Reconstruction Probing

Najoung Kim,[†] **Jatin Khilnani**,[△] **Alex Warstadt**,[δ] **and Abed Qaddoumi**[ρ]
[†]Boston University [△]University of Pittsburgh [δ]ETH Zurich [ρ]Independent
najoung@bu.edu jatin.khilnani@pitt.edu
awarstadt@ethz.ch abdulrahim.qaddoumi@gmail.com

Abstract

We propose *reconstruction probing*, a new analysis method for contextualized representations based on reconstruction probabilities in masked language models (MLMs). This method relies on comparing the reconstruction probabilities of tokens in a given sequence when conditioned on the representation of a single token that has been fully contextualized and when conditioned on only the decontextualized lexical prior of the model. This comparison can be understood as quantifying the contribution of contextualization towards reconstruction—the difference in the reconstruction probabilities can only be attributed to the representational change of the single token induced by contextualization. We apply this analysis to three MLMs and find that contextualization boosts reconstructability of tokens that are close to the token being reconstructed in terms of linear and syntactic distance. Furthermore, we extend our analysis to finer-grained decomposition of contextualized representations, and we find that these boosts are largely attributable to static and positional embeddings at the input layer.

1 Introduction

Model building in contemporary Natural Language Processing usually starts with a neural network pretrained on the objective of context reconstruction ("language modeling"). Contextualized representations of complex linguistic expressions from such models have been shown to encode rich lexical and structural information (Tenney et al., 2019b; Rogers et al., 2020), making these models an effective starting point for downstream applications.

Probing pretrained language models aims to understand the linguistic information they encode, and how well it aligns with our understanding of human language (see Belinkov 2022 for a review). The methodologies employed include supervised classifiers targeting specific linguistic properties of interest (Ettinger et al. 2016; Giulianelli et al. 2018; Tenney et al. 2019a; Conia and Navigli 2022), similarity-based analyses (Garí Soler and Apidianaki, 2021; Lepori and McCoy, 2020), cloze-type tests (Goldberg, 2019; Pandit and Hou, 2021), and causal intervention-based methods (Vig et al., 2020; Elazar et al., 2021; Geiger et al., 2021). This methodological diversity is beneficial given the high variability of conclusions that can be drawn from a study using a single method (Warstadt et al., 2019)—converging evidence is necessary for a more general picture.

We contribute to this line of research with a new analysis method that we name *reconstruction probing*, which relies on token probabilities obtained from context reconstruction, applicable to models pretrained on objectives of this kind.[1] Our method is characterized by two core properties. First, it is causal: rather than asking "what features can we extract from the contextualized representations?", we ask "what effect does contextual information have on the model predictions?" through intervention at the input level. Second, our method is behavioral: it relies on the context reconstruction objective that the model was trained on. This obviates the need to train specialized probes, which can be difficult to interpret due to the added confound of task-specific supervision.

Our method aims to probe how much information the contextualized representation of a **single** token contains about the other tokens that co-occur with it in a given sequence in masked language models. Our approach is to measure the difference between the reconstruction probability of a co-occurring token in the sequence given the full contextualized representation being probed, and the reconstruction probability of the same co-occurring token only from the lexical priors of the model. This method can be generalized to compare two arbitrary representations where one representation

[1]Code and data available at https://github.com/najoungkim/mlm-reconstruction.

Findings of the Association for Computational Linguistics: ACL 2023, pages 8240–8255
July 9-14, 2023

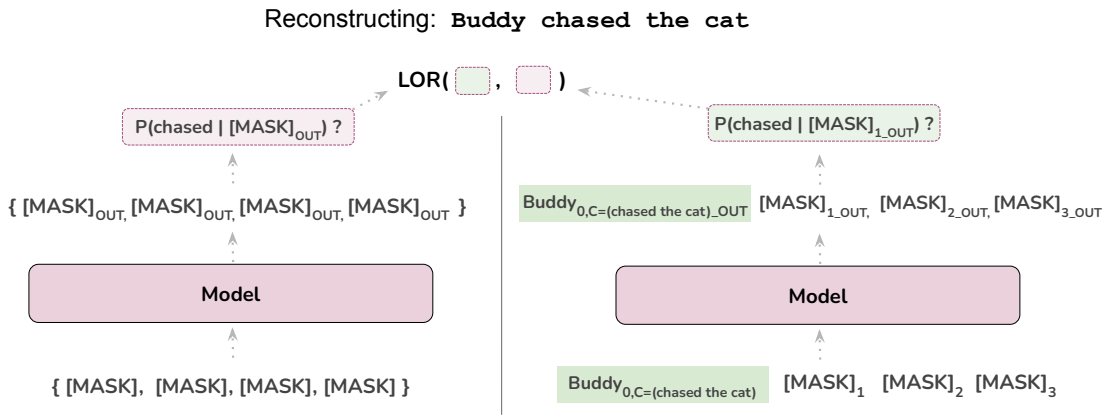


Figure 1: (Left) How the probability of *chased* from only the lexical priors of the model is obtained. The input to the model is a sequence of masked tokens of the same length as the original sentence, without any positional embeddings. (Right) How the probability of *chased* given a fully contextualized representation of the token *Buddy* is computed (see Figure 2 for more details). The reconstruction probabilities from (Left) and (Right) are compared using log odds ratio (LOR; Eq. 1).

is expected to contain strictly more features than the other (e.g., a static embedding of a token vs. an embedding of the same token created by summing the static embedding and its positional embedding in context). Any difference between the reconstruction probabilities can be attributed to the presence/absence of those features.

Using this method, we find that the contextualized representation of a token contains more information about tokens that are closer in terms of linear and syntactic distance, but do not necessarily encode identities of those tokens. A follow-up analysis that decomposes contextualized representations furthermore shows that the gains in reconstructability we find are largely attributable to static and positional embeddings at the input layer.

2 Proposed Approach

Pretrained Transformer models such as BERT (Devlin et al., 2019) learn to construct contextual representations through context reconstruction objectives like masked language modeling (MLM; e.g., predicting the token in place of `[MASK]` in *The* `[MASK]` *sat on the mat*). Often, the models are also trained to reconstruct a randomly substituted token (e.g., predicting the token in place of *door* in *The cat sat door the mat*, created by randomly substituting a word in *The cat sat on the mat*). The classifier that makes these predictions can only make use of a single token representation from the final layer, meaning these representations are optimized to contain information about other tokens of the sequence and the position of the token itself insofar as this information can help to resolve the identity of the token. Our approach aims to quantify how much the contextualization of these tokens contributes to changing the MLM predictions.

