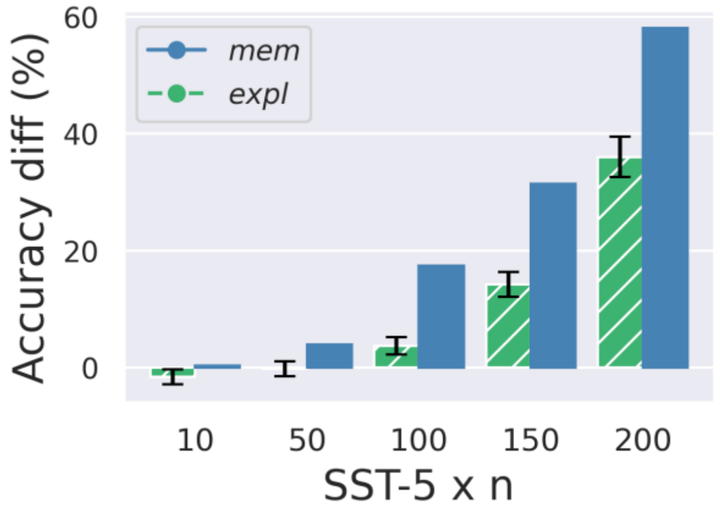
Motivation

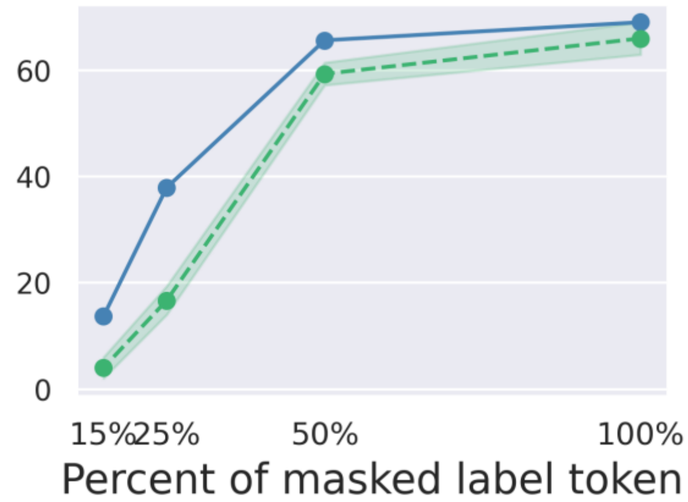
- web-based datasets are often “contaminated” with downstream test sets
- what extent models exploit the contaminated data for downstream tasks



Which Factors Affect Exploitation

- English Wikipedia (60M tokens)
  + 3 subsets SST-5 (1000)
- Pretrain one epoch





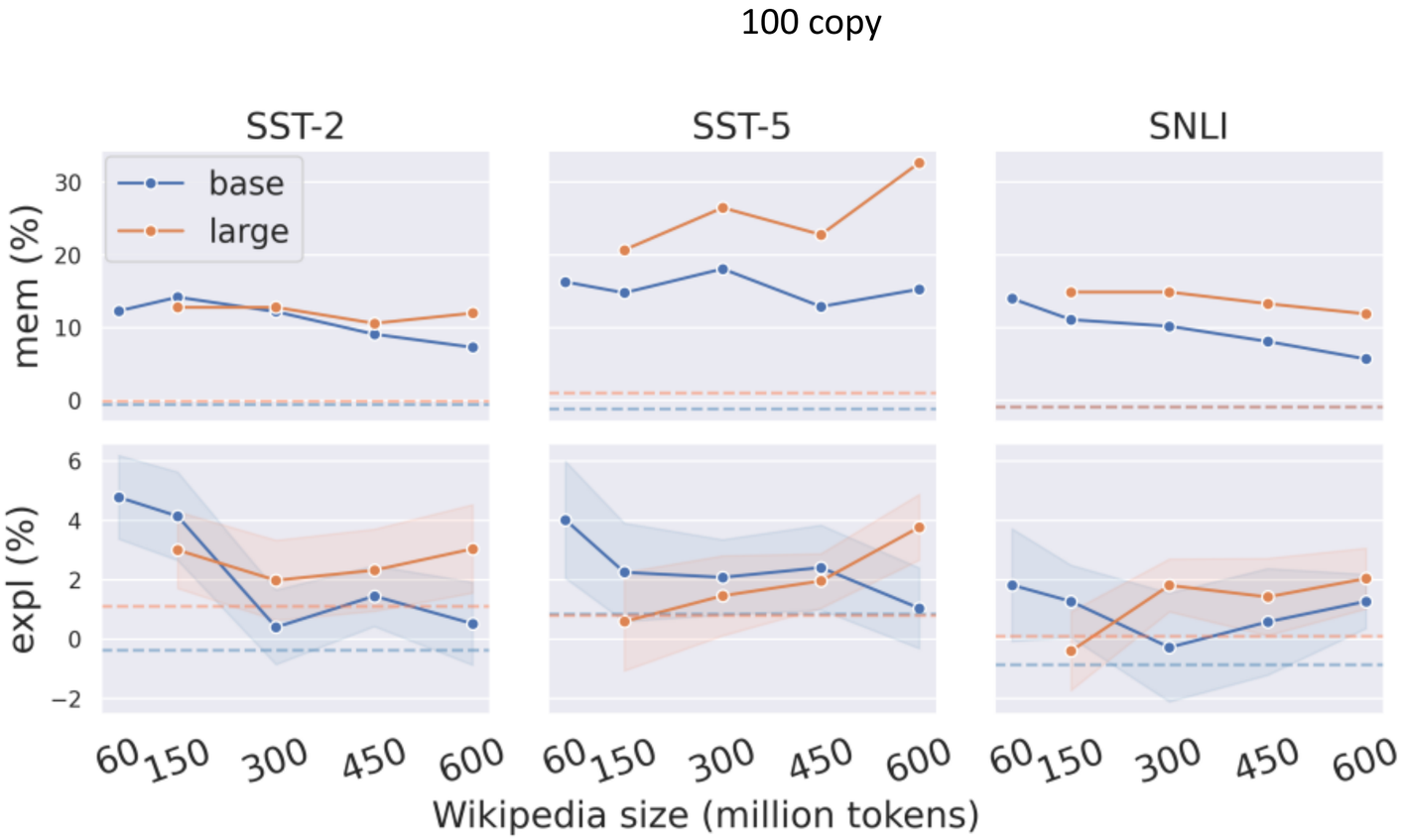
100 copy

Exploitation grows with contaminated data duplicates

higher mask probability, the higher memorization and exploitation value

Which Factors Affect Exploitation

- Memorization does not guarantee exploitation
- Model and corpus sizes matter
  - larger model -> higher mem
  - larger model -> increasing expl, benefit more from additional data



Dotted lines are mem/expl baselines of BERT-{base,large} pretrained on uncontaminated data

Which Factors Affect Exploitation

- contamination time and LR affects
  - Early contamination leads to high exploitation
  - in early contamination mem levels are lower then expl -> implicit and explicit memory

