PreCog: Exploring the Relation between Memorization and Performance in Pre-trained Language Models

Leonardo Ranaldi (*,•), **Elena Sofia Ruzzetti**(*), **Fabio Massimo Zanzotto**(*)
(•) Idiap Research Institute, Martigny, Switzerland
(*) ART Group,
Department of Enterprise Engineering, University of Rome Tor Vergata, Rome, Italy
`[first name].[last name]@uniroma2.it`

Abstract

Large Language Models (LLMs) are impressive machines with the ability to memorize, possibly generalized learning examples. We present here a small, focused contribution to the analysis of the interplay between memorization and performance of BERT in downstream tasks. We propose *PreCog*, a measure for evaluating memorization from pre-training, and we analyze its correlation with the BERT's performance. Our experiments show that highly memorized examples are better classified, suggesting memorization is an essential key to success for BERT[1].

1 Introduction

Large Language Models (LLMs) (Brown et al., 2020; Touvron et al., 2023) are intriguing machines dominating the arena of NLP tasks with their ability to memorize generalizations of texts in synthetic neurons. After long pre-training on large amounts of unlabeled data, LLMs have been shown to learn effectively downstream tasks with limited labeled data (Howard and Ruder, 2018) and generalize in out-of-distribution examples (Hendrycks et al., 2020). Extensive studies have shown that these models tend to mimic traditional linguistic syntactic models (McCoy et al., 2019; Ranaldi and Pucci, 2023) and traditional NLP. Hence, a crucial issue is to clarify why PLTMs exploit pre-training better than traditional NLP modules exploit annotated corpora.

Understanding the learning process of LLMs may help in understanding their results in downstream tasks and in improving their linguistic representations in scenarios where they fail (Kumar et al., 2020). Indeed, unlike traditional general NLP modules in pipelines, LLMs need to be fine-tuned for the specific tasks (Devlin et al., 2019) and, eventually, domain-adapted on the specific language of the novel corpus (Jin et al., 2022). Moreover, as with many other machine learning models, fine-tuned PTLMs lose their ability to solve a task if subsequently fine-tuned to another task (Xu et al., 2020) although they apparently do not change their language models (Merchant et al., 2020). This phenomenon is known as *catastrophic forgetting* (Kirkpatrick et al., 2017) in machine learning. Then, it is still unclear how these models exploit pre-training and training examples.

LLMs, such as BERT (Devlin et al., 2019), have shown to have an impressive ability to memorize and possibly generalize learning examples. This ability has been largely investigated as it may be extremely harmful. In fact, these models may reveal sensitive information that has been acquired during pre-training. For example, memories of GPTs (Radford and Narasimhan, 2018) have been violated and produced phone numbers, and usernames (Carlini et al., 2021; Thakkar et al., 2021). However, this simple ability to memorize may play a crucial role in the performances of LLMs in downstream tasks (Ranaldi et al., 2022a; Uppaal et al., 2023).

This paper presents a small, focused contribution to the role of memorization in the performance of BERT in downstream tasks. We propose *PreCog*, a very simple measure of coverage that evaluates how much pre-training covers the information needed to model a given example or, better, if BERT has already partially seen the example - it *pre*-cognizes the example. The aim is to evaluate if PreCog *precognizes* which examples BERT adapted to a downstream task performs better inferences. We have extensively experimented with PreCog by using BERT over the GLUE tasks (Wang et al., 2018), and we observed the ability of PreCog to predict examples where a task-adapted BERT performs

[1]The code and is publicly available at: `https://github.com/ART-Group-it/PreCog`

Proceedings of Recent Advances in Natural Language Processing, pages 961–967
Varna, Sep 4–6, 2023
https://doi.org/10.26615/978-954-452-092-2_103

better. Besides being a predictive measure, PreCog showed that example memorization is a crucial part of the success of LLMs.

2 Related Work

The ability of linguistic neural models to memorize facts is out of doubt (Ranaldi et al., 2022a). This ability has been deeply explored as it is a problem for privacy issues. Indeed, LSTM language models remember facts so well that individual facts can be retrieved during inference (Carlini et al., 2019). These facts may reveal sensitive personal information such as names and addresses associated with people. Moreover, revitalizing the idea of sparse distributed memories (Kanerva, 1988), Petroni et al. (2019) hypothesized that Large Language Models might be used as clever and inexpensive ways to build up effortlessly knowledge bases. Even in other areas like image classification, it appears that large neural networks may memorize entire datasets as these networks achieve very low error rates over datasets with randomly generated target labels (Zhang et al., 2017). This also proves to be a problem for the de-biasing phenomenon (Ranaldi et al., 2023). Yet, it is still unclear to what extent this ability to memorize facts helps neural networks in downstream tasks.