2.1 Metric

We operationalize *contextual informativeness* of a token representation as its contribution to predicting other tokens in the same sequence—i.e., the contribution to the MLM probability, or *reconstruction probability*. We quantify the contribution of a more informative token representation j^{++} towards reconstructing a different token i, by comparing the reconstruction probability $P(i|j^{++})$ to the reconstruction probability of i given a less informative token representation j, $P(i|j)$.

For example, you can obtain the contextualized representation of *Buddy* in the input sequence *Buddy chased Cookie* by passing this through a model. If $Buddy_{contextual}$ encodes information helpful for predicting *chased*, the masked language modeling probability $P(chased|C[\texttt{[MASK]}_1, Buddy_{contextual}])$ would be higher than $P(chased|C[\texttt{[MASK]}, \emptyset])$—the lexical prior of the model for *chased*.[2] The difference between these probabilities is measured

[2]We use C[`[MASK]`$_{pos}$, SOURCE] to refer to the contextualized representation of the `[MASK]` token at position *pos* at the output layer of the model, which is the input to the final classifier that produces the probability distribution for masked token prediction. See Section 2.2 for a full description of how we compute reconstruction probabilities.

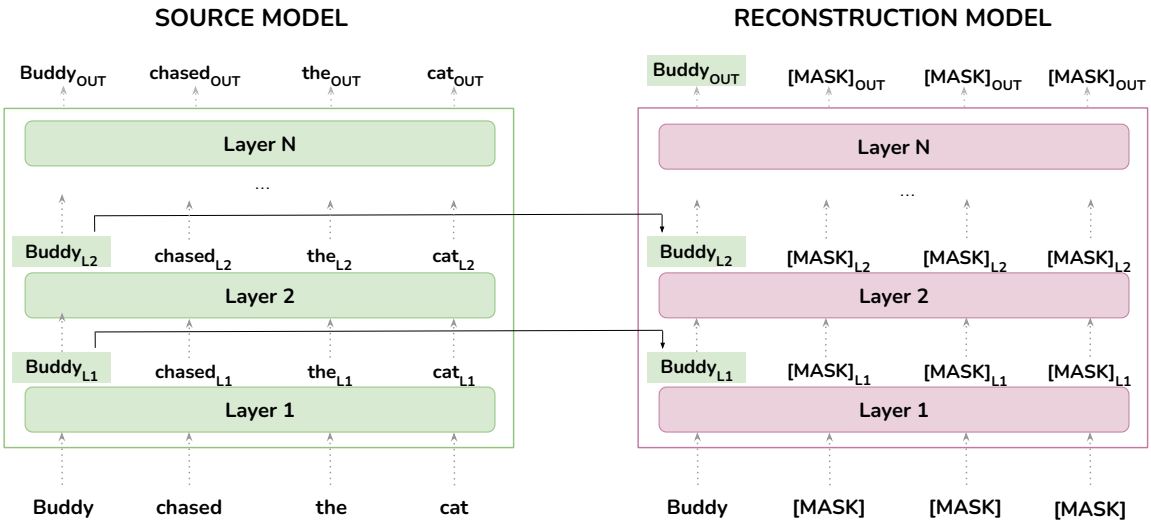


Figure 2: Diagram of the fully contextualized reconstruction setting, providing more details about how the right hand side of Figure 1 is implemented.

in terms of the log odds ratio given the base reconstruction probability q (predicting from less context) and the contextualized reconstruction probability p (predicting from more context):

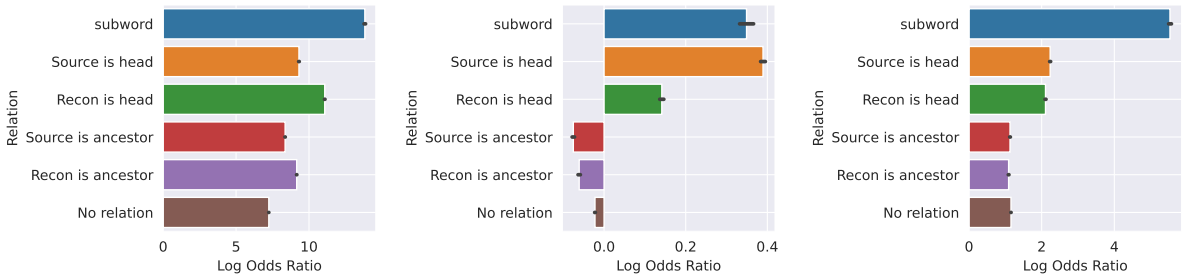
$$\text{LOR}(p, q) = \ln\left(\frac{p/(1-p)}{q/(1-q)}\right) \quad (1)$$

The probabilities p and q are defined with respect to SOURCE and RECONSTRUCTION (shortened as RECON) tokens. SOURCE tokens refer to tokens that are revealed to the model at prediction time (e.g., *Buddy* in the running example). RECON tokens are tokens in the original sequence the model is asked to predict (e.g., *chased* in the running example). In obtaining probabilities p and q, the RECON tokens are replaced with `[MASK]` tokens, only leaving the SOURCE token revealed to the model (more detailed description is given in Section 2.2). MLM probability of the token in the original sequence is computed for each `[MASK]` token in the probe input—for instance, for $Buddy_{contextual}$ [MASK] [MASK], we compute the probability of *chased* at position 1 given this sequence, and *Cookie* at position 2 given this sequence. We compute Eq. 1 for every pair of tokens (t_i, t_j) in a given sequence, where t_i is SOURCE and t_j is RECON. This value represents the degree of change in the probability of the reconstruction token t_j induced by the contextualization of the source token t_i.

2.2 Obtaining the Reconstruction Probabilities

We use the metric proposed above to gauge the contribution of a contextualized representation of a single token in reconstructing its context, over and above the lexical prior (i.e., completely context-independent) of the model as illustrated in Figure 1. We describe below how the reconstruction probabilities from a fully contextualized representation and from the lexical prior of the model are obtained.