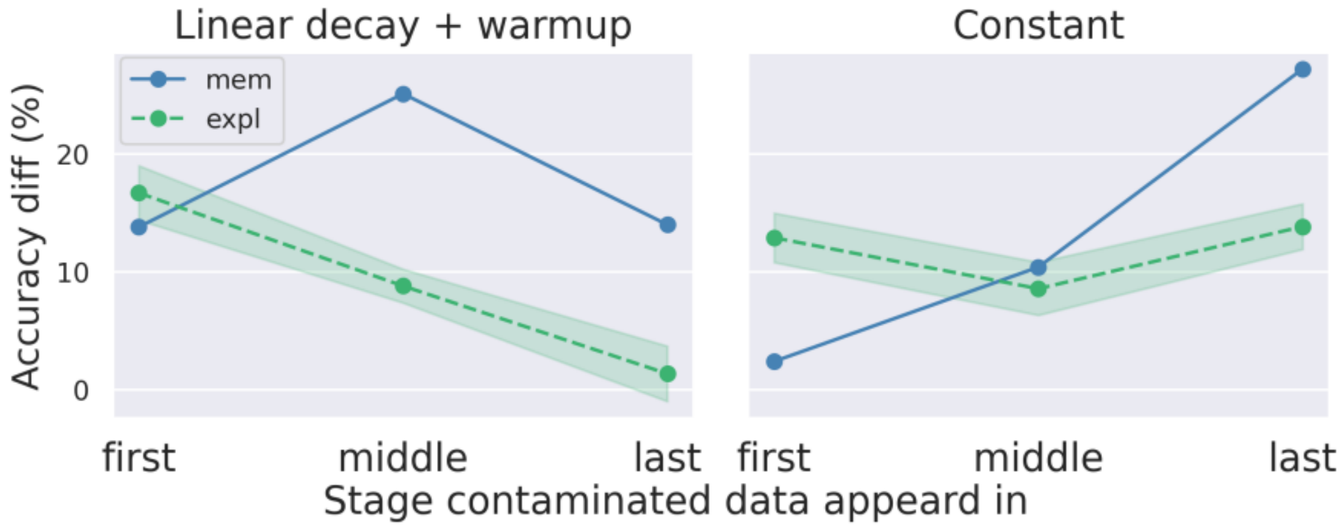
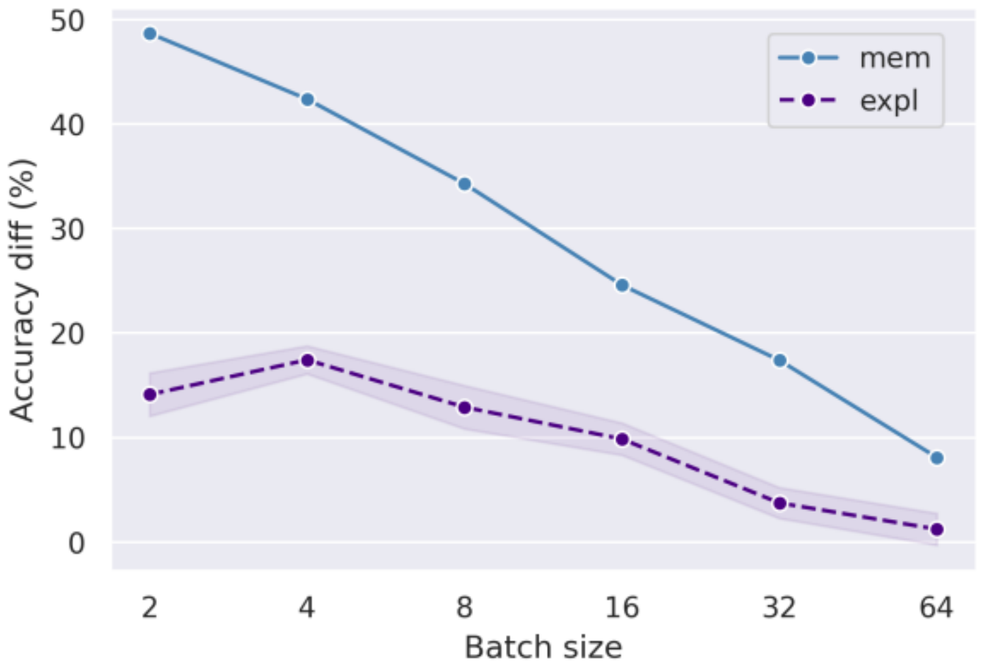


Figure 4: SST-5 `mem` and `expl` when contamination is inserted in different stages of pretraining, using a linear learning rate decay, and a constant learning rate.

Which Factors Affect Exploitation



Large batch size during pretraining reduces exploitation.

Conclusion

- memorization and exploitation are not highly connected
- affected by different factors, such as the number of duplications and the model size
- the importance of analyzing massive web-scale dataset

Memorization Without Overfitting: Analyzing the Training Dynamics of Large Language Models

Kushal Tirumala* **Aram H. Markosyan*** **Luke Zettlemoyer** **Armen Aghajanyan**

Meta AI Research
{ktirumala,amarkos,lsz,armenag}@fb.com

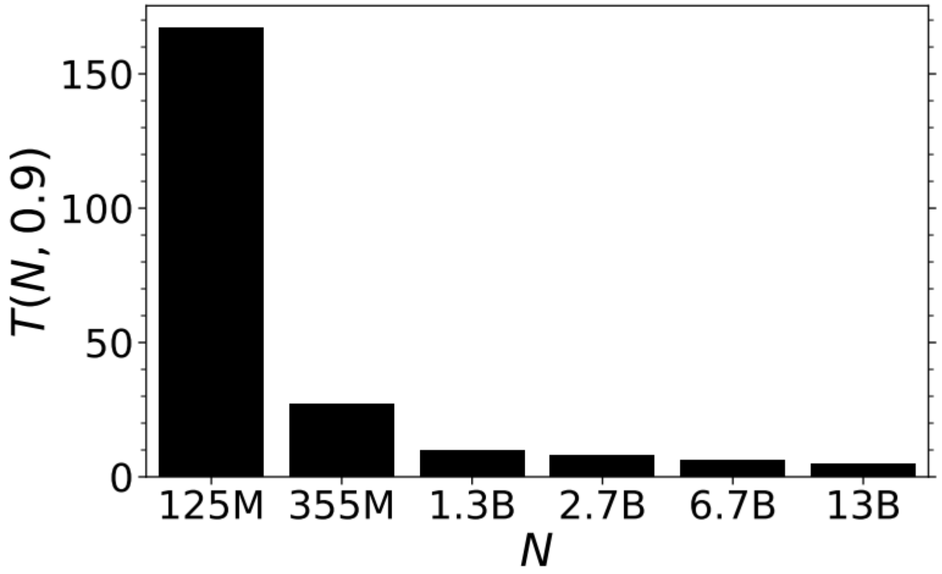
NeurIPS *2022*

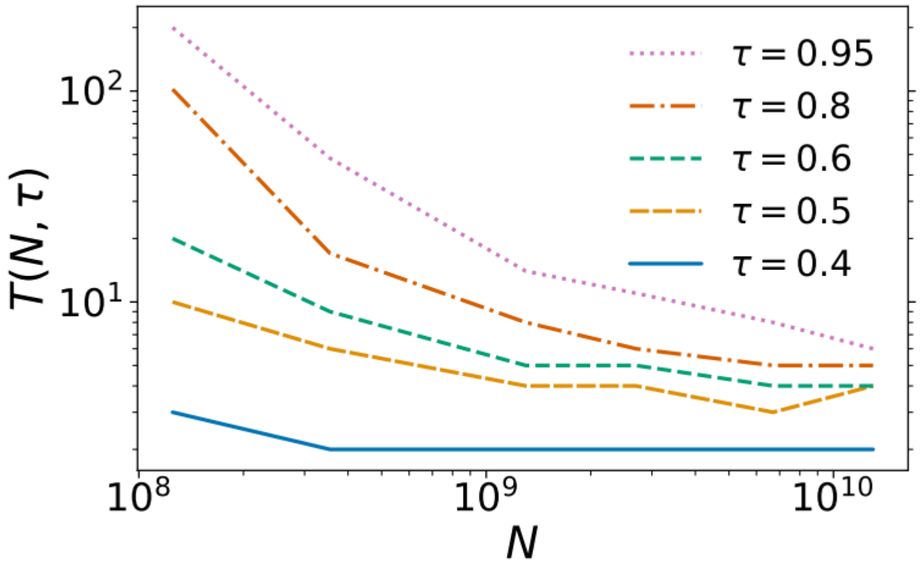
Motivation

- what is the dependence of memorization dynamics over training?
- how language models naturally forget memories throughout training?

