Improving Causal Interventions in Amnesic Probing with Mean Projection or LEACE

Alicja Dobrzeniecka
NASK -
National Research Institute
Warsaw, Poland
alicja.dobrzeniecka@nask.pl

Antske Fokkens and **Pia Sommerauer**
Computational Linguistics and Text Mining Lab
Dept. of Language, Literature and Communication
Vrije Universiteit, Amsterdam, The Netherlands
antske.fokkens,pia.sommerauer@vu.nl

Abstract

Amnesic probing is a technique used to examine the influence of specific linguistic information on the behaviour of a model. This involves identifying and removing the relevant information and then assessing whether the model's performance on the main task changes. If the removed information is relevant, the model's performance should decline. The difficulty with this approach lies in removing *only* the target information while leaving other information unchanged. It has been shown that Iterative Nullspace Projection (INLP), a widely used removal technique, introduces random modifications to representations when eliminating target information. We demonstrate that Mean Projection (MP) and LEACE, two proposed alternatives, remove information in a more targeted manner, thereby enhancing the potential for obtaining behavioural explanations through Amnesic Probing.

1 Introduction

A major paradigm for analyzing the linguistic information represented in language models is probing (Hupkes et al., 2018; Belinkov and Glass, 2019). At its core, the probing paradigm assumes that the information captured by the model can be learnt from its intermediate representations. While this paradigm enables linguistically motivated analyses, it has several limitations, as summarised by Belinkov (2022). A more recent framework of model analysis methods aims to establish *causal* connections between specific linguistic properties and their impact on the model (Feder et al., 2022). Amnesic Probing (Elazar et al., 2021) tests whether a model has specific linguistic information by first *identifying and removing* a property and then testing the impact of the removal on the behaviour of the model. Elazar et al. (2021) remove syntactic information and then test whether the model can still perform well on next-word prediction (a task

that requires knowledge about syntax). If the syntactic properties were successfully identified and removed, the performance on the task should drop.

A major factor in the Amnesic Probing framework is the *removal* of the target property. Elazar et al. (2021) use Iterative Nullspace Projection (Ravfogel et al., 2020, INLP), a method that has originally been proposed to debias models. INLP removes information by iteratively applying projections to the embedding space. Each projection removes more of the target information. A major risk of this method is removing information *beyond* the target information and thus damaging other aspects of the vector space (Haghighatkhah et al., 2022; Zhang et al., 2022). Multiple alternative methods that can remove information in a more targeted way have been proposed since (Ravfogel et al., 2022a,b; Haghighatkhah et al., 2022; Zhang et al., 2022; Belrose et al., 2024). Belrose et al. (2024) prove the efficiency of these approaches. They also show that their LEACE approach indeed removes pos-information more effectively and distorts the overall space less than INLP hinting at LEACE being superior for amnesic probing. However, the impact of using alternative removal methods such as LEACE or Mean Projection (MP) has, to our knowledge, not been investigated in full Amnesic Probing experiments. Additionally, despite the limitations of INLP, it remains a popular approach.[1] Therefore, it remains important to explain why alternative methods should be favoured.

This paper fills this gap and provides the following contributions. We provide an easy-to-follow step-by-step explanation of the Amnesic Probing framework, including additional checks and evaluations to the original experiments of Elazar et al.

---

[1] We found that INLP received 125 citations and Elazar et al 59 citations of papers not mentioning LEACE or MP since 2024 on Google Scholar, including from papers using INLP. This shows that not all researchers using this method are aware of superior alternatives.

(2021). We systematically compare INLP against MP method proposed by Haghighatkhah et al. (2022) and Zhang et al. (2022) and against LEACE on three control sets from Elazar et al. (2021). Our results demonstrate that, unlike INLP, both MP and LEACE improve the ability to uncover behavioural explanations via amnesic probing and successfully pass information control tests in two settings. Further analyses using our extended framework reveal that MP and LEACE more effectively remove the target property than INLP while causing less distortion to the remaining representation space. Although our original focus was on comparing INLP and MP — hence the more detailed treatment of MP — LEACE performs comparably to MP across all evaluations. We are also releasing a user-friendly framework.[2]

2 Amnesic Probing

The Amnesic Probing method, as proposed by Elazar et al. (2021), is based on the following idea. If we want to know, for example, whether a specific model uses syntactic information to predict the next word, we test the following: (1) whether the model contains syntactic information; (2) whether it performs worse at predicting the next word after the syntactic information is removed; (3) whether this drop in performance is greater than that observed in models with the same amount of random information removed; (4) whether explicitly adding syntactic information brings performance close to that of the original model. First, we provide an overview of the experimental setup and then describe the intuition behind the removal procedures used in this study.

2.1 General Experimental Set-Up

We motivate the general idea here, closely following Elazar et al. (2021), and provide a more detailed description of our experimental setup and results in Section 3. We first report reproduced INLP results that are not identical, but tell the same overall story as in Elazar et al. We hypothesize that Amnesic Probing will yield clearer results when MP or LEACE is used instead of INLP since it will remove target information more precisely. In particular, we expect that (1) MP and LEACE will lead to a larger performance drop than models modified by the same number of random projections (infor-

mation control). (2) We expect that the original performance on the task can be well restored by adding explicit gold labels to the modified vector space (selectivity control).