A key research question is to understand how large pre-trained neural networks generalize over memorized examples. Pre-training seems to be a winning strategy to boost generalization. In fact, pre-trained models generalize better on out-of-distribution data and can detect such data better than non-pre-trained methods (Hendrycks et al., 2020; Ranaldi et al., 2022b). However, these models need a significant number of training instances to exploit this generalization ability in downstream tasks (Tänzer et al., 2022). Hence, since fine-tuning on specific datasets seems to be connected to *catastrophically forgetting* examples (Xu et al., 2020), generalization and memorization can be strictly correlated.

To explore the correlation between memorization and performance on downstream tasks, we propose a mechanism for analyzing sentence coverage. In particular, we investigate how many sentences are seen in the pre-training phase in transformer-based PLMs using perturbation masking methods. These methods allow us to observe the impact of pre-training on the performance of downstream tasks. This novel measure is needed as current measures for understanding coverage, such as "forgetting event" (Toneva et al., 2019) and counterfactual memorization (Zhang et al., 2021), mix performance, and actual memorization.

3 Method and Data

This section introduces PreCog, which is our measure to evaluate how much pre-training covers the information needed to model a given example (Sec. 3.1), two comparative measures $Lenght$ and $LexCov$ (Section 3.2), and the experimental setting (Section 3.3).

3.1 *PreCog*: a measure to evaluate pre-training coverage

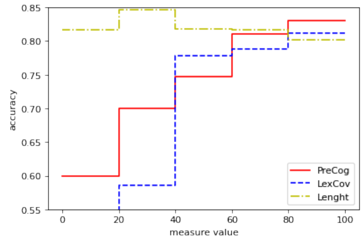
BERT (Devlin et al., 2019) is pre-trained on billions of text tokensby using Masked Language Modeling (MLM) as one of the two main learning tasks.Indeed, during pre-training, MLM randomly selects and masks 15% of all tokens in any given sequence. This 15% of tokens are either (a) replaced with the special token [MASK], (b) replaced by a random token, or (c) kept unchanged with a respective probability of 80%, 10%, and 10%. Then, BERT learns to predict the masked tokens. This task is learned till near the overfitting.Then, one of the main ability of BERT is unmasking masked tokens.

We aim to captureto which extent a sequence of tokens is covered by pre-training in Transformers such as BERT .For this reason, we build on the core capacity of BERT, that is, unmasking masked tokens. Hence, if BERT can predict masked tokens of a given sequence of tokens, it possibly has the knowledge to better deal with that sequence.Our intuition is that a measure built on unmasking masked tokens describes the "prior" knowledge of BERT over sequences.

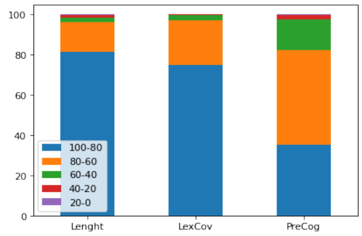
Given a sentence or text excerpt as a list of tokens $x = [x_1, ..., x_T]$, our function $PreCog(x)$ is defined as follows.Firstly, we mask one by one each token in x obtaining T different sequences $\hat{x}_i = [x_1, ..., x_{i-1}, [MASK], x_{i+1}.., x_T]$. Then, the measure is straightforwardly defined as:

$$PreCog_l(x) = \frac{\sum_{i=0}^{T} \delta(x_i \in BERT_{MLM}(\hat{x}_i))}{T} \quad (1)$$

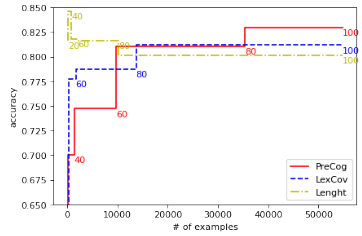
where $BERT_{MLM}(\hat{x}_i)$ is the set of the first 100 tokens predicted by BERT for the position i and $\delta(x_i \in X)$ is 1 if $x_i \in X$ and 0 otherwise.



(a) Accuracy $BERT_{FT}$ on bins of 20 points plotted vs. value of proposed measures.



(b) Percent of coverage of the dataset for intervals of values of the proposed measures.



(c) Accuracy of $BERT_{FT}$ bins of 20 points plotted vs. the coverage of the test set.

Figure 1: Accuracy plots of $BERT_{FT}$ for each GLUE task's weighted sum of accuracies.