Larger Language Models Memorize Faster

- Casual language model + WIKITEXT103



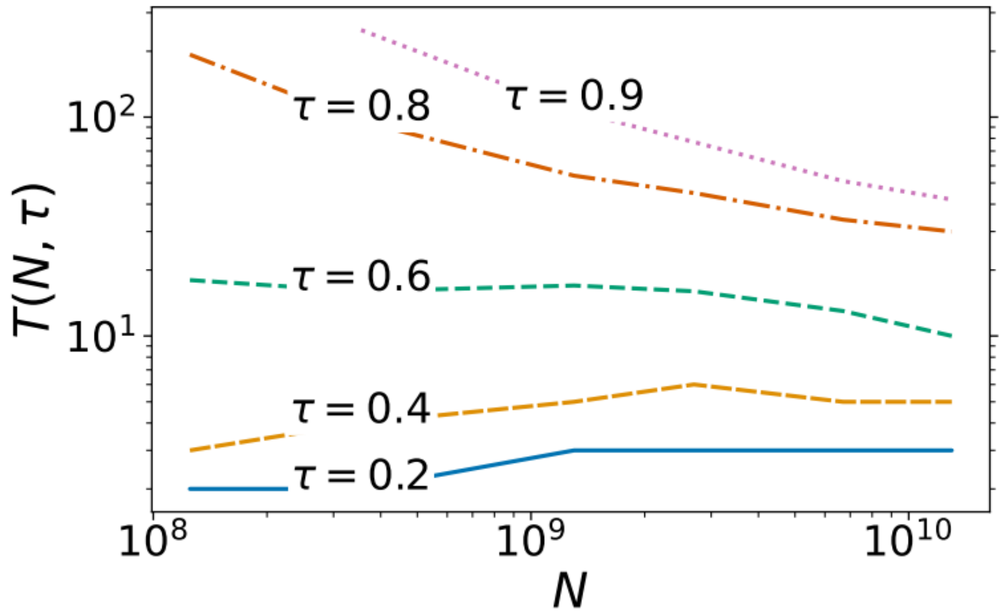


$T(N, \tau)$ the number of times a language model needs to see each training example before memorizing τ fraction of the training data

Larger Language Models Memorize Faster

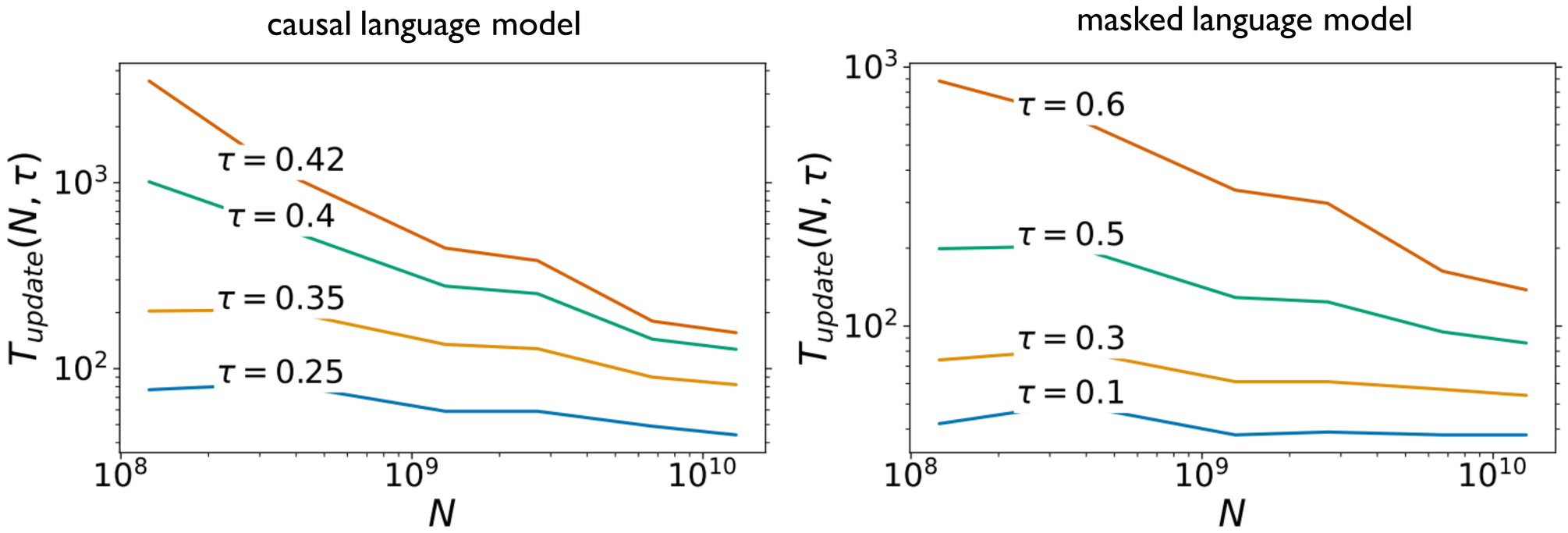
- masked language model + WIKITEXT103

larger models initially memorize training data slower, but reach high proportions of training data memorization faster



Larger Language Models Memorize Faster

- ROBERTA dataset



$T_{update}(N, \tau)$ the number of gradient descent updates a language model needs to perform before memorizing τ fraction of the data

larger models memorize faster, regardless of τ

Why Do Larger Models Memorize Faster?