Elazar et al. (2021) study the impact of 6 linguistic properties: *syntactic dependency (dep)*, *f-pos*, *c-pos*, *named-entity (ner)*, *phrase start* and *phrase end*. Our experiments cover the properties *dep*, *f-pos* and *c-pos* from the Universal Dependencies (UD) English dataset (McDonald et al., 2013). Removing these properties had a clear impact on performance in the main task in Elazar et al. (2021) and the original data splits were available for reproduction. Data was not available for the other three tasks and INLP did not yield clear results for them. Including these additional experiments would likely confirm that neither INLP, MP, nor LEACE is effective in this context. Due to the lack of data and the limited expected insights, we chose not to include these experiments.

Elazar et al. (2021) use two metrics to test the impact of the information they removed. First, they evaluate performance on language modeling as a main task. Here they measure next-word prediction accuracy. Second, they calculate the Kullback-Leibler Divergence (D_{KL}), a useful measure to investigate the difference between two probability distributions.

2.2 INLP, MP and LEACE

The effectiveness of Amnesic Probing depends entirely on the quality of the method used to remove information. If the method is not precise enough, it becomes impossible to determine whether the drop is due to the removal of target information or loss of other information.

INLP Elazar et al. (2021) used INLP (Ravfogel et al., 2020) in their amnesic probing setup. INLP removes information as follows: a linear SVM model is trained to distinguish the classes of the target information. An orthogonal projection is applied along the decision boundary to *remove* the target information for each class. This process is repeated n times or until the classification performance remains below a threshold. This approach requires a relatively high number of projections to remove the target information: in Elazar et al. (2021) n is set to 20, meaning that for target information with 41 classes, we can end up with a maximum of 820 projections. This relatively high number of projections is seen as the main cause that

INLP removes other information beyond the target information (Haghighatkhah et al., 2022; Zhang et al., 2022).

MP Haghighatkhah et al. (2022); Zhang et al. (2022) propose an alternative method that does not require multiple iterative projections. Instead of training a model on the target property, this method simply takes the mean of the data points in each class to find the directions for their projections. This is a more efficient way to find an optimal projection compared to learning-based approaches, particularly when class distributions are unbalanced (Haghighatkhah et al., 2022). Unlike INLP, MP requires only one projection per class and the number of directions to be removed is stable and equal to the number of target classes. This can significantly reduce its detrimental effect on the rest of the representation. Zhang et al. (2022) confirm that MP removes syntactic information in a more targeted way than INLP. We create the set of directions in the same way as Zhang et al. (2022): $w_i = u_i - u_r$, where u_i is for the mean of a current class and u_r stands for the total mean of all the remaining classes excluding u_i.

LEACE The LEACE method, proposed in Belrose et al. (2024), removes target information in a more precise manner, similarly to MP. Both LEACE and MP perform linear projection-based concept erasure using label-driven subspaces. However, LEACE is more theoretically grounded and proves the removal of all linear information about the concept. Both methods have their pros and cons: LEACE provides more theoretical guarantees, while MP is heuristic. At the same time, LEACE is more computationally intensive, whereas MP is simple and straightforward to apply.

3   Experiments and Results

We walk through the amnesic probing procedure step-by-step and present the results of our comparison of INLP, MP and LEACE. Elazar et al. (2021) conducted experiments on two variants of the BERT model (masked and unmasked) (Devlin et al., 2019). Both led to the same insights. We present the results for unmasked BERT here, and for masked BERT in Appendix C.

Step 1: Can we identify and remove target information? Following Elazar et al. (2021), we first verify whether the target information is present in the original model through a regular probing procedure. We test whether a linear classifier can learn the target information from the vanilla model (i.e. in its original state). We then apply INLP as well as MP and LEACE to create alternative models where target information is removed. We add an additional test compared to Elazar et al. (2021) and check whether probing indeed fails on the target property after applying INLP and MP. The results are presented in Table 1. INLP, MP and LEACE all lead to a large drop in accuracy for probing the target information. For *dep* and *f-pos* the model performs below the majority class after applying INLP and slightly above it when MP or LEACE is applied. For *c-pos*, MP and LEACE bring the model down to performance below majority class and INLP remains slightly above. MP and LEACE yield highly comparable performance: MP does slightly better for *f-pos*, LEACE for *c-pos*.

We add two comparisons to investigate the impact of each method on the overall model: (1) the change in matrix rank (Strang, 2009) and (2) cosine similarity between layers before and after modification. Both tests show that INLP has a much larger impact on the model than MP and LEACE. A detailed description of these metrics and the results can be found in Appendix E.

Step 2: What is the impact on model behaviour? We verify whether removing the target information has an impact on the main task. We first compare the vanilla model's performance with the modified model's performance to see if removing the information leads to a decrease. If this is not the case, the information we removed did not provide support to carry out the task in the original model.

	dep	*f-pos*	*c-pos*
N. classes	41	45	12
Majority class %	11.44	13.22	31.76
Vanilla	76.77	90.04	92.44
Vanilla (LEACE)	74.65	89.03	92.01
Amnesic INLP	**9.73**	**10.79**	33.91
Amnesic MP	14.33	16.66	**24.06**
Amnesic LEACE	14.33	16.28	25.87

Table 1: Results for probing for non-masked BERT encoding. Values in bold indicate the largest change among the same properties. Probing on Vanilla was repeated when adding results for LEACE.