Fully Contextualized To obtain a fully contextualized representation of a token in a particular sequence (e.g., *Buddy chased Cookie*), we first pass the original, unmasked sequence of tokens through a masked language model. Here, we save each contextualized token representation at each layer of the model (e.g., $Buddy_{L1}$, $Buddy_{L2}$, ..., $Buddy_{Lm}$ where m is the number of layers). Then, we create n ($n = |seq|$) versions of the input sequence where only a single token is revealed (*Buddy* `[MASK] [MASK]`, `[MASK]` *chased* `[MASK]`, `[MASK] [MASK]` *Cookie*). We pass each sequence through the same masked language model, but at each layer, we replace the representation of the unmasked token with the stored contextualized representation of that token (see Figure 2 for an illustration). Then, in order for the masked language modeling head to predict each `[MASK]` token in the sequence, it can only rely on the information from the representation of the single unmasked token (SOURCE), where the SOURCE token representation is contextualized with respect to the original,



(a) BERT (b) DistilBERT (c) RoBERTa

Figure 3: Reconstructibility boost by syntactic relation measured by log odds ratio.

fully unmasked sequence. For each [MASK] token in the sequence, we take the probability of the token in the same position in the original sequence as the reconstruction probability. For example, $P(\textit{chased}|C[\texttt{[MASK]}_1, \textit{Buddy}_{contextual}])$ and $P(\textit{Cookie}|C[\texttt{[MASK]}_2, \textit{Buddy}_{contextual}])$ are the reconstruction probabilities of *chased* and *Cookie*, respectively, given the representation of fully contextualized *Buddy*.

Lexical Prior Only Baseline We pass through a fully masked version of the input sequence as above, but do not add the positional embeddings at the input layer. The reconstruction probability that we obtain here corresponds to the probability of predicting the token in the original sequence in the absence of any lexical information *and* positional information. We expect this probability to reflect a general prior of the model over the vocabulary, for instance based on frequency in the training corpus.

3 Experiment Setup

3.1 Models

We analyzed three Transformer-based masked language models widely used for obtaining contextualized representations: BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019) and DistilBERT (Sanh et al., 2019). BERT and RoBERTa were both pretrained using the masked language modeling objective (BERT also on Next Sentence Prediction), and DistilBERT is a more compact version of BERT obtained through knowledge distillation. DistilBERT has been claimed to retain much of the downstream task performance of BERT despite being substantially smaller (Sanh et al., 2019), and has been shown to be highly similar to BERT in terms of constituency trees that can be reconstructed from linear probes (Arps et al., 2022).

3.2 Data

We used sentences from the Multi-Genre Natural Language Inference (MNLI; Williams et al. 2018) dataset for this analysis. We selected MNLI because it contains sentences of varying lengths from a range of domains, and is not a part of the pretraining data of the models we are probing. We then sampled 10K premise sentences from the non-spoken genres of the dataset (i.e., excluding TELEPHONE and FACE-TO-FACE). We excluded spoken data as it is less typical of the data domain the models were trained on, and we excluded hypothesis sentences because they were generated by crowdworkers given the naturally-occurring premises.

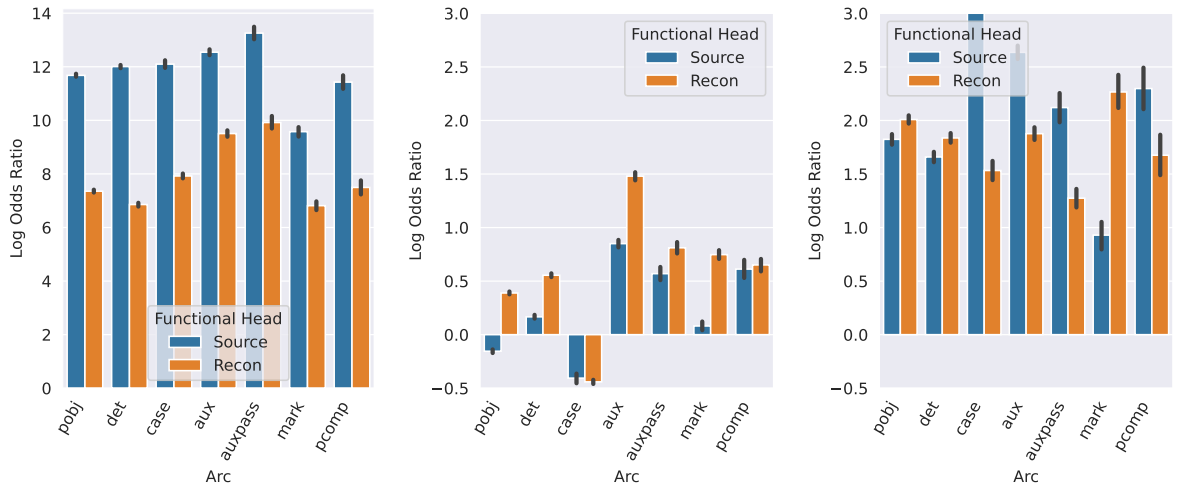
3.3 Procedure

For each of the 10K sentences, we created two different sets of probe inputs as illustrated in Figure 1. We passed the probe inputs to the models to obtain the two different reconstruction probabilities (from lexical prior only vs. from a fully contextualized source token) of each of the tokens in the input, as described in Section 2.2. Finally, we computed the log odds ratios between the two reconstruction probabilities using Eq. 1 to quantify the contribution of contextualization for all possible (SOURCE, RECON) token pairs in the original sentence.

4 Analyses

4.1 Is Token Identity Exactly Recoverable from Contextualized Representations?

The RECON token is among the top 10 MLM predictions of the model only a small percent of the time (BERT: 22.1%. RoBERTa: 7.9%, DistilBERT: 8.2%), even though the SOURCE token provided to the model has been contextualized with all co-occurring tokens revealed. This observation suggests that the information encoded in the



(a) BERT (b) DistilBERT (c) RoBERTa

Figure 4: Reconstructibility boost (log-odds ratio with vs. without source) broken down by the functional relation between a functional head and a content-word dependent.

contextualized representations is a degree more abstract than directly encoding the identities of co-occurring tokens in the same sequence. This is in line with Klafka and Ettinger's (2020) finding that the features of co-occurring tokens rather than their identities are often more recoverable from the contextual representations.

4.2 Is Reconstructability Greater when Tokens are in a Syntactic Relation?

We hypothesize that the contextual information in an embedding should disproportionally reflect the syntactic neighbors of the word. To test this hypothesis, we partition reconstructability scores based on the syntactic relation between the SOURCE and RECON tokens as follows:[3] (1) **SOURCE/RECON is head**: Cases where there is a single dependency arc between two tokens, the closest dependency relation possible with the exception of subword tokens. Reconstructing *cat* from *chased* in Figure 5 would be a case of SOURCE is head, and *chased* from *cat* would be RECON is head. (2) **SOURCE/RECON is ancestor**: Cases where there is more than one dependency arc connecting the two tokens. Reconstructing *the* from *chased* would be a case of SOURCE is ancestor, and *chased* from *the* would be RECON is ancestor. (3) **subword**: SOURCE/RECON tokens are subwords of the same lexical item. *Bud* and *##dy* is an example. (4) **No relation**: None of the above relations holds. For example, tokens *Bud* and *the* are not in a dependency relation.