- overfitting: the first epoch when the perplexity of the language model on a validation set increases

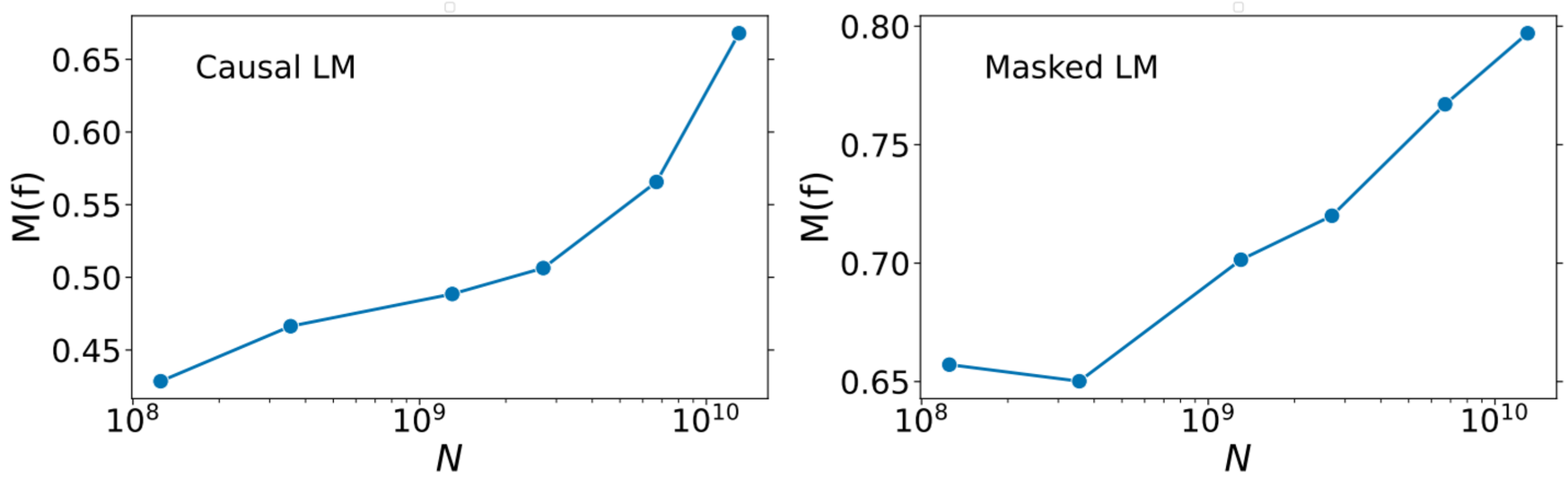
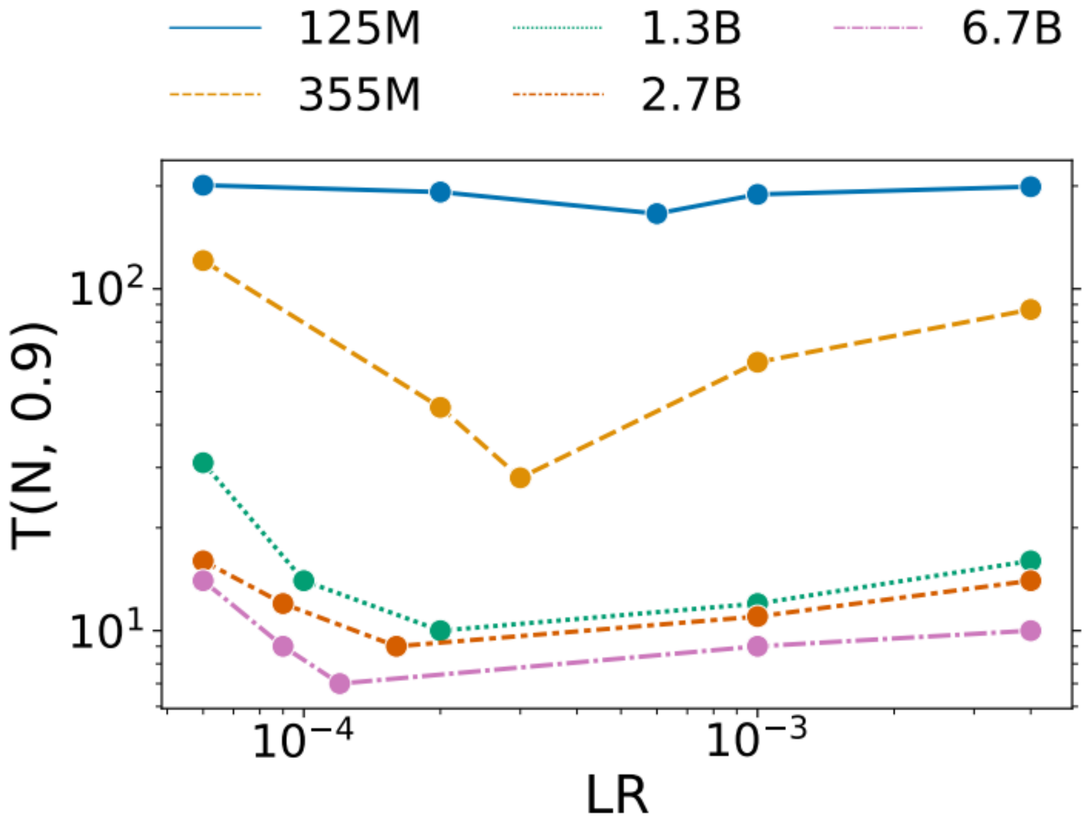


Figure 4: Proportion of training data memorized $M(f)$ before overfitting, as a function of model size N (plotted on a log scale). Results are for causal (left) and masked (right) language modeling on WIKITEXT103. Note that larger models memorize more before overfitting.

Why Do Larger Models Memorize Faster?

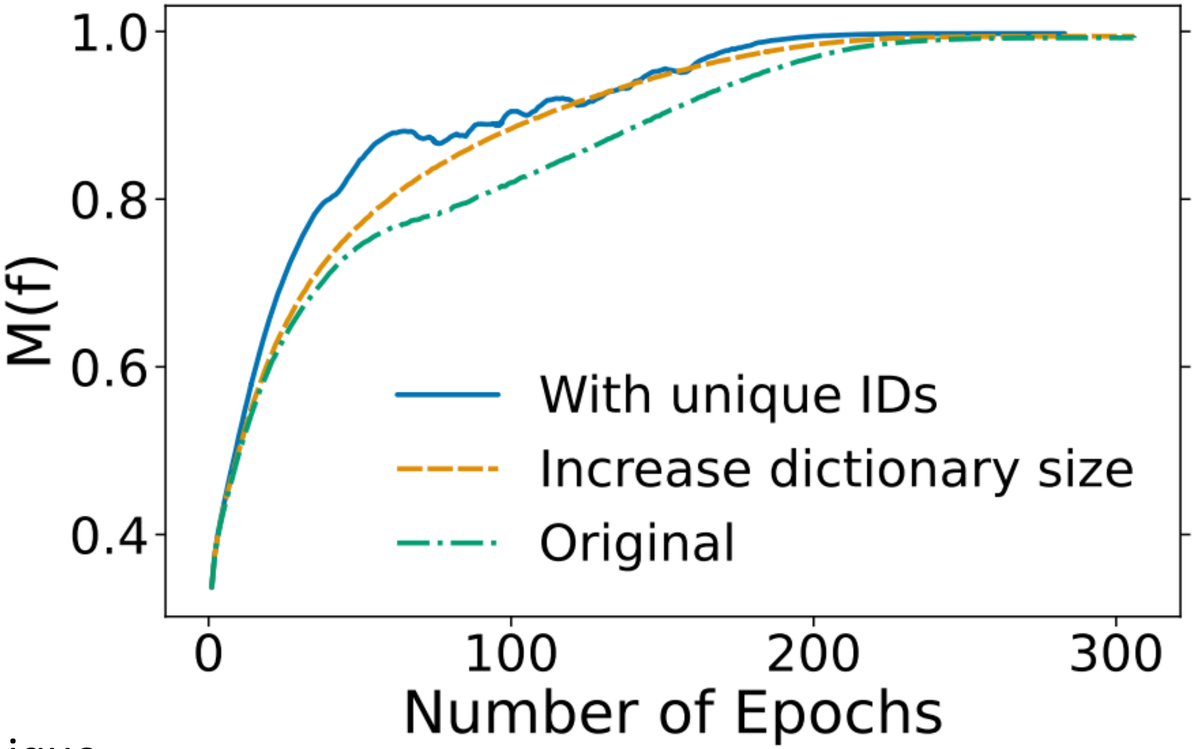
- larger models memorize faster for a fixed learning rate
- sensitivity to learning rate generally decreases as we increase the model size.



causal language modeling on WIKITEXT103

Why Do Larger Models Memorize Faster?