However, a drop in performance does not automatically mean that the target information helped. The drop can also be caused by other aspects of the modification. We compare the effect of removing target information with the impact of removing a comparable amount of random information (information control). If the target information is relevant, the removal of the target information should have a *higher negative impact* than the removal of random information. The Amnesic Probing framework contains two baselines that apply random modifications to the embedding space: For a Random baseline, we apply a projection with randomly chosen directions to the embedding space. The number of random directions is equal to the number of directions removed by the method we are comparing to. For INLP we apply 779 projections for *dep*, 765 for *f-pos* and 240 for *c-pos*. For MP, this amounts to 41, 45 and 12 projections and for LEACE to 41, 45 and, 12 respectively.[3]

Table 2 provides the results on language model accuracy and calculating D_{KL} for all models and their baselines. Applying INLP leads to a much larger drop in accuracy for the main task and a much higher D_{KL} compared to MP and LEACE. When INLP is used to remove dependencies or f-pos, more damage is done by applying the same amount of random projections in our experiments. This check shows that we cannot establish a causal link between the intervention of removing information with INLP and model behaviour on the main task. When MP or LEACE is used, we can establish this link; here, we can consistently observe a higher impact of the target information removal by MP compared to random removal.

Table 2 also includes the original results reported by Elazar et al. (2021). Results from INLP are influenced by random factors and in our reproduced study more projections were needed (see Limitations). Elazar et al. (2021) obtained meaningful for *f-pos*, next to *c-pos*. However, the results for MP and LEACE still appear to be much more precise than those reported by Elazar et al. (2021). Although the same number of random projections as in the original INLP study led to a significant drop in results compared to the vanilla model, the corresponding amount of random projections applied by

		dep	*f-pos*	*c-pos*
Acc	Vanilla	94.12	94.12	94.12
	Amn. INLP (repr)	8.74	14.6	**72.95**
	Rand. INLP (repr)	4.67	4.86	90.21
	Amn. MP	**79.49**	**65.39**	**86.87**
	Rand. MP	93.98	93.85	94.05
	Amn. LEACE	**86.82**	**70.96**	**90.18**
	Rand. LEACE	93.71	94.01	93.95
	Amn. INLP (Elazar)	**7.05**	**12.31**	**61.92**
	Rand. INLP (Elazar)	12.31	56.47	89.65
D_{kl}	Rand. INLP (repr)	8.98	8.9	0.47
	Rand. MP	0.28	0.02	0.0
	Rand. LEACE	0.02	0.02	0.00
	Amn. INLP (repr)	7.97	6.08	**1.99**
	Amn. MP	**0.89**	**1.63**	**0.49**
	Amn. LEACE	**0.34**	**1.06**	**0.19**
	Rand. INLP (Elazar)	8.11	4.61	0.36
	Amn. INLP (Elazar)	**8.53**	**7.63**	**3.21**

Table 2: LM task for non-masked BERT encoding comparing results of Amnesic (Amn) probing methods and the impact of the same number of random projection (Rand). Bold values reflect scenarios where the amnesic value had a bigger impact on accuracy (Acc) or led to a lower change in general distribution of tokens (D_{kl}) compared to random projections. We include the results reported by Elazar et al. (2021) (Elazar) for comparison to our reproduced results (repr).

MP or LEACE hardly affects the results.

Table 3 provides the results of an analysis testing the impact of distortions for specific cpos categories, in line with the analysis carried out by Elazar et al. (2021). It presents the accuracy of the original model, and the drop in accuracy after applying INLP, MP or LEACE for each pos-category. We observe that all projection methods mainly impact function words (conjunction, determiners and pronouns). This difference is even more pronounced when using MP compared to INLP. MP and LEACE have less impact on nouns than INLP and no impact (MP) or an improvement (LEACE) on accuracy of predicting verbs and adpositions, whereas INLP does impact these negatively. MP and LEACE do not appear to have an impact on predicting particles (which are also function words), INLP does impact their prediction.

Step 3: Can the performance be restored by putting the target information back into the vec-

---

[3]We also apply a Dropout baseline, where we randomly select columns in the embedding space and replace them with $0s$. The number of columns replaced is equal to the number of directions removed by the method. Dropout yields the same results as the random baseline, so we omitted it to save space.

Cat	Vanilla	δ INLP	δ MP	δ LEACE
ADJ	16.67	0.00	-4.17	0.00
NOUN	53.85	16.24	5.13	-0.85
ADP	56.52	21.74	0.00	-8.70
DET	88.24	79.41	67.65	67.65
CONJ	80.00	80.00	80.00	60.00
VERB	42.31	9.62	0.00	0.00
ADV	54.55	18.18	0.00	18.18
.	74.47	31.91	2.13	4.26
PRT	100.00	44.44	0.00	0.00
-	100.00	0.00	0.00	0.00
PRON	100.00	50.00	50.00	50.00
NUM	22.22	-11.11	0.00	-11.11

Table 3: Analysis of the impact of the distortion per c-pos category. δ INLP/MP/LEACE show the drop in results after projections. Negative values represent increased results.