PreCog is a very simple measure.Yet, it may reveal important facts about how BERT uses pre-training text in downstream tasks.A very important issue is to understand if PreCog correlates with the performance of BERT in these tasks.A positive and steady correlation will be an important hint for understanding the role of pre-training.

3.2 Alternative Coverage Measures

To comparatively evaluate $PreCog$, we use two measures: Length and LexCov. Length aims to correlate the accuracy of BERT to the length of samples and LexCov to the coverage of the dictionary of BERT. Then, the measures are defined as follows:

- $Length(x) = \frac{T-min_D}{max_D-min_D}$ where T is the length of x, min_D and max_D are the min and the max length of samples in a dataset D;
- $LexCov(x) = \frac{T-|OOV(x)|}{T}$ where $OOV(x)$ is the set of the out-of-vocabulary words of the example x with respect to BERT's vocabulary.

3.3 Experimental set-up

To experiment with a variety of tasks, we use the GLUE benchmark (Wang et al., 2018) containing tasks for: (1) natural language inference, that is, Multigenre NLI (MNLI) (Williams et al., 2018), Question NLI (QNLI) (Wang et al., 2018), Recognizing Textual Entailment (RTE) (Bentivogli et al., 2009), and Winograd NLI (WNLI) (Levesque et al., 2012); (2) semantic similarity, that is, the Microsoft Research Paraphrase Corpus (MRPC) (**?**), the Semantic Textual Similarity Benchmark (STS-B) (Cer et al., 2017), and Quora Question Pairs (QQP) (Sharma et al., 2019); sentiment classification - Stanford Sentiment Treebank (SST-2) (Socher et al., 2013); and corpus of linguistic acceptability (CoLA) (Warstadt et al., 2019). SST-2 and CoLA are single-sentence tasks.

We used two versions of BERT (Devlin et al., 2019): $BERT_{FT}$ with fine-tuning and $BERT_{DA}$ with domain-adaptation. These two are based on the pre-trained version of BERTforSequenceClassification (see (Wolf et al., 2020)). The fine-tuning procedure is that of traditional BERT. For each downstream task, we chose the Adam optimizer (Kingma and Ba, 2015) with a batch size of 16 and fine-tuned BERT for 4 epochs, following the original paper (Devlin et al., 2019). For hyperparameter tuning, the best learning rate is different for each task, and all original authors choose one between 1×10^{-5} and 5×10^{-5}.

We conduct our experiments on NVIDIA RTX A6000 GPUs with CUDA v11.3. We run the models from the Transformers library (Wolf et al., 2020) using PyTorch v1.12.0.

To study the correlation between the performance of BERT on the one side and one of the three measures - PreCog, Length, or LexCov - on the other side, we divided the sequences x in test sets in 5 bins according to the value of the measure, we plotted histograms of accuracies of BERT with respect to the three measures (Fig. 1), and we computed the Pearson's correlation of the measure with respect to the accuracies (Tab. 2).

4 Experimental Results and Discussion

Accuracies reported in Fig. 1a and Fig. 1c and used in Tab. 2 are the weighted sum of accuracies in each GLUE task. This guarantees that the 20-point bins have a sufficient set of samples to compute stable accuracies.

PreCog correlates with the accuracy of $BERT_{FT}$ better than Lenght and LexCov (see Fig. 1a and Tab. 2). Accuracies of PreCog in the different bins degrade more uniformly than the other two measures (red solid line in Fig. 1a). Moreover, the Pearson's correlation between PreCog values

Task	Global $BERT_{FT}$	Global $BERT_{DA}$	interval	Length # samples	Length $BERT_{FT}$	Length $BERT_{DA}$	LexCov # samples	LexCov $BERT_{FT}$	LexCov $BERT_{DA}$	*PreCog* # samples	*PreCog* $BERT_{FT}$	*PreCog* $BERT_{DA}$
COLA	0.920	0.935	(80,100]	499	0.906	0.918	857	0.926	0.940	577	0.951	**0.972**
			[0,80]	446	0.935	0.955	88	0.852	0.886	368	0.870	0.878
MNLI	0.716	0.721	(80,100]	7782	0.717	0.721	6512	0.739	0.745	3508	0.759	**0.770**
			[0,80]	1361	0.716	0.718	2631	0.660	0.660	5635	0.690	0.690
MRPC	0.806	0.861	(80,100]	59	0.780	0.831	924	0.818	0.877	376	0.867	**0.880**
			[0,80]	1590	0.806	0.861	725	0.789	0.839	1273	0.787	0.854
QNLI	0.808	0.829	(80,100]	3245	0.802	0.832	3123	0.809	0.831	1769	0.832	**0.846**
			[0,80]	1970	0.817	0.825	2092	0.807	0.827	3446	0.796	0.821
QQP	0.822	0.845	(80,100]	32728	0.820	0.845	28862	0.823	0.843	12810	0.840	**0.860**
			[0,80]	3990	0.834	0.842	7856	0.816	0.850	23908	0.812	0.837
RTE	0.646	0.653	(80,100]	146	0.671	0.678	155	0.716	**0.723**	46	0.652	0.674
			[0,80]	122	0.615	0.623	113	0.549	0.558	222	0.644	0.649
SST2	0.939	0.924	(80,100]	151	0.907	0.887	607	0.951	0.946	333	0.970	**0.970**
			[0,80]	655	0.947	0.933	199	0.905	0.859	473	0.918	0.892
WNLI	0.565	0.594	(80,100]	31	0.452	0.484	61	0.590	0.623	39	0.590	0.615
			[0,80]	38	**0.658**	0.684	8	0.375	0.375	30	0.533	0.567