- increasing the dictionary size does improve the speed of memorization
- adding unique identifiers leads to faster memorization of training data

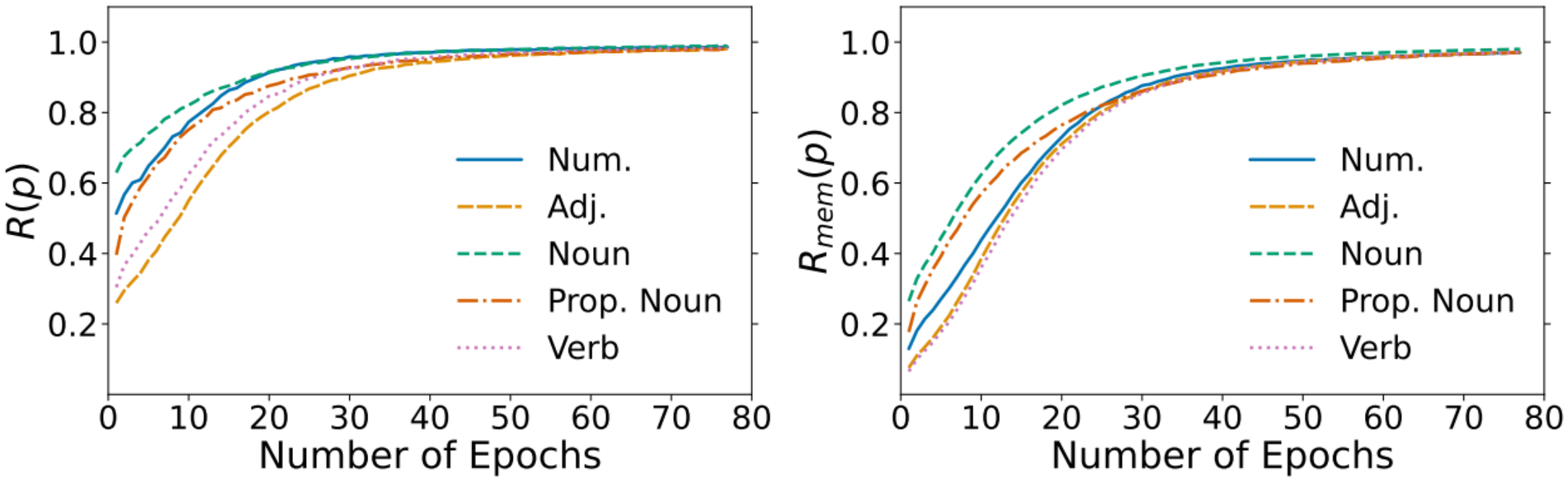


causal language modeling on WIKITEXT103

Regular text also contains strong proxies to unique identifiers in the form of numerals and proper nouns

Why Do Larger Models Memorize Faster?

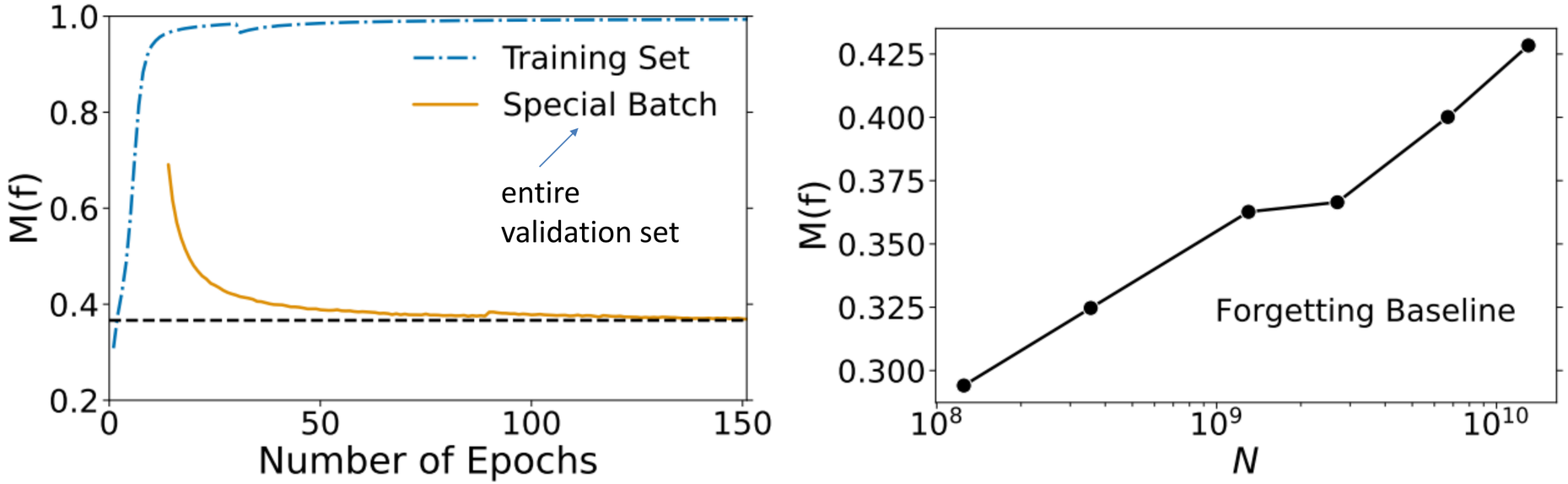
- some parts of speech are memorized faster -> nouns, proper nouns, and numerals
- loosely align with work studying child language acquisition



causal language modeling on WIKITEXT103

Forgetting Curves in Language Models

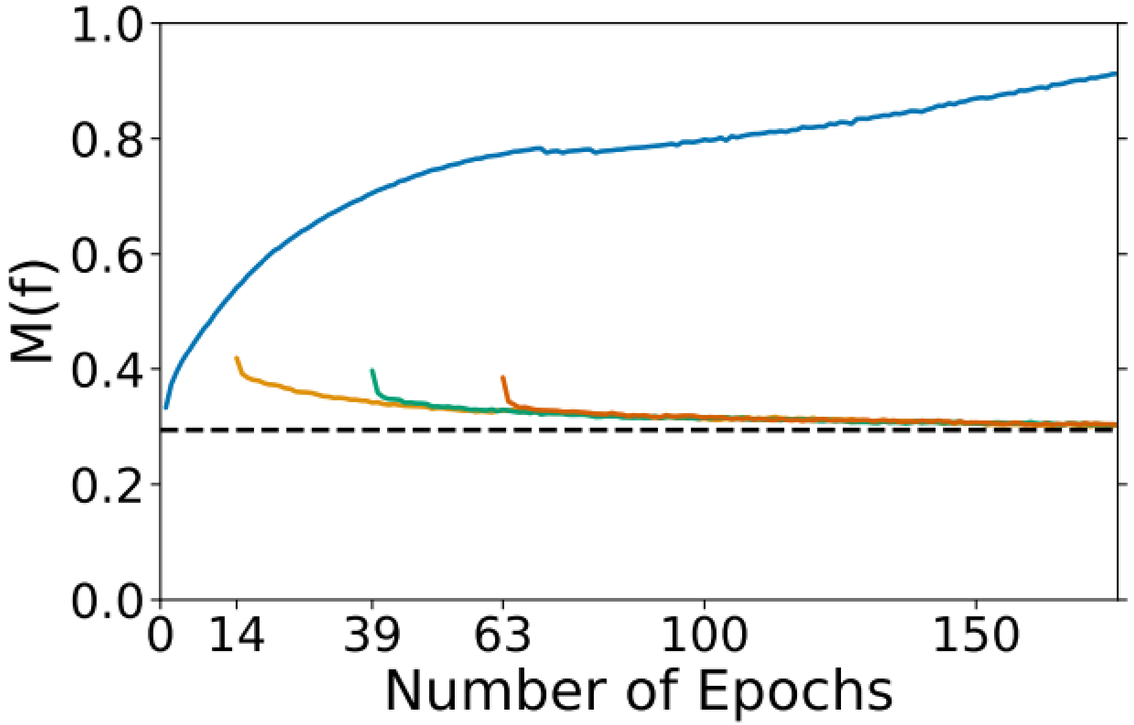
- forgetting baseline -> lowest memorization value on the special batch throughout training
- larger models forget less