tor space? We carry out the selectivity control test and see whether the performance on the main task can be restored by adding explicit gold labels of the target information to the modified space. If the target information contributed to the original result and the loss in performance was mostly due to this information now being absent, performance should be largely restored. In contrast, if the drop in performance was caused by other information that was removed, these gold labels will not be able to recover for the effect. We present the results in Table 4.[4] When adding gold labels, results for INLP, MP and LEACE all improve showing that the target information was responsible for the drop in performance at least to some extent. As expected, the results approximate the original result much better when MP or LEACE is used, coming close to the original result again. In INLP, we see that loss of other information also played a role. This confirms that MP and LEACE removed the target information more precisely and the results obtained while using MP provide a more reliable insight into the extent to which the target information

---

[4]Note that the modification needed to add gold syntactic information requires us to remove the sequential container. This explains the, sometimes substantial, drop in results of the amnesic models in Table 4 compared to Table 2.

supported the task.

	dep	*f-pos*	*c-pos*
Amnesic INLP	5.92	4.95	0.49
Amnesic MP	0.24	1.22	0.19
Amnesic LEACE	0.11	0.55	1.17
Gold labels INLP	87.06	92.23	96.56
Gold labels MP	96.75	96.81	96.8
Gold labels LEACE	96.58	96.83	96.79
Accuracy INLP Δ	81.14	87.28	96.07
Accuracy MP Δ	95.51	95.59	**96.61**
Accuracy LEACE Δ	**96.47**	**96.28**	95.62

Table 4: Selectivity Accuracy Δ show the highest change between embeddings without the information and embeddings with returned gold label information.

4 Conclusion

We presented a systematic comparison of three different information removal methods for amnesic probing. We hypothesized that the potential of amnesic probing as a model analysis method increases when using a more precise removal method, specifically MP or LEACE instead of INLP. Our results confirm the hypothesis; MP and LEACE outperform INLP on the information and selectivity control tests. In particular, in two out of three cases, random projections had a bigger impact on the main task than the model modified by INLP, which did not happen when MP or LEACE was used (information control). In all three cases, MP or LEACE-modified models recover performance better than INLP when explicit gold information about the target property is added (selectivity control). Information and selectivity control provide evidence that the drop in performance on the main task is indeed caused by the target information missing rather than by other artifacts of the modification. As such, our results show that MP and LEACE are more precise methods for removing information, which directly increases the potential of Amnesic Probing. MP and LEACE yield highly comparable results and we provide easy-to-use code for both approaches. We recommend to first use MP since this is more efficient and adding experiments with LEACE when checks and balances on the MP results call for it.

5 Limitations

Sensitivity to outliers MP calculates the mean target property vector for each class and uses the difference between these means to define the projection directions. As such, it addressed the problem of unbalanced classes (which is a problem for INLP). MP is, however, sensitive to outliers, as the mean of a class can be affected significantly by extreme values, shifting the projection direction in a way that does not accurately reflect the central tendency of the class distribution. If there are a small number of outliers in a class, they can disproportionately influence the calculated mean, resulting in projection directions that are misaligned with the true data distribution. Because the projection direction is determined by class means, an outlier can cause MP to remove information that is not representative of the class structure. This can result in incomplete removal of the target attribute or unnecessary distortion of the embedding space. Therefore, the control tests of the amnesic probing framework (conducted in this paper) should be standardly included in any Amnesic Probing experiment.

If control tests do point to a problem with outliers, this limitation is manageable and can potentially be addressed by outlier detection methods and does not outweigh the usefulness of MP. For example, we could identify and remove outliers before computing the class means, which could improve the robustness of MP.

Different Results in Reproduction Our reproduced results differ from those reported in Elazar et al. (2021). The differences in the results for dep and f-pos are largely due to random factors in the INLP method (initialization of the weights, different seeds). These random factors determine the number of iterations INLP needs to remove a property. We observe that INLP needed a higher number of iterations in our experiments than in the original experiments. The results we reproduced for *cpos* and *dep* are relatively close to the original results: the *cpos* results tell the same story. Looking at it from the three dimensions of reproduction introduced by Cohen et al. (2018): the numbers differ (dimension 1), but the overall findings (dimension 2) and conclusions (dimension 3) are the same for these two properties. In the case of *dep*, Elazar et al. (2021) report around 12% accuracy with random projections and 7% for INLP amnesic, whereas we report around 5% for random and 9% for INLP am-

nesic: even though our experiments show a bigger loss for random compared to amnesic, the drop is extremely large leading to very poorly performing models in all cases. We therefore also consider these results reproduced according to the second and third dimension of Cohen et al. (2018).

The reproduction difference is biggest for *fpos*. Here, random projections still perform at 56.47 for Elazar et al. (2021) with INLP amnesic dropping to 12.3. In our case, *fpos* drops to 4.8, which is more than the drop caused by INLP amnesic, which drops to 14.6%. This means that while Elazar et al. (2021) show an impact of *fpos* on language modeling, we do not reproduce this finding. We have added the original results from Elazar et al. (2021) to the results in Table 2. Note that MP (and LEACE) also reveal a more precise measurement of the impact of *fpos* compared to the original results in Elazar et al. (2021). We can observe that model performance barely changes when applying as many random projections as projections are required in MP/LEACE.