Table 1: Accuracies on the GLUE tasks computed grouping datasets according to the values of three measures - PreCog, LexCov, and Lenght - for $BERT_{FT}$ and $BERT_{DA}$.

Measure	*Correlation*	*p-value*
Length	-0.5922	0.292
LexCov	0.9014	0.037
PreCog	0.9737	0.005

Table 2: Pearson's correlation between the measures and the accuracy bins of $BERT_{FT}$ for the combined GLUE tasks.

and the accuracies of $BERT_{FT}$ is 0.9737 with a p-value of 0.005 and it is higher than the ones of both LexCov, 0.9014 with a p-value of 0.037, and Length which is not correlated (see Tab. 2).

PreCog values better separate examples in testing sets. At first glance, LexCov may seem a better model to separate samples with high with respect to those with fewer accuracy expectations. Samples with a value of LexCov less than 40 have low accuracy (see Fig. 1a). However, samples having LexCov between 0 and 40 are rare (Fig. 1b). Better observations are derived by plotting accuracies over bins rescaled according to their coverage (Fig. 1c). Indeed, PreCog separates samples better than LexCov (red solid line vs. dashed blue line in Fig. 1c): samples from 18,000 to 55,000 fall in two bins for PreCog and in only one bin for LexCov. Hence, PreCog has better discriminative power than LexCov.

Results are substantially confirmed on task basis: PreCog is a better predictor of the accuracy on tasks and a better separator of classes of samples (see Tab. 1). Accuracies of $BERT_{FT}$ are generally higher for samples with PreCog in the interval $[80, 100]$ than for samples with the other two measures in the same interval. $LexCov$ has higher accuracy for samples in $[80, 100]$ only for RTE. Moreover, accuracies of samples in the interval $[80, 100]$ are always higher than those in the interval $[0, 80]$ for both PreCog and LexCov. Yet, PreCog partitions more evenly samples, and the differences in accuracies between intervals $[80, 100]$ and $[0, 80]$ are generally higher.

Moreover, domain adaptation is not changing the above findings. Accuracies for $BERT_{DA}$ are generally higher than those without domain adaptation for all the tasks except for SST2 and WNLI (Tab. 2). Moreover, focusing on PreCog, the overall increase in accuracies in CoLa, MNLI, and RTE derives from an increase in the samples of the interval $[80, 100]$. This fact suggests that $BERT_{DA}$ is gaining a better model for these samples.

As a final observation, BERT seems to behave better on sentences that have been, at least, partially seen during pre-training. Indeed, PreCog is a measure capturing how much the sentence is covered with the pre-training task Masked Language Model (MLM). Typically, BERT overfits MLM during pre-training. Then, PreCog is a measure telling whether sentences have already been partially seen. Instead, LexCov describes how many words of sentences are covered by BERT's vocabulary. Since there is a great difference in predicting accuracy on tasks between PreCog and LexCov, we can conclude that BERT behaves better when general knowledge of the target sentence is already acquired during pre-training.

5 Conclusion

Memorization of pre-training examples plays a very important role in the performance of BERT. Indeed, our PreCog, which measures how much memorized pre-training knowledge cover target examples, is highly correlated with BERT's performance in inference. PreCog can also be used to measure confidence for BERT-based decisions in downstream tasks.

As BERT success is partially due to simple memorization of examples and given the overwhelming presence of ChatGPT, one area of future research should be on better understanding the relation between actual training examples and inferences in order to give credit to knowledge producers.