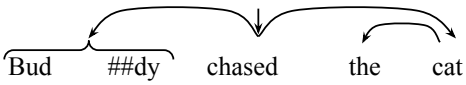


Figure 5: The dependency parse of the sentence *Buddy chased the cat*.

Our results in Figure 3 confirm our hypothesis. In general, we find that the degree to which contextual information improves reconstruction depends on the existence of a syntactic relation between the SOURCE and RECON as expected. In all models, tokens in a subword or head-dependent relation are more reconstructable from each other compared to tokens with no relation. Furthermore, among tokens that are in a dependency relation, the closer the relation, the higher the reconstruction boost: reconstruction boost is the greatest for tokens in a subword relation, then for tokens in a head-dependent relation, and then for tokens in ancestor-descendant relation. These trends were consistent across all models we evaluate, with the exception of DistilBERT where reconstruction boost when SOURCE is head was greater than tokens in a subword relation. The models showed more variation in whether ancestor relations boosted reconstructability significantly. While tokens in an ancestor-descendant relation (excluding direct dependents) were more reconstructable than tokens not in a dependency relation in BERT, this was not the case for RoBERTa and DistilBERT. We also did not find a large or con-

[3]For these and subsequent analyses, we parse the sentences using the spaCy dependency parser (https://spacy.io/models/en#en_core_web_trf).

sistent effect of whether the SOURCE token or the RECON token is the ancestor (including direct head-dependent relations). Thus we cannot conclude that ancestors tend to contain more information about descendants than vice-versa.

4.3 Finer-Grained Syntactic Properties

In the next set of analyses, we study how fine-grained syntactic properties of the words affect reconstructability, focusing on cases where there is a syntactic relation between SOURCE and RECON.

Dependency Relations One natural way to break down the results is by the label of the dependency relation that holds between SOURCE and RECON when such a relation exists. However, we did not find overarching trends; results were generally idiosyncratic, although boost for token pairs in ROOT and PRT (particle) relations was high across all models. See Appendix A for full results.

Functional Relations Next, we zoom in on relations between functional heads and their content-word dependents (Figure 4). Table 1 lists all the dependency arcs we use to identify functional heads.[4] First, we find that reconstructability is generally high for these pairs. Second, auxiliary-verb relations are associated with particularly high reconstructability for all models. One possible explanation for this finding is the fact that there is always morphological agreement between auxiliaries and verbs, unlike most other functional relations. Third, among functional relations, reconstructability is always lowest for complementizer-verb relations (labeled *mark*). We speculate that the complementizer might encode contextual information about the entire complement clause, which often includes many more content words than just the head verb.

We hypothesized that functional heads encode more information about their dependents in context than vice-versa, due to function words carrying less information than content words, but their contextual representations are equal in size, leaving more space for information about the rest of the sentence. Results from BERT support the hypothesis for all relations. On the other hand, no consistent asymmetry was observed for RoBERTa, and for DistilBERT, the observed pattern mostly contradicts our hypothesis. The large difference between BERT and DistilBERT results goes against prior results that suggest that the syntactic trees recoverable from these two models are highly similar (Arps et al., 2022).

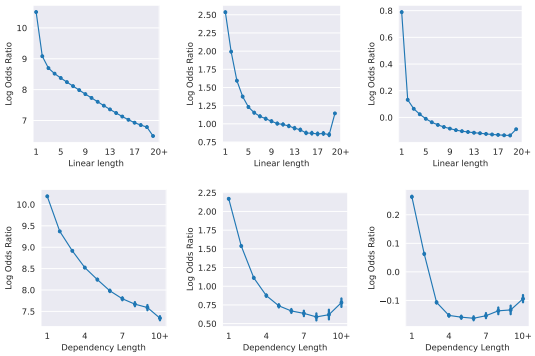
[4]While function words are typically considered heads of content words in linguistic theory, the opposite is often true in dependency labeling schemes.

Relation	FW is ...	Example
aux	Dependent	*The dog is sleeping.*
auxpass	Dependent	*The dog was taken out.*
case	Dependent	*The dog 's bone is gone.*
det	Dependent	*The dog barked.*
mark	Dependent	*I think that the dog ate.*
pcomp	Head	*I dream about dogs playing.*
pobj	Head	*I played with the dog.*

Table 1: Dependency arcs denoting functional relations. **FW is ...** indicates whether the function word is considered the head or the dependent in the clearNLP labeling scheme used by spaCy (https://github.com/clir/clearnlp-guidelines).

4.4 Linear and Structural Distance

We also hypothesized that the distance between two tokens (both in linear and structural terms) would affect reconstruction. Linear distance is the difference between the linear indices of SOURCE and RECON: if they are the i^{th} and j^{th} tokens respectively, their linear distance is $|i - j|$. Structural distance is the number of arcs in the directed path between SOURCE and RECON tokens (if there is a path). For example, in Figure 5 the structural distance between *the* and *chased* is 2.



(a) BERT (b) RoBERTa (c) DistilBERT

Figure 6: Reconstructibility boost (log odds ratio) broken down by linear distance (top) and structural distance (bottom) between SOURCE and RECON.

Linear Distance Predictably, we find that information encoded in contextualized representations is biased towards nearby tokens in linear space (Figure 6, row 1). In other words, we find that reconstructability generally decreases with increase

	Ablated sequence	Probe input (SOURCE == 'Buddy')
Sequence to reconstruct	Buddy chased Cookie	-
Fully contextualized	$\text{Buddy}_{contextual}$ $\text{chased}_{contextual}$ $\text{Cookie}_{contextual}$	$\text{Buddy}_{contextual}$ [MASK] [MASK]
Static lexical embedding (+position)	Buddy_{static} chased_{static} Cookie_{static}	Buddy_{static} [MASK] [MASK]
Static lexical embedding (-position)	{Buddy_{static}, chased_{static}, Cookie_{static}}	{Buddy_{static}, [MASK], [MASK]}
All mask (+position)	[MASK] [MASK] [MASK]	[MASK] [MASK] [MASK]
All mask (-position) (Lexical prior only)	{[MASK], [MASK], [MASK]}	{[MASK], [MASK], [MASK]}

Table 2: Ablated sequence and an example of an input passed through the model to obtain the output representations when SOURCE is 'Buddy'. {} denotes an unordered set (i.e., no positional information).

in linear distance. For all models, the sharpest decrease is observed between 1- and 2-token distances. Beyond this, reconstructability decreases approximately linearly in BERT, and more gradually in RoBERTa and DistilBERT.

Structural Distance The second row of Figure 6 shows the decline in reconstructability as the number of intervening nodes in the dependency path between the tokens increases when comparing reconstruction. This trend is strictly monotonic in BERT, but there is an small increase starting from dependency depth 7 in RoBERTa and DistilBERT. Due to the high variance in the deeper depth cases, it is unclear whether this is a genuine effect of contextualization.