causal language modeling on WIKITEXT103

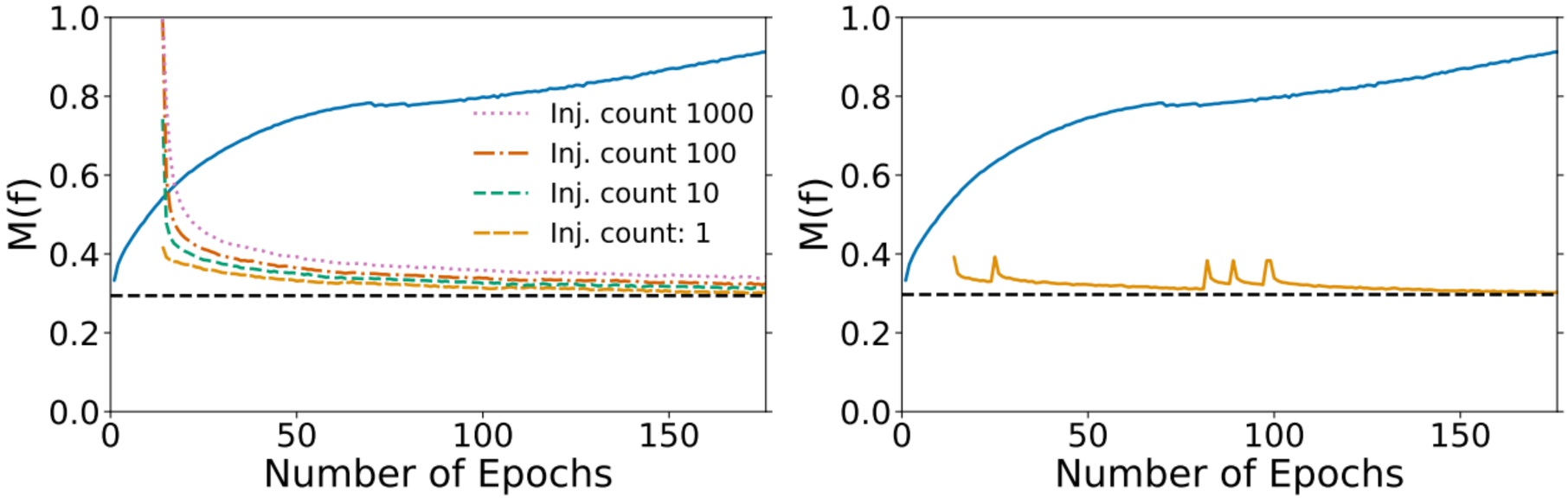
Forgetting Curves in Language Models

- the forgetting baseline is not sensitive to data batch order



Forgetting Curves in Language Models

- We show that repeated injection increases the forgetting baseline, whereas spaced repetition has minimal effect.



inject the special batch into the training set multiple times before continuing training

Conclusion

- larger models can memorize a larger portion of the data before over-fitting and forget less
- models memorize nouns and numbers first

Understanding Transformer Memorization Recall Through Idioms

Adi Haviv[τ] **Ido Cohen**[τ] **Jacob Gidron**[τ] **Roei Schuster**[ν] **Yoav Goldberg**[αβ] **Mor Geva**[α]

[τ]Tel Aviv University [ν]Vector Institute for AI [β]Bar-Ilan University [α]Allen Institute for AI

Motivation

- provide a definition and construct a dataset that allows probing memorization recall in LMs
- analyze memory recall behavior using constructed dataset

Methods

- Prior work
  - increased accuracy -> differ across models and training parameterization

- Definition (sufficient condition)
  - Single target, independent of context: We require that the target is the only correct continuation, regardless of the textual context where the prompt is placed
  - Irreducible prompt: The target is the single correct completion only if the entire prompt is given exactly. Changing or removing parts from the prompt would make the correct target non-unique

Methods

- Definition (sufficient condition)
  - Single target, independent of context
  - Irreducible prompt

idioms

“to get there fast, you can take this ____”

route
highway
road
train
plan
advice
…

“it’s raining cats and ____”

dogs

Methods

- The IDIOMEM Dataset
  - Datasets from MAGPIE, EPIE, LIDIOMS, Education First website(EF)
  - Split into (prompt, target)
  - Filter out those targets can be predicted by spurious correlations

Methods

1. idioms with < 4 words, multiple plausible continuations. “break a ___”

2. commonly predicted from the prompt’s subsequence (n-gram)

3. targets are semantically similar to tokens in the prompt. Glove similarity > 0.75

Prompt	Target	Pred.	Sim.	IDIOMEM
“make a mountain out of a”	molehill			✓
“think outside the”	box			✓
“there’s no such thing as a free”	lunch			✓
“go back to the drawing”	board	✓		
“boys will be”	boys		✓	
“take it or leave”	it	✓	✓	

Source	# of Idioms	Idiom Length (words)
MAGPIE	590	4.5 ± 0.9
LIDIOMS	149	5.1 ± 1.2
EF	97	5.6 ± 1.9
EPIE	76	4.4 ± 0.7
Total (unique)	814	4.7 ± 1.8