Limitations

This paper presents a small, focused contribution towards the understanding of the relation between memorization and the performance of pre-trained Large Language Models (LLMs). However, we leave some issues unresolved for this more long-term goal. Indeed, we have explored our idea only for a specific LLM that is BERT with a specific pre-training task, that is, masked language model (MLM). Future analysis should explore whether our findings hold for other LLMs based on MLM. Moreover, we have not explored to what extent task examples are really covered by pre-training corpora used by LLMs. The correlation between PreCog and the actual training examples should be investigated. Finally, PreCog is not suitable for LLMs that are based on pre-training tasks that are not MLM. Then, other coverage measures should be defined in those cases.

Acknoledgements

This paper has been supported by "Knowledge At the Tip of Your fingers: Clinical Knowledge for Humanity" (KATY) project funded by the European Union's Horizon 2020 research and innovation program under grant agreement No. 101017453.

We would also like to thank all the anonymous reviewers that helped to strengthen this paper.

References

Luisa Bentivogli, Bernardo Magnini, Ido Dagan, Hoa Trang Dang, and Danilo Giampiccolo. 2009. The fifth PASCAL recognizing textual entailment challenge. In *Proceedings of the Second Text Analysis Conference, TAC 2009, Gaithersburg, Maryland, USA, November 16-17, 2009*. NIST.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Xiaodong Song. 2019. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *USENIX Security Symposium*.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, Alina Oprea, and Colin Raffel. 2021. Extracting training data from large language models.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2744–2751, Online. Association for Computational Linguistics.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Annual Meeting of the Association for Computational Linguistics*.

Xisen Jin, Dejiao Zhang, Henghui Zhu, Wei Xiao, Shang-Wen Li, Xiaokai Wei, Andrew Arnold, and Xiang Ren. 2022. Lifelong pretraining: Continually adapting language models to emerging corpora. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4764–4780, Seattle, United States. Association for Computational Linguistics.

Pentti Kanerva. 1988. *Sparse distributed memory*. MIT press.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.

Ankit Kumar, Piyush Makhija, and Anuj Gupta. 2020. Noisy text data: Achilles' heel of BERT. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 16–21, Online. Association for Computational Linguistics.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'12, page 552–561. AAAI Press.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Amil Merchant, Elahe Rahimtoroghi, Ellie Pavlick, and Ian Tenney. 2020. What happens to BERT embeddings during fine-tuning? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 33–44, Online. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *ArXiv*, abs/1909.01066.

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pretraining.

Leonardo Ranaldi, Aria Nourbakhsh, Arianna Patrizi, Elena Sofia Ruzzetti, Dario Onorati, Francesca Fallucchi, and Fabio Massimo Zanzotto. 2022a. The dark side of the language: Pre-trained transformers in the darknet.

Leonardo Ranaldi and Giulia Pucci. 2023. Knowing knowledge: Epistemological study of knowledge in transformers. *Applied Sciences*, 13(2).

Leonardo Ranaldi, Federico Ranaldi, Francesca Fallucchi, and Fabio Massimo Zanzotto. 2022b. Shedding light on the dark web: Authorship attribution in radical forums. *Information*, 13(9).

Leonardo Ranaldi, Elena Sofia Ruzzetti, Davide Venditti, Dario Onorati, and Fabio Massimo Zanzotto. 2023. A trip towards fairness: Bias and de-biasing in large language models.

Lakshay Sharma, Laura Graesser, Nikita Nangia, and Utku Evci. 2019. Natural language understanding with the quora question pairs dataset. *ArXiv*, abs/1907.01041.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Michael Tänzer, Sebastian Ruder, and Marek Rei. 2022. Memorisation versus generalisation in pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7564–7578, Dublin, Ireland. Association for Computational Linguistics.

Om Thakkar, Swaroop Ramaswamy, Rajiv Mathews, and Françoise Beaufays. 2021. Understanding unintended memorization in federated learning. In *Third Workshop on Privacy in Natural Language Processing (PrivateNLP 2021) at 2021 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2021)*.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. 2019. An empirical study of example forgetting during deep neural network learning. In *ICLR*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Rheeya Uppaal, Junjie Hu, and Yixuan Li. 2023. Is fine-tuning needed? pre-trained language models are near perfect for out-of-domain detection.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American*

Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Ying Xu, Xu Zhong, Antonio Jose Jimeno Yepes, and Jey Han Lau. 2020. Forget me not: Reducing catastrophic forgetting for domain adaptation in reading comprehension. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2017. Understanding deep learning requires rethinking generalization. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Chiyuan Zhang, Daphne Ippolito, Katherine Lee, Matthew Jagielski, Florian Tramèr, and Nicholas Carlini. 2021. Counterfactual memorization in neural language models. *ArXiv*, abs/2112.12938.