5 Decomposing Contextualization

While we examined the effect of contextualization compared to the lexical prior only baseline, our method allows for a finer-grained decomposition of the components of contextualization. In pretrained Transformer models, the input representation of a token is a function of the static lexical embedding and a (context-specific) positional embedding. Using our method, we can study the individual influence of the lexical embedding, positional embedding, and remaining sequence-specific contextualization (i.e., everything that happens beyond the input layer, *full contextualization* henceforth).

We create various ablated versions of a fully contextualized sequence, as shown in the **Ablated sequence** column of Table 2. The reconstruction probabilities from these ablated sequences allow us to probe the contribution of the various components of contextualized language models. **Fully contextualized** and **All mask (-position)** in Table 2 correspond to the reconstruction probabilities described and compared in Section 2.2, and the rest are intermediate ablations.

5.1 Results

Surprisingly, we find that there is often no clear benefit to reconstruction of providing the model with the contextualized embeddings at each layer, over just providing the input embedding (lexical + positional embeddings) of the source token (Figure 7, bottom). While BERT does gain reconstructability from full contextualization for subwords and when SOURCE is a head/ancestor, contextualization is generally harmful or at least not helpful to reconstruction for RoBERTa and DistilBERT. This indicates that the positive reconstruction boost observed in Figure 3 must be driven by static lexical and positional embeddings. Indeed, there are generally positive gains in reconstructability in models provided with the lexical embeddings of the SOURCE tokens compared to models given only [MASK] tokens (Figure 7, top), and also in models provided with positional embeddings on top of lexical embeddings (Figure 9, middle column; Appendix B.3). We provide full comparisons between ablations and their interpretation in Appendix B.

When is full contextualization helpful/harmful? To better understand the effect of full contextualization, we manually examined token pairs where the greatest differences in reconstruction probabilities with the static lexical + positional and fully contextual SOURCE tokens. In BERT and DistilBERT, the majority (52% and 80%) of the 100 most helpful scenarios of full contextualization involved reconstruction of an apostrophe in a contraction from single-character or bi-character tokens (e.g., *m*, *t*, *re*). As the source token is highly ambiguous on its own, contextualization seems to provide additional information that these (bi)character tokens are a part of a contraction (e.g., *I'm*, *wasn't*, *we're*. In RoBERTa, we found no interpretable pattern.

Cases where full contextualization negatively affected reconstruction were often when SOURCE and RECON formed a common bigram (e.g., (*prix*,

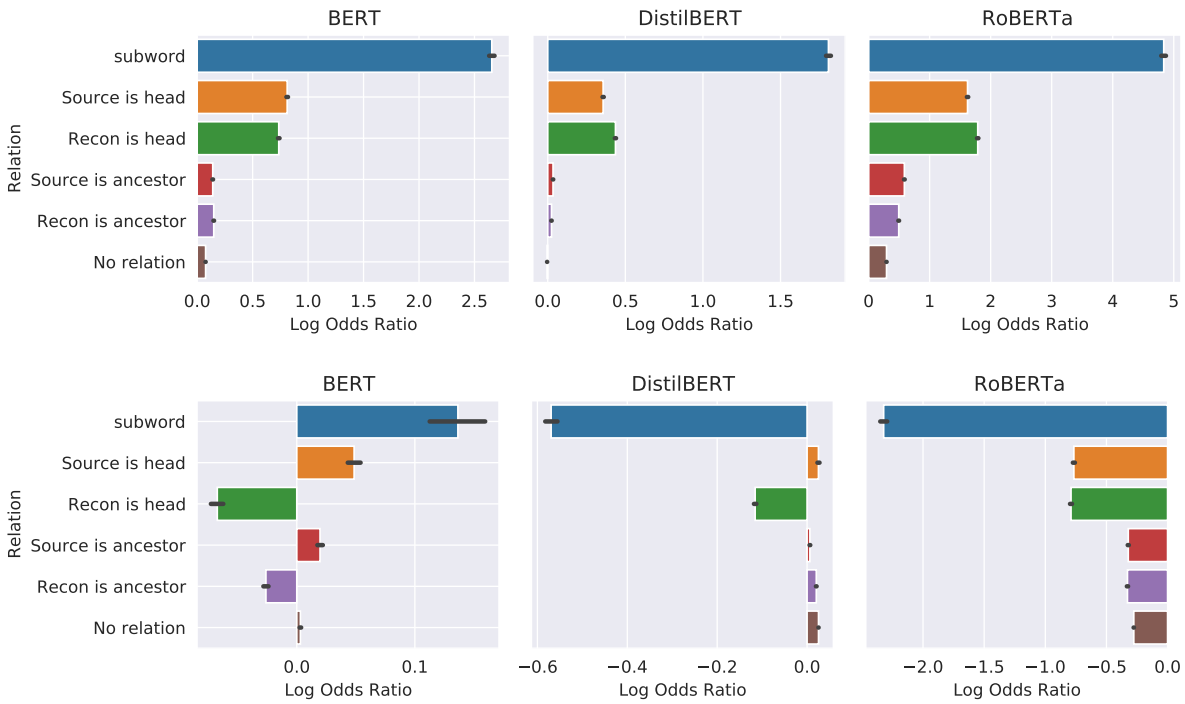


Figure 7: Relative reconstructibility (log odds ratio) for static lexical (+pos) and all masked (+pos) SOURCE embeddings (top) and for fully contextualized and static lexical (+pos) SOURCE embeddings (bottom).

grand), (*according*, *to*), (*##ritan*, *pu*), (*United*, *States*)). Since the RECON token is predictable from SOURCE alone, full contextualization seems to only dilute the signal.

Although we found that reconstruction is often better given only input embeddings (i.e., static + positional embeddings) than fully contextualized embeddings, we take caution with the interpretation that full layerwise contextualization is in general harmful to the models, especially given prior evidence (Tenney et al., 2019a) that transformations across layers yield meaningful changes. One possible interpretation is that the idiosyncrasy of the procedure for transferring the contextualized source token falls outside the setting in which these models were trained, adding noise to the process.