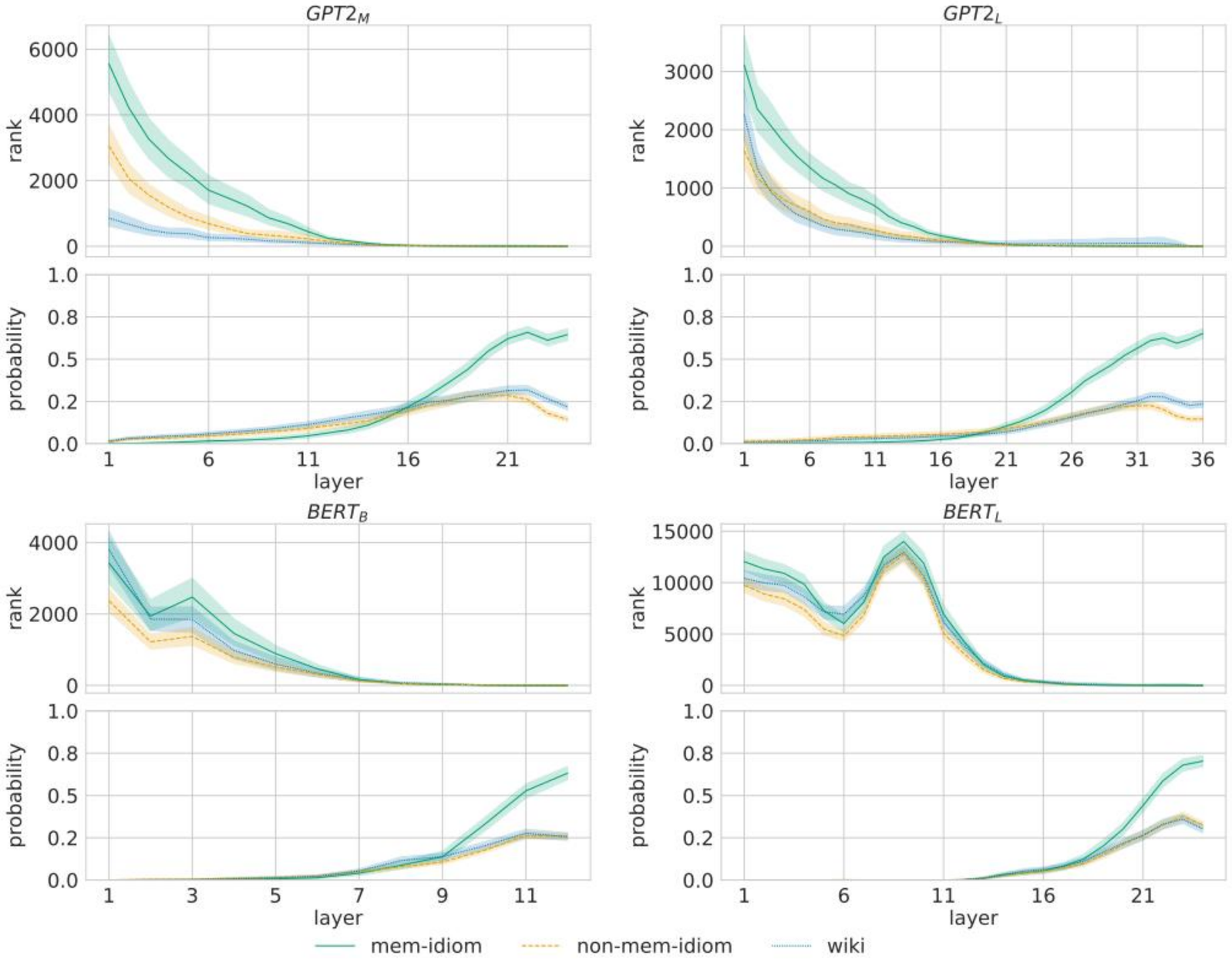
Probing Methodology

- Three subsets
  - “memorized” set
  - “non-memorized” set
  - natural language set

	GPT2_M		GPT2_L		BERT_B		BERT_L	
Memorized idioms (`mem-idiom`)	364	44.7%	392	48.2%	230	28.3%	305	37.5%
Non-memorized idioms (`non-mem-idiom`)	450	55.3%	422	51.8%	584	71.7%	509	62.5%

Table 3: Number of memorized idioms vs. non-memorized idioms from the IDIOMEM dataset for each model. An instance is considered a memorized example if the model correctly predicts the target.

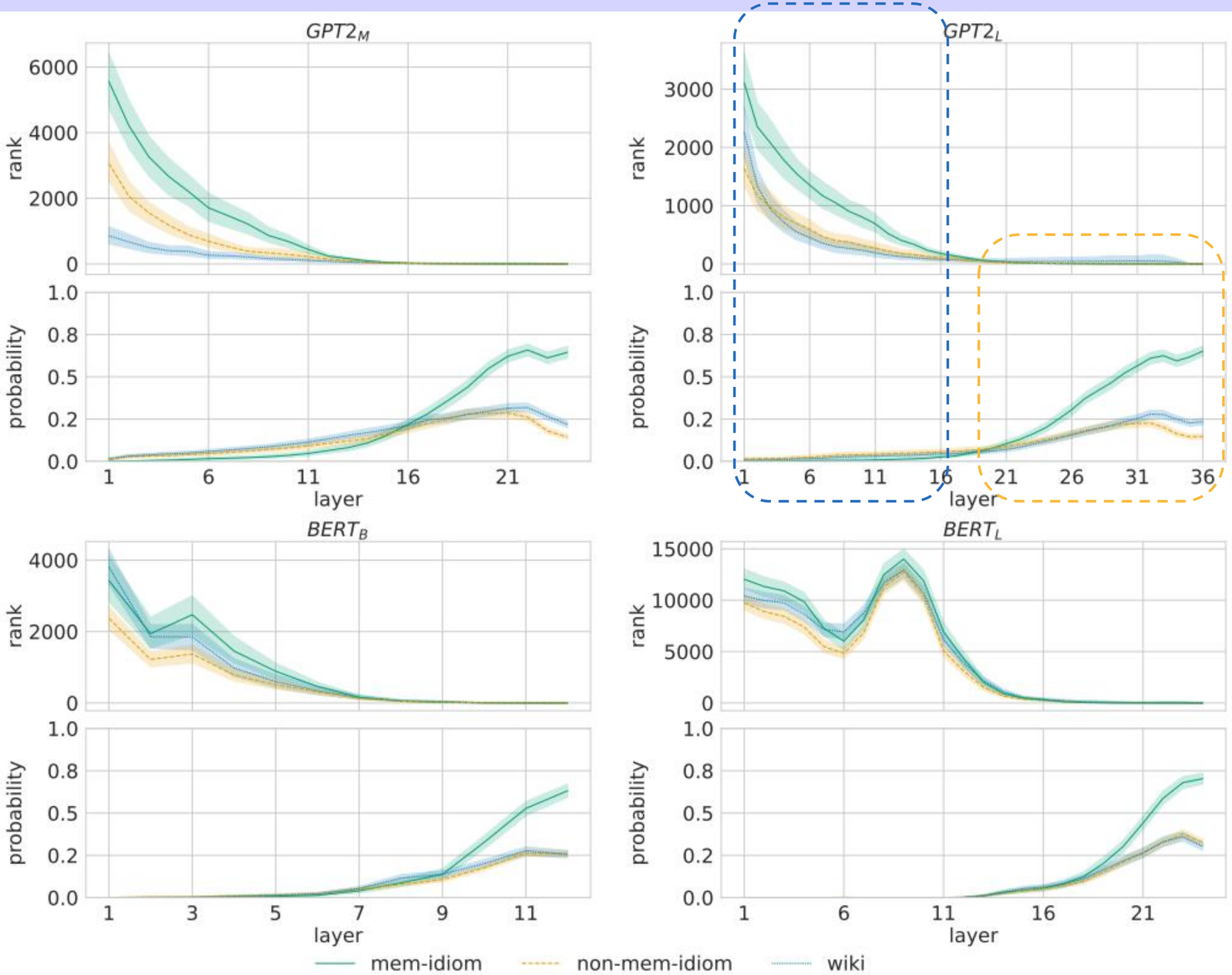
Probing Methodology



Probing Methodology

candidate promotion

non-memorized predictions are often promoted in early layers that detect local “shallow” patterns such as common bigrams



confidence boosting

memorized idioms have a single correct target

Probing Methodology

- Testing the Roles of Different Layers Through Intervention

$$\mathrm{FFN}^{\ell}(\mathbf{h}_i^{\ell}) = \sum_{j=1}^{d_m} f(\mathbf{h}_i^{\ell} \cdot \mathbf{k}_j^{\ell})\mathbf{v}_j^{\ell} = \sum_{j=1}^{d_m} m_j^{\ell}\mathbf{v}_j^{\ell},$$