General Limited Scope and Expanded Motivation for Focusing on the 3 Tasks It is important to consider that the experiments carried out in this paper included only a limited set of linguistic properties and models. It is possible that the method behaves differently on other properties or models. Note, however, that including the three other control tasks from the original paper by Elazar et al. (2021) could not contradict the claims in this paper. INLP does not work for these tasks. Most likely, MP and LEACE do not work (well) either, but they could not do significantly worse. In other words, INLP cannot outperform MP or LEACE here. The chances of MP or LEACE doing better than INLP are very small, therefore we highlight that it is unlikely that adding these experiments would lead to new insights. We still considered adding these three tasks for the sake of completeness, but the data was not available and the authors of the original paper could not share it with us. This made it impossible to reproduce their original results. The combination of the limited insights we could gain from it and the problems around the data led to the decision to not include these experiments in our paper.

Runs The results reported in the paper are based on a single run. We did test whether they are stable across multiple runs, notably for INLP and check whether they were influenced by batch size. We

got similar results in all runs. Due to computational constraints, we were unable to consistently carry out the same number of runs for each experiment. Since our results do not depend on minor differences (i.e. they would not tell a different story based on multiple runs), we decided to report results based on one run.

Acknowledgements

References

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72.

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. 2024. Leace: Perfect linear concept erasure in closed form. *Advances in Neural Information Processing Systems*, 36.

Adi Ben-Israel. 2015. Projectors on intersection of subspaces. *Contemporary Mathematics*, 636:41–50.

K Bretonnel Cohen, Jingbo Xia, Pierre Zweigenbaum, Tiffany J Callahan, Orin Hargraves, Foster Goss, Nancy Ide, Aurélie Névéol, Cyril Grouin, and Lawrence E Hunter. 2018. Three dimensions of reproducibility in natural language processing. In *LREC international conference on language resources & evaluation*, volume 2018, page 156. NIH Public Access.

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.

Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. 2021. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175.

Amir Feder, Katherine A Keith, Emaad Manzoor, Reid Pryzant, Dhanya Sridhar, Zach Wood-Doughty, Jacob Eisenstein, Justin Grimmer, Roi Reichart, Margaret E Roberts, et al. 2022. Causal inference in natural language processing: Estimation, prediction, interpretation and beyond. *Transactions of the Association for Computational Linguistics*, 10:1138–1158.

Pantea Haghighatkhah, Antske Fokkens, Pia Sommerauer, Bettina Speckmann, and Kevin Verbeek. 2022. Better hit the nail on the head than beat around the bush: Removing protected attributes with a single projection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8395–8416.

Dieuwke Hupkes, Sara Veldhoen, and Willem Zuidema. 2018. Visualisation and'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *Journal of Artificial Intelligence Research*, 61:907–926.

Alfirna Rizqi Lahitani, Adhistya Erna Permanasari, and Noor Akhmad Setiawan. 2016. Cosine similarity to determine similarity measure: Study case in online essay assessment. In *2016 4th International Conference on Cyber and IT Service Management*, pages 1–6.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, et al. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256, Online. Association for Computational Linguistics.

Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan D Cotterell. 2022a. Linear adversarial concept erasure. In *International Conference on Machine Learning*, pages 18400–18421. PMLR.

Shauli Ravfogel, Francisco Vargas, Yoav Goldberg, and Ryan Cotterell. 2022b. Adversarial concept erasure in kernel space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6034–6055.

Gilbert Strang. 2009. *Introduction to Linear Algebra*, fourth edition. Wellesley-Cambridge Press, Wellesley, MA.

Xiaohan Zhang, Shaonan Wang, Nan Lin, Jiajun Zhang, and Chengqing Zong. 2022. Probing word syntactic representations in the brain by a feature elimination method. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

A INLP and MP Differences

Both MP and INLP reply on the concept of projection from linear algebra. Projection involves finding the closest vector for a given subspace. Geometrically, projection can be thought of as the shadow of the target vector on a subspace. The steps behind removing the target properties from the embedding using projection methods are as follows:

1. Find the direction for the projection onto the subspace

    (a) use weights from a linear classifier (INLP)

    (b) use the difference between mean class vector and the mean of the remaining vectors for all classes (MP)

2. Get the orthogonal projection matrix over the rowspace, which is equivalent to obtaining a nullspace (Ravfogel et al., 2020)

    (a) Use the formula from Adi Ben–Israel (Ben-Israel, 2015) to get orthogonal projection to the intersection of nullspaces while running multiple iterations (INLP)

    (b) Use the orthogonal projection over the rowspace once (MP)

3. Get a projection onto the orthogonal complement

Point 1 refers to the fact that a direction or set of directions is needed to perform a projection. The directions define the subspace on which the projection is performed. The goal of both methods is to find the directions that most accurately capture the information about the target property classes. Point 2 talks about getting an orthogonal projection using the directions defined in the previous step. Such a projection P, when applied to the embedding, should ensure that the classes of the target property are no longer linearly separable. Point 3 refers to the last step, where subtracting the intersection of the nullspaces from the identity matrix gives the projection to the union of the rowspaces of each direction (Ravfogel et al., 2020).

Figure 1 provides a visualisation of the step of removing target information.