6 Related Work

Our research question is similar to Klafka and Ettinger (2020) which use supervised classifiers to investigate how much information about other tokens in context is contained in the contextualized representation(s) of a token. Our approach addressed a similar question through reconstruction probability given more/less informative token representations. Our findings about better reconstructability between tokens in a syntactic dependency relation echo prior work that show sensitivity of MLMs to part-of-speech and other syntactic relations (Tenney et al., 2019b; Goldberg, 2019; Htut et al., 2019; Kim and Smolensky, 2021). A novel finding is that some of the syntactic dependency between tokens can be traced back to information in the input embeddings, complementing the dynamic layerwise analysis in work such as Tenney et al. (2019a) and Jawahar et al. (2019). This result aligns with Futrell et al. (2019)'s observation that syntactic dependency is reflected in the corpus distribution as encoded in static embeddings. Existing work that analyzes static embeddings from contextualized models (Bommasani et al., 2020; Chronis and Erk, 2020; Sajjad et al., 2022) mostly concerns the *distillation* of static embeddings rather than isolating the contribution of static embeddings in contextualized prediction as in our work. More broadly, our work shares goals with intervention-based methods such as Geiger et al. (2021) and Wu et al. (2020), but we examine what the effect of our intervention is on masked language modeling probabilities rather than on separate downstream tasks. Karidi et al. (2021) employs the most similar methodology to ours, in their use of predictions from the masked language modeling objective directly for probing. However, their primary analysis concerns the role

of contextualization in word sense disambiguation.

7 Conclusion

We proposed *reconstruction probing*, a novel method that compares reconstruction probabilities of tokens in the original sequence given different amounts of contextual information. Overall, reconstruction probing yields many intuitive results. We find that the information encoded in these representations tend to be a degree more abstract than token identities of the neighboring tokens—often, the exact identities of co-occurring tokens are not recoverable from the contexutalized representations. Instead, reconstructability is correlated with the closeness of the syntactic relation, the linear distance, and the type of syntactic relation between the SOURCE and RECON tokens. These findings add converging evidence to previous probing studies about the implicit syntactic information of contextual embeddings (Tenney et al. 2019b). Furthermore, our method is generalizable to comparing reconstruction probabilities from any pair of representations that differ in the degree of informativeness. We extended our analysis to finer-grained decomposition of the components that constitute contextualized representations using this method, finding that most of the reconstruction gains we saw were attributable to information contained in static lexical and positional embeddings at the input layer. This calls for deeper investigations into the role of token representations at the input layer, complementing a large body of existing work on layer-wise analysis of contextualized language models.

Limitations

As we discussed in Section 5.1, further work is needed to investigate whether the negative effect of full contextualization beyond static + positional embeddings at the input layer is an idiosyncrasy of the embedding transfer procedure, or if this is a true effect. In future work, an experimental setup that is closer to the training setup, such as masking only the RECON token instead of all tokens and transferring the SOURCE could be adopted, in order to reduce the noise potentially introduced by the distributional change in the inputs. Regardless, we believe that findings regarding the information content of the representation at the input layer (static + positional embeddings) are novel and meaningful, and the quantification method we propose for comparing two representations in terms of their predictive utility is a generalizable methodological contribution.

We furthermore note that our attempts to conduct evaluation on newer masked language models were made challenging due to several technical issues in the library (e.g., masked language modeling being unavailable in DeBERTa (He et al., 2021): https://github.com/huggingface/transformers/pull/18674).

Acknowledgments

We thank Sebastian Schuster, Grusha Prasad, Sophie Hao, and the members of the NYU Computation and Psycholinguistics lab for helpful discussions. This research was conducted through the NYU IT High Performance Computing resources, services, and staff expertise.

References

David Arps, Younes Samih, Laura Kallmeyer, and Hassan Sajjad. 2022. Probing for constituency structure in neural language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6738–6757, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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

Rishi Bommasani, Kelly Davis, and Claire Cardie. 2020. Interpreting Pretrained Contextualized Representations via Reductions to Static Embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4758–4781, Online. Association for Computational Linguistics.

Gabriella Chronis and Katrin Erk. 2020. When is a bishop not like a rook? when it's like a rabbi! multi-prototype BERT embeddings for estimating semantic relationships. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 227–244, Online. Association for Computational Linguistics.

Simone Conia and Roberto Navigli. 2022. Probing for predicate argument structures in pretrained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4622–4632, Dublin, Ireland. 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.

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.

Allyson Ettinger, Ahmed Elgohary, and Philip Resnik. 2016. Probing for semantic evidence of composition by means of simple classification tasks. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 134–139, Berlin, Germany. Association for Computational Linguistics.

Richard Futrell, Peng Qian, Edward Gibson, Evelina Fedorenko, and Idan Blank. 2019. Syntactic dependencies correspond to word pairs with high mutual information. In *Proceedings of the Fifth International Conference on Dependency Linguistics (Depling, SyntaxFest 2019)*, pages 3–13, Paris, France. Association for Computational Linguistics.

Aina Garí Soler and Marianna Apidianaki. 2021. Let's play mono-poly: BERT can reveal words' polysemy level and partitionability into senses. *Transactions of the Association for Computational Linguistics*, 9:825–844.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. 2021. Causal abstractions of neural networks. In *Advances in Neural Information Processing Systems*, volume 34, pages 9574–9586. Curran Associates, Inc.

Mario Giulianelli, Jack Harding, Florian Mohnert, Dieuwke Hupkes, and Willem Zuidema. 2018. Under the hood: Using diagnostic classifiers to investigate and improve how language models track agreement information. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 240–248, Brussels, Belgium. Association for Computational Linguistics.

Yoav Goldberg. 2019. Assessing BERT's syntactic abilities. *arXiv:1901.05287*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTa: Decoding-enhanced BERT with disentangled attention. In *International Conference on Learning Representations*.

Phu Mon Htut, Jason Phang, Shikha Bordia, and Samuel R. Bowman. 2019. Do attention heads in BERT track syntactic dependencies? *arXiv:1911.12246*.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Taelin Karidi, Yichu Zhou, Nathan Schneider, Omri Abend, and Vivek Srikumar. 2021. Putting words in BERT's mouth: Navigating contextualized vector spaces with pseudowords. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10300–10313, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Najoung Kim and Paul Smolensky. 2021. Testing for grammatical category abstraction in neural language models. In *Proceedings of the Society for Computation in Linguistics 2021*, pages 467–470, Online. Association for Computational Linguistics.

Josef Klafka and Allyson Ettinger. 2020. Spying on your neighbors: Fine-grained probing of contextual embeddings for information about surrounding words. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4801–4811, Online. Association for Computational Linguistics.

Michael Lepori and R. Thomas McCoy. 2020. Picking BERT's brain: Probing for linguistic dependencies in contextualized embeddings using representational similarity analysis. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3637–3651, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv:1907.11692*.