- sample 100 random memorized idiom, for each range of up to 3 consecutive layers
  - sort the sub-updates $|m_i^{\ell}|||\mathbf{v}_i^{\ell}||$, set $m_i^{\ell} = 0$ for 10 sub-updates with the highest contribution
  - zero-out other sub-updates

Probing Methodology

- Testing the Roles of Different Layers Through Intervention

candidate promotion

confidence boosting

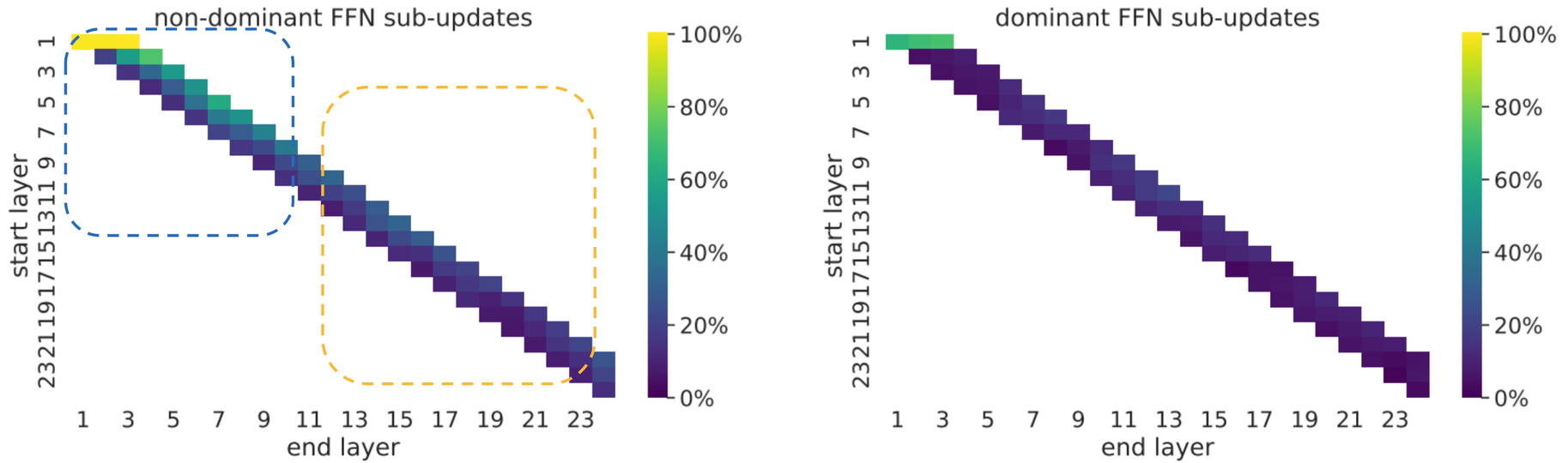


Figure 2: Intervention in non-dominant (left) and dominant (right) FFN sub-updates in $GPT2_M$. Each cell shows the percentage of memorized idioms for which the prediction was changed by zeroing-out the FFN sub-updates between the start and end layers.

Intervention in upper layers rarely changes the predicted token, and its effect is limited to reducing the model's confidence

Memorization of Factual Statements

- generalize beyond idioms to other types of memory recall
- LAMA-UHN (subset of LAMA)
  - only queries where the blank appears at the end
  - keep only queries with a single correct completion
  - 17,855 factual statements with 22 unique relations

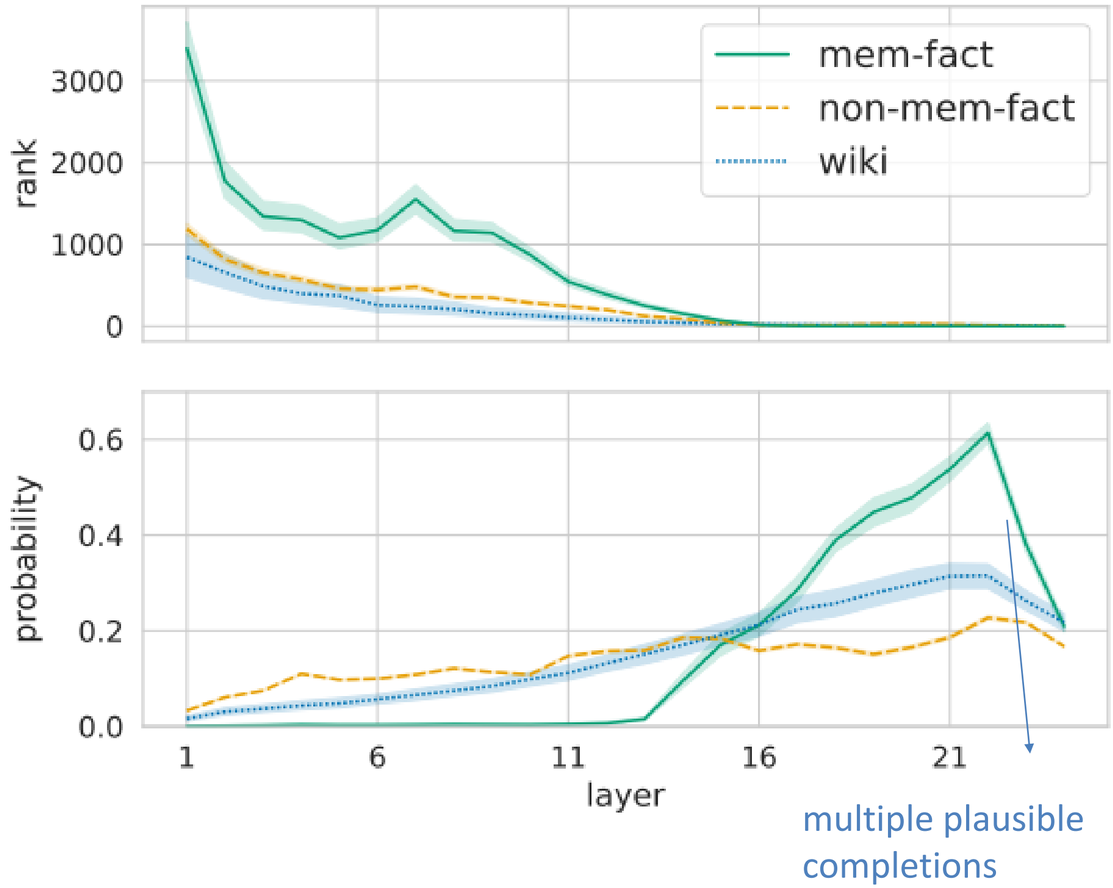


Figure 3: The predicted token's probability and rank across layers of GPT2_M, for memorized (`mem-fact`) and non-memorized (`non-mem-fact`) facts and short sequences from Wikipedia (`wiki`).

Conclusion

- memorized predictions are a two-step process
- memorized information is stored and retrieved in the early layers of the network