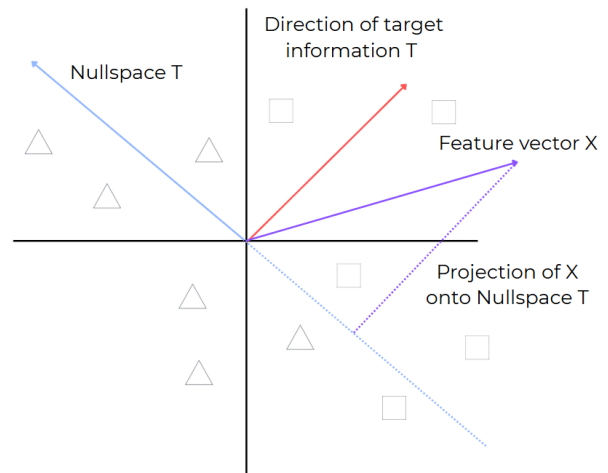An overview of the comparison of the two methods is shown in Table 5.

Figure 1: A simplified visualisation of the removal of target information using linear projection.

Why MP removes information in a more targeted way compared to INLP Amnesic probing aims to determine whether a model actually uses a particular property by removing it and observing the change in performance. If the embedding space is significantly altered beyond simply removing the target property, the results can be misleading - suggesting that the model relied on the property, when in fact the performance drop could be due to broader distortions introduced by the projection method. While both MP and INLP aim to remove certain properties from embedding space, MP is a more targeted and precise approach. MP applies a single projection, whereas INLP requires multiple iterations. This fundamental difference results in several advantages for MP in terms of stability, efficiency and reduced damage to the general embedding space.

INLP iteratively identifies and removes directions in the embedding space that encode the target property. However, after several iterations, the projections become increasingly random and less targeted, leading to unintended alterations in the embedding space (Haghighatkhah et al., 2022). This phenomenon results in greater distortion of non-target information, making it difficult to isolate the effect of the target property removal. Belrose et al. (2024) also highlight this risk of "collateral damage" in INLP, noting that as the number of projections increases, the embedding space undergoes cumulative distortions that are not necessarily related to the removal of target information.

Studies comparing INLP and MP show that MP alters the embedding space less than INLP while

goal	similar class distributions (MP)	
	linear guarding (INLP)	
multiple iterations	no (MP)	
	yes (INLP)	
directions source	means of the classes (MP)	
	classifier's weights (INLP)	
no. directions removed	equal to the no. classes (MP)	
	minimum the no. classes (INLP)	
orthogonal projection	yes (MP)	
	yes (INLP)	

Table 5: Similarities and differences between MP and INLP.

achieving comparable target property removal. For example, KL divergence measurements indicate that MP produces a more stable token distribution than INLP. In addition, probing accuracy tests confirm that MP successfully removes the target property while better preserving the original space. INLP also typically requires 10-15 iterations to make the target information linearly inseparable. MP achieves the same level of separability with a single projection, which means less computation, less distortion, and a more precise modification of space. Because of these advantages, MP is recommended over INLP for applications that require precise information removal.

Figures 2 (before) and 3 (after) provide an illustration of how target information is identified and removed through projection.
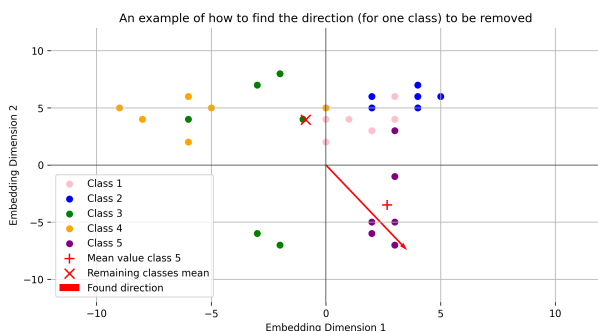

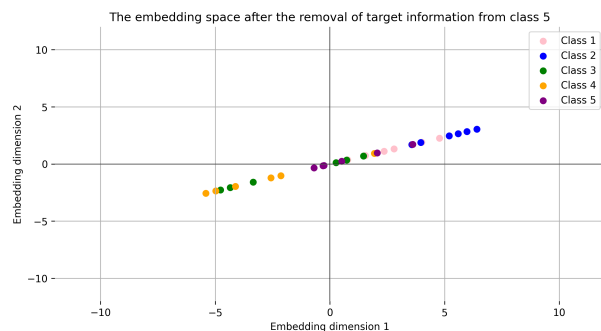Figure 2: Intuitive example of how to identify a target direction for removal in two-dimensional space.

Figure 3: Intuitive example of how the embedding space changes after the removal of one target direction in two-dimensional space.

B   Visualisation of the Amnesic Probing Experimental Set-up

A schematic representation of the steps involved in our experimental setup can be found in Figure **??**.

C   Results for Masked Setup

C.1   Probing Evaluation for Masked BERT

The results of the probing evaluation for masked BERT can be found in Table 6.

	dep	*f-pos*	*c-pos*
Vanilla	72.87	77.81	84.44
Vanilla LEACE	83.88	77.9	72.39
Amnesic INLP	**11.07**	**11.51**	27.5
Amnesic MP	15.44	15.94	**25.7**
Amnesic LEACE	25.69	16.13	16.14

Table 6: Results for probing for masked BERT encoding. In bold the lowest values achieved per property.

C.2   Evaluation Controls for Masked BERT

The results of the evaluation controls for masked BERT can be found in Table 7.

C.3   LM Task for Masked BERT

The results of the LM task for masked BERT can be found in Table 8.