Onkar Pandit and Yufang Hou. 2021. Probing for bridging inference in transformer language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4153–4163, Online. Association for Computational Linguistics.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Hassan Sajjad, Firoj Alam, Fahim Dalvi, and Nadir Durrani. 2022. Effect of post-processing on contextualized word representations. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3127–3142, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. In *The 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing, NeurIPS 2019*.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019a. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. 2019b. What do you learn from context? Probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations*.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. 2020. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, volume 33, pages 12388–12401. Curran Associates, Inc.

Alex Warstadt, Yu Cao, Ioana Grosu, Wei Peng, Hagen Blix, Yining Nie, Anna Alsop, Shikha Bordia, Haokun Liu, Alicia Parrish, Sheng-Fu Wang, Jason Phang, Anhad Mohananey, Phu Mon Htut, Paloma Jeretic, and Samuel R. Bowman. 2019. Investigating BERT's knowledge of language: Five analysis methods with NPIs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2877–2887, Hong Kong, China. Association for Computational Linguistics.

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.

Zhiyong Wu, Yun Chen, Ben Kao, and Qun Liu. 2020. Perturbed masking: Parameter-free probing for analyzing and interpreting BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4166–4176, Online. Association for Computational Linguistics.

A Dependency Relations

Figure 8 shows the full reconstructability boost results for all dependency arc labels in our dataset.

B Detailed Decomposition Analysis

B.1 Creating Ablated Sequences

Fully contextualized See Section 2.2.

Static embedding (+position) We pass through the masked language model the n versions of the input sequence described above, each of which has a single token revealed, at the input layer only. Again, for each `[MASK]` token in the input sequence, we take the probability of the token in the same position in the original sequence as the reconstruction probability. This value corresponds to the probability of predicting the token in the original sequence given only the static lexical information of the source token and the positional information of the source and recon tokens.

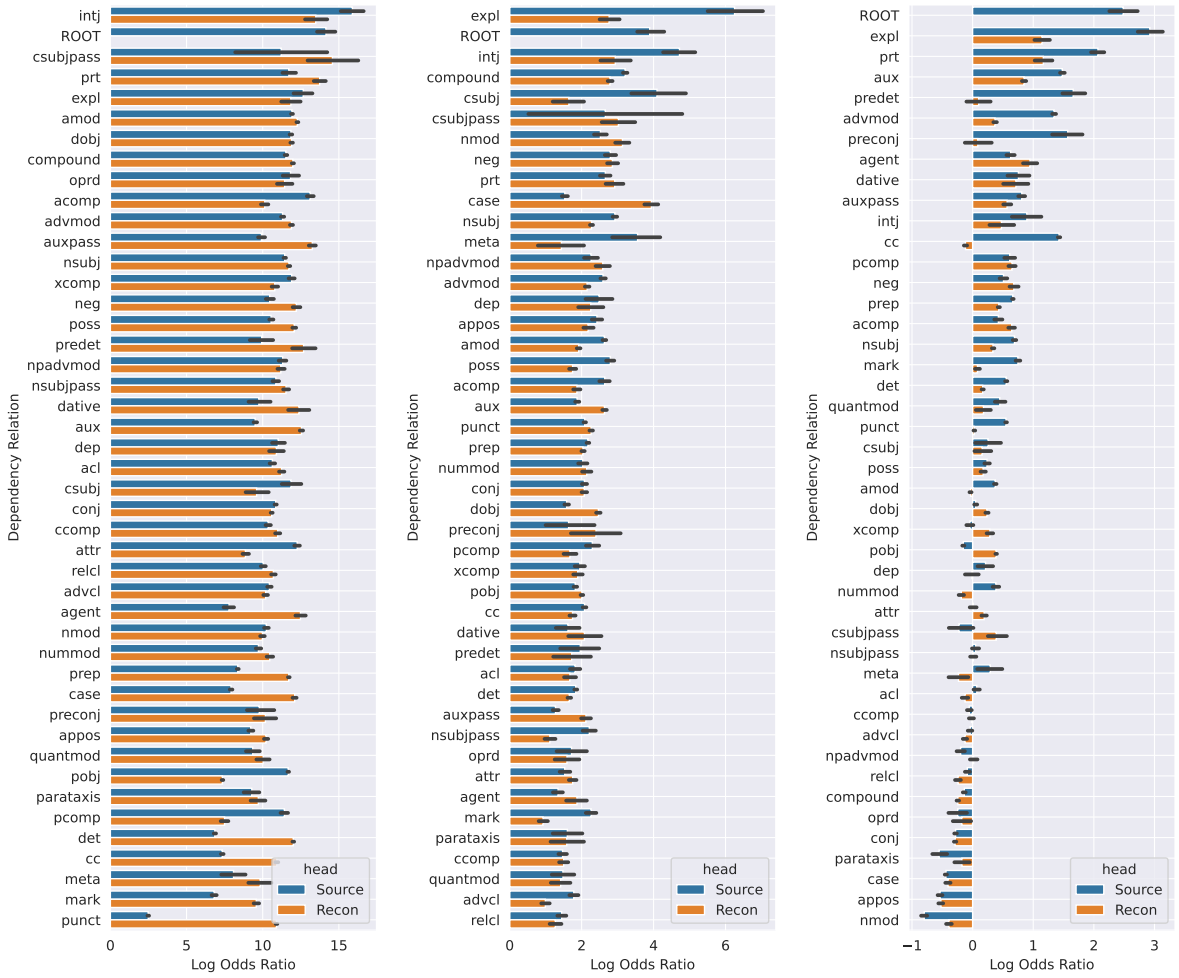
Static embedding (-position) We pass through the n single token-revealed versions of the input sequence as described above, but at the input layer, we do not add the positional embeddings. The reconstruction probability obtained, then, corresponds to the probability of predicting the token in the original sequence given only the static lexical information of the source token and no positional information of any of the tokens.

All mask (+position) We pass through a fully masked version of the input sequence that consists of the same number of `[MASK]` tokens and obtain the reconstruction probability of the tokens in the original sequence. Hence, in this scenario, there is no `source`. The value obtained through this input corresponds to the probability of predicting the token in the original sequence in the absence of any lexical information. Note that the model still has access to the positional embeddings of the `recon` token, which may still be weakly informative for token prediction.

All mask (-position) See Section 2.2, 'Lexical prior only baseline'.

B.2 Representations Compared

By comparing the reconstruction probabilities described above using Eq. 1, we can gauge the effect of the additional contextual information on performing masked language modeling. For example,



(a) BERT (b) RoBERTa (c) DistilBERT

Figure 8: Reconstructibility boost (log odds ratio with vs. without source) broken down by the depedency relation label between a SOURCE and RECON.

if we compare **Fully contextualized** and **Static embedding (+position)**, we can quantify the benefit of having the contextualization that happens through applying the model weights to the static representation of the input. If we compare **Static embedding (+position)** and **Static embedding (-position)**, we can quantify the benefit of positional embeddings (when given the same static lexical information). We make six different comparisons illustrated in Table 3, each comparison serving a different analytic role.

B.3 Further Discussion

We furthermore hypothesized the reconstruction boost from the availability of positional embeddings to be sensitive to the presence of a syntactic relation between SOURCE and RECON. This hypothesis is borne out in BERT and RoBERTA, but not in DistilBERT, suggesting that positional embeddings in DistilBERT are qualitatively different (Figure 9, left column).

C License and Terms for Use

License information for scientific artifacts used in this paper is as follows: MNLI (MIT License), BERT (Apache-2.0 License), RoBERTa (MIT License), and DistilBERT (Apache-2.0 License). Our own code follows the GPL-3.0 License. All of the publicly available artifacts are used in ways that comply with their licenses.

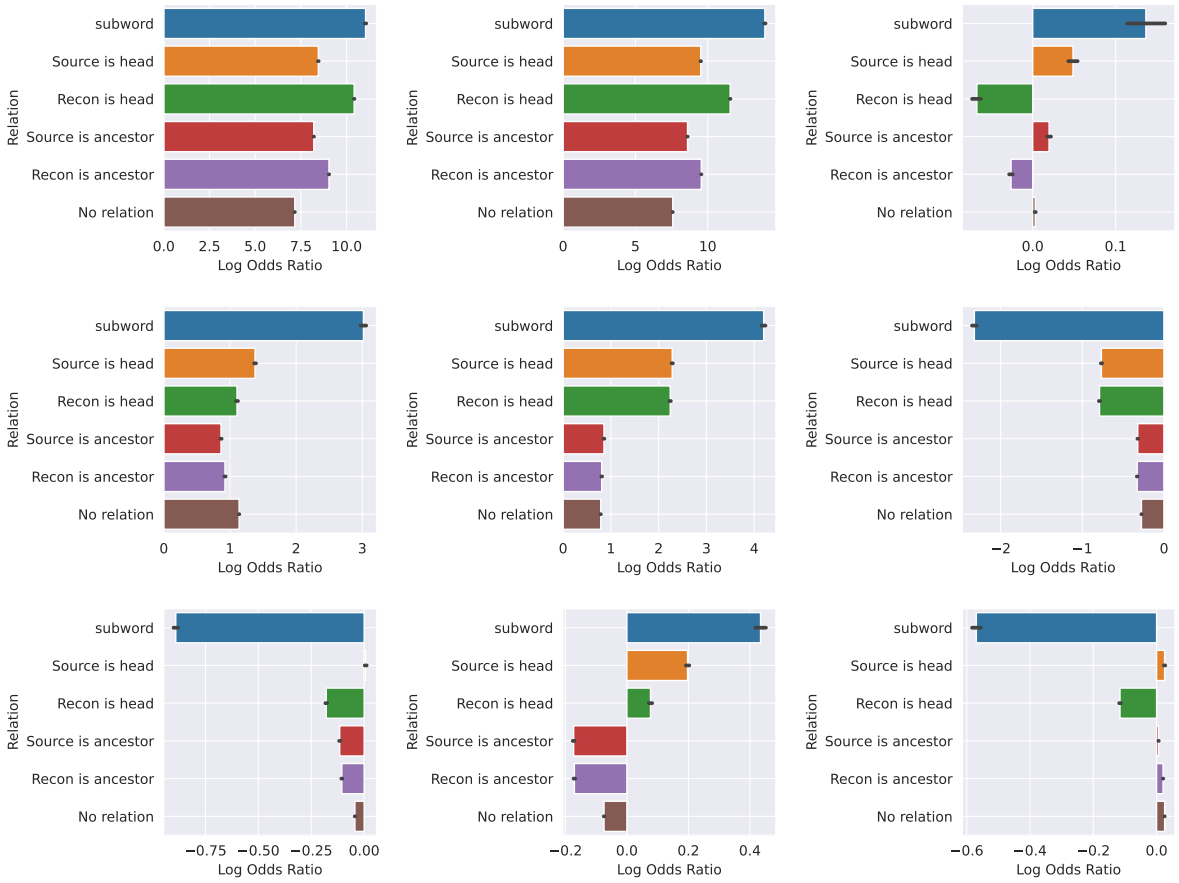
D Model and Implementation Details

The models that we used in this paper are all pretrained checkpoints from HuggingFace (Wolf et al., 2020). Specifically, they are: `bert-large-uncased` (340M parameters), `roberta-large` (355M parameters), and

Base	Augmented	What Base vs. Augmented can tell us
All mask (-position)	All mask (+position)	Reflects the effect of positional information in the absence of any lexical information other than the most general lexical priors of the model.
All mask (-position)	Static (-position)	Reflects the effect of static lexical information in the absence of positional information.
All mask (+position)	Static (+position)	Reflects the effect of static lexical information in the presence of positional information.
Static (-position)	Static (+position)	Reflects the effect of positional information in the presence of full lexical information.
Static (+position)	Fully contextualized	Reflects the effect of the contextualization through the layers of the model, beyond the input layer.
All mask (-position)	Fully contextualized	Comparison between the least and most contextualized reconstruction scenarios. Reflects the overall change induced by contextualization over the lexical priors of the model.

Table 3: The comparisons examined and the purpose of the comparisons. Eq. 1 is computed by taking the reconstruction probability of a given (token, input sequence) under **Base** as q and under **Augmented** as p.

`distilbert-base-uncased` (66M parameters). We inherited tokenization and any applicable hyperparameter settings from the specifications of the pretrained checkpoints. Computing reconstruction probabilities took around 3 CPU days for each model.



(a) All mask vs. All mask (-position) (b) Static vs. Static (-position) (c) Fully contextualized vs. Static

Figure 9: Relative reconstructibility (log odds ratio) for BERT (top), RoBERTa (middle), and DistilBERT (bottom).

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 5.1 and a separate limitations section

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Primarily evaluation work on syntactic relations between token representations—no particular risk scenario envisioned.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Our own code for analysis described in Section 2, pretrained models (Section 3.1), the MNLI dataset (Section 3.2).

☑ B1. Did you cite the creators of artifacts you used?
Analysis code is our own. Citations are provided in the relevant sections: pretrained models (Section 3.1), the MNLI dataset (Section 3.2).

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Appendix C

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Appendix C

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. We did not collect our own data.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Sections 3.2 and 3.3

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

C ☑ Did you run computational experiments?

Sections 3–5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix D

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Experimental setup is discussed throughout Sections 3–5 and in Appendix D.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Sections 4–5, Appendix A–B

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Appendix D

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.