C.4   Selectivity Results for Masked BERT

The results of the selectivity control for masked BERT can be found in Table 8.
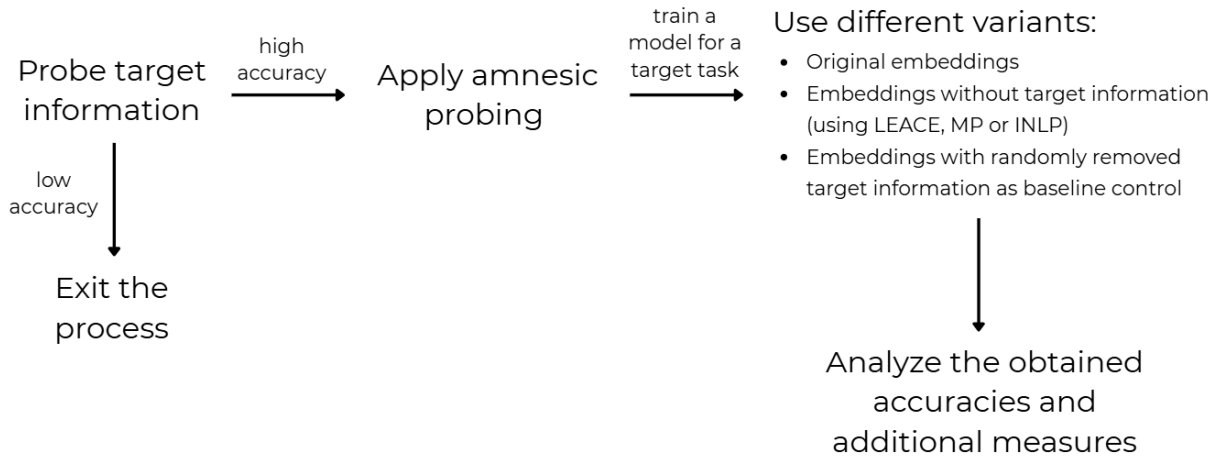
Probe target information →(high accuracy)→ Apply amnesic probing →(train a model for a target task)→ **Use different variants:**
- Original embeddings
- Embeddings without target information (using LEACE, MP or INLP)
- Embeddings with randomly removed target information as baseline control

Probe target information ↓(low accuracy) Exit the process

Analyze the obtained accuracies and additional measures

Figure 4: The conceptual visualisation of the amnesic probing framework.

	dep	*f-pos*	*c-pos*
Dir. removed INLP	**738**	**900**	**240**
Dir. removed MP	41	45	12
Dir. removed LEACE	41	45	12
Rank INLP Δ	**425**	**440**	**210**
Rank MP Δ	33	37	4
Rank LEACE Δ	32	34	10
Cosine random INLP	0.19	-0.03	0.83
Cosine random MP	0.97	0.97	0.99
Cosine random LEACE	0.97	0.97	0.99
Cosine similarity INLP	0.50	0.47	0.75
Cosine similarity MP	0.81	0.78	0.88
Cosine similarity LEACE	**0.89**	**0.86**	**0.93**

Table 7: Embedding evaluation for masked BERT. The original number of columns in the embedding is 768. The rank of the matrix for the test set before any modification is 760. Values in bold for Properties indicate the largest change amount of Directions removed and the largest change in the rank between the same properties. Values in bold for Cosine similarity indicate which information removal resulted in the highest cosine similarity between the original embedding and the modified one between the same properties.

		dep	*f-pos*	*c-pos*
Acc	Vanilla	56.98	56.98	56.98
	Random INLP	9.24	4.67	54.29
	Random MP	56.70	56.69	56.87
	Random LEACE	56.54	56.58	56.74
	Amnesic INLP	16.96	13.94	**33.29**
	Amnesic MP	**44.61**	**34.20**	**48.56**
	Amnesic LEACE	**48.20**	**37.60**	**50.44**
D_{kl}	Random INLP	7.06	7.77	0.49
	Random MP	0.03	0.04	0.01
	Random LEACE	0.03	0.04	0.01
	Amnesic INLP	5.75	6.08	3.1
	Amnesic MP	**1.45**	**2.09**	**0.86**
	Amnesic LEACE	**0.74**	**1.37**	**0.50**

Table 8: LM task for masked BERT encoding. Values in bold for LM-Acc Amnesic show which values obtained the accuracy lower than their Random and Dropout baselines. Values in bold for the $LM - D_{kl}$ indicate which properties with which method resulted in the lowest change of the general distribution of the tokens.

	dep	f-pos	c-pos
Amnesic INLP	4.1	5.43	2.96
Amnesic MP	1.63	1.7	1.54
Amnesic LEACE	0.70	1.59	0.22
Gold labels INLP	61.95	63.97	64.63
Gold labels MP	64.72	66.95	65.09
Gold labels LEACE	64.41	66.80	64.78
Accuracy INLP Δ	57.85	58.54	61.67
Accuracy MP Δ	63.09	**65.25**	63.55
Accuracy LEACE Δ	**63.71**	65.21	**64.56**

Table 9: LM task for masked BERT encoding. Values in bold for the difference in accuracy for Selectivity show the highest change between embeddings without the information and embeddings with returned gold label information.

D Experimental Details

D.1 Data details

The details of our dataset can be found in Table 10.

	no. sentences	no. tokens
train set	39832	1113133
test set	1700	47095

Table 10: Number of sentences and tokens used per Language Modeling (LM) task.

D.2 Model details

The details of our models can be found in Table 11.

E The Overall Impact on the Model

In this section, we describe the two additional metrics we used to study the impact of INLP, MP and LEACE on the model. The dimensions of a matrix do not necessarily indicate the true size of a linear system. If there are duplicate rows, or rows that are combinations of other rows, then, in the reduced echelon form, they are all zeros. To better understand the peculiarities of a given matrix, instead of referring to its dimensions, we include in the experiments the measure of matrix rank (Strang,

	model	Library
Probing	Linear classifier with SGD training	Sklearn
Task (LM)	Linear transformation $y = xA^T + b$ and Sequential container	Pytorch
Selectivity (LM)	Linear transformation $y = xA^T + b$	Pytorch
Rank of matrix	Matrix rank using SVD method	Numpy
Cosine Similarity		Sklearn

Table 11: Implementation details

2009). Matrix rank indicates a minimum number of linearly independent rows and columns in a given matrix. It does not give a direct measure of the amount of information present in the data, but it can give an indication of the decrease in linearly independent columns. This measure can provide additional information on how much the rank decreases when LEACE, MP or INLP is used. A larger change can potentially indicate a larger modification of the original data.

To further account for the difference in the embedding that LEACE, MP and INLP perform, we also include a cosine similarity measure. Cosine similarity is used to determine the degree of similarity between two vectors (Lahitani et al., 2016).[5] This measure can be a good additional check of

---

[5]It is calculated by dividing the vectors' dot product by the product of their lengths. The resulting value ranges from -1 to 1, where a value of 1 indicates the maximum similarity, 0 indicates no similarity, and -1 indicates the maximum dissimilarity.

how much the LEACE, MP and INLP methods modify the original data while removing target information. If one method correctly identifies and eliminates the target property, but modifies the original space less than the other method, then that method should be preferred for the Amnesic Probing analysis. The cosine similarity between the test data before and after information removal is calculated by first taking the cosine similarities between all corresponding rows from the original and modified data and then averaging the result.

In Table 12 the number of removed directions, the change in the rank and the cosine similarity between the original and the modified data are given for each method and linguistic property. The Rank Δ gives an estimate of how much the rank of the matrix has changed after the information removal.

It can be observed that the number of directions removed and the change in the rank is the largest for INLP. It is also worth noting that as the number of directions removed increases, so does the change in the rank of the matrix, meaning that less linearly independent information is encoded. One outstanding value for the change of the rank for MP is -17 for the *c-pos* property.[6]

Looking at the cosine similarity results, we can see that MP and LEACE modify the embedding space in such a way that the similarity between original and modified data is still quite high, with values between 0.80-0.91 for MP and 0.89-0.95 for LEACE depending on the linguistic property. In the case of INLP, the similarity with the original embedding after information removal is much lower for two out of three properties, for the linguistic properties *dep* and *f-pos* it is 0.31 and 0.37 respectively. The *c-pos* property is an exception for INLP, as it achieves a cosine similarity of 0.80, probably due to the lower amount of direction removed compared to the other properties. In Table 12 there is additional indication of Cosine similarity when random directions are removed. In row Rand we can see the relation between the amount of directions removed and the cosine similarity value. The fewer directions are removed, the more the modified embedding resembles the original one.

In summary, LEACE, MP and INLP differ in

	dep	*f-pos*	*c-pos*
Dir. removed INLP	**779**	**765**	**240**
Dir. removed MP	41	45	12
Dir. removed LEACE	41	45	12
Rank INLP Δ	**629**	**617**	**198**
Rank MP Δ	12	16	-17
Rank LEACE Δ	31	34	9
Cosine random INLP	-0.04	0.05	0.83
Cosine random MP	0.97	0.96	0.99
Cosine random LEACE	0.97	0.99	0.97
Cosine similarity INLP	0.31	0.37	0.80
Cosine similarity MP	0.83	0.80	**0.91**
Cosine similarity LEACE	**0.92**	**0.95**	0.89

Table 12: Embedding evaluation for non-masked BERT. The original number of columns in the embedding is 768. The rank of the matrix for the test set before any modification is 739. Values in bold for Properties indicate the largest amount of directions removed and the largest change in the rank between the same properties. Values in bold for Cosine similarity indicate which information removal resulted in the highest cosine similarity between the original embedding and the modified one between the same properties.

---

[6]This value indicates that after the projection the rank of matrix increased by 17. There is no clear answer as to why this opposite trend occurred. One of the possible explanations is that because MP and LEACE only removed 12 directions for the *c-pos* property, instead of losing linearly independent rows and columns, we actually gained some because the projection removed some noise.

how they modify the general embedding space. INLP removes many more directions and also reduces the rank of the matrix to a greater extent. The cosine similarity between the original data and the data with the target information removed is higher for the MP and LEACE methods. This suggests that MP and LEACE can potentially preserve the general embedding space better while identifying and removing the target linguistic property as well as INLP.

F Computing Resources

The experiments were run on a local laptop with the following specifications:

- Processor: AMD Ryzen™ 7 4800H (8 cores, 16 threads, 2.90-4.20 GHz, 12 MB cache)

- Graphics card: NVIDIA GeForce GTX 1660 Ti AMD Radeon™ Graphics

It took around 8-12 hours to run selectivity for one case (e.g. MP and dependency as target information) and around 4-6 hours to run the rest of the evaluation.