

# Probing for Understanding of English Verb Classes and Alternations in Large Pre-trained Language Models

David K. Yi, James V. Bruno, Jiayu Han, Peter Zukerman, Shane Steinert-Threlkeld

Department of Linguistics, University of Washington

{davidyi6, jbruno, jyhan126, pzoek, shanest}@uw.edu

## Abstract

We investigate the extent to which verb alternation classes, as described by Levin (1993), are encoded in the embeddings of Large Pre-trained Language Models (PLMs) such as BERT, RoBERTa, ELECTRA, and DeBERTa using selectively constructed diagnostic classifiers for word and sentence-level prediction tasks. We follow and expand upon the experiments of Kann et al. (2019), which aim to probe whether static embeddings encode frame-selectional properties of verbs. At both the word and sentence level, we find that contextual embeddings from PLMs not only outperform non-contextual embeddings, but achieve astonishingly high accuracies on tasks across most alternation classes. Additionally, we find evidence that the middle-to-upper layers of PLMs achieve better performance on average than the lower layers across all probing tasks.

## 1 Introduction

We investigate the extent to which verb alternation classes are represented in word and sentence embeddings produced by Pre-trained Language Model (PLM) embeddings (Qiu et al., 2020). As first comprehensively cataloged by Levin (1993), verbs pattern together into classes according to the syntactic alternations in which they can and cannot participate. For example, (1) illustrates the *causative-inchoative* alternation. *Break* can be a transitive verb in which the subject of the sentence is the agent and the direct object is the theme, as in example (1a). It can also alternate with the form in (1b), in which the subject of the sentence is the theme and the agent is unexpressed. However, (2) demonstrates that *cut* cannot participate in the same alternation, despite its semantic similarity.

- (1) a. Janet broke the cup.
- b. The cup broke.
- (2) a. Margaret cut the bread.

- b. \*The bread cut.

(3) demonstrates an alternation of a different class – namely, the *spray-load* class, in which the theme and locative arguments can be syntactically realized as either direct objects or objects of the preposition. *Spray* can participate in the alternation, but as shown in (4), *pour* cannot.

- (3) a. Jack sprayed paint on the wall.
- b. Jack sprayed the wall with paint.
- (4) a. Tamara poured water into the bowl.
- b. \*Tamara poured the bowl with water.

The alternations in which a verb may participate is taken to be a lexical property of the verb (e.g. Pinker, 1989; Levin, 1993; Levin et al., 1995; Schafer, 2009). Moreover, we hypothesize that the alternations should be observable within large text corpora, and are therefore available during the pre-training procedure for PLMs such as BERT (Devlin et al., 2018). In contrast, ungrammatical examples such as (2b) and (4b) should be virtually absent from the training data. This leads us to hypothesize that PLM embeddings should encode whether particular verbs are allowed to participate in syntactic frames of various alternation classes. Our research questions are as follows:

1. Do PLM word-level contextual representations encode information about which syntactic frames an individual verb can participate in?
2. At the sentence level, do PLM embeddings encode the frame-selectional properties of their main verb?

Through our series of experiments, we find that PLM embeddings indeed encode information about verb alternation classes at both the word and sentence level. While performance is relatively consistent on the word-level task for the four PLMs

we analyze, we find that ELECTRA (Clark et al., 2020) significantly outperforms the other models for the sentence-level task. Furthermore, we find evidence suggesting that middle-to-upper layers encode more information about verb alternation classes since they consistently improve upon the lower layers across all tasks.

The rest of the paper is organized as follows: after a brief review of related literature in Section 2, we present datasets and models that are relevant to our experiment in Sections 3 and 4. We then present two experiments to answer our research questions in Sections 5 and 7. Section 6 presents an additional *control task* (Hewitt and Liang, 2019) to test whether our linear probes are selective for the given tasks. Finally, we offer a discussion in Section 8 and overall conclusions in Section 9.

## 2 Related work

Our work follows Kann et al. (2019), who attempt to predict verb class membership and sentence grammaticality judgments on the basis of GloVe embeddings (Pennington et al., 2014) and embeddings derived from the 100M-token British National Corpus with a single-directional LSTM (Warstadt et al., 2019). For the sentence-level task, they further process the input embeddings using a sentence encoder trained on a “real/fake” sentence classification task. Varying multi-layer perceptron (MLP) architectures are used for the classification step. Because their primary research focus has to do with how neural language models inform learnability (in the sense of human language acquisition), they intentionally use smaller language models derived from “an amount of data similar to what humans are exposed to during language acquisition” and avoid models trained on “several orders of magnitude more data than humans see in a lifetime” (p. 291).

As described in Section 5, we depart and build upon Kann et al. 2019 by examining the embedding representations of PLMs instead of static embeddings. We then use an intentionally simple and selective linear diagnostic classifier to probe the representations, as our research questions focuses on the PLM embeddings themselves. We note that Kann et al. (2019) achieves only modest performance in prediction accuracy and MCC, and only for a limited number of verb classes. While this is a valuable result for their research goals, our hypothesis is that PLMs will achieve better performance

due to a combination of their contextual representations, complex architectures, and larger training corpora.

To our knowledge, attempting to predict verb alternation class membership along the lines of Levin 1993 from PLM representations is novel. However, two very closely related lines of work include the experiments of Warstadt and Bowman (2019), which respectively evaluate the performance of various PLMs on the CoLA (Warstadt et al., 2019) and BliMP (Warstadt et al., 2020) benchmarks, which include acceptability judgment examples from a wide variety of linguistic phenomena (including verb argument structures). We distinguish our experiments from these papers in two major ways. First, we attempt to directly probe the linguistic knowledge of individual PLM embedding layers with a classification probe instead of specifically finetuning the models to a specific task. Second, we limit our focus to verb alternation classes and present detailed analysis about patterns and trends across different alternations and their corresponding syntactic frames.

## 3 Data

In our experiments, we use two dataset created by Kann et al. (2019). One is the **Lexical Verb-frame Alternations** dataset (LaVA), which is based on the verbs and alternation classes defined in Levin (1993). It contains a mapping of 516 verbs to 5 alternation classes, which are further subdivided into two syntactic frames for each alternation. The broad categories of the alternation classes are: *Spray-Load*, *Causative-Inchoative*, *Dative*, *There-insertion*, and *Understood-object*. Table 1<sup>1</sup> provides the class distributions for each syntactic frame. **Frames and Alternations of Verbs** (FAVA), the other dataset, is a corpus of 9413 semi-automatically generated sentences formed from the verbs in LaVA along with human grammaticality judgments. The sentences in FAVA are categorized according to the relevant alternation class, and are separated into train, development, and test sets by the authors for each category.

---

<sup>1</sup>A similar table appears in Kann et al. (2019), but we present it again here because of discrepancies that we found in the distribution counts. Notably, it appears that the authors flipped the positive and negative counts for the *there-Insertion* and *Understood-Object* alternation classes which carries over to their results.

LEVIN-CLASS	CAUS-INCH		DATIVE		SPRAY-LOAD		there-INSERTION		UNDERSTOOD-OBJECT	
	Inch.	Caus.	Prep.	2-Obj	with	loc.	no-there	there	Refl	No-Refl
Positive	73	124	65	74	101	86	149	50	84	11
Negative	144	0	377	442	242	257	0	192	419	503
Total	217	124	442	516	343	343	149	242	503	514

Table 1: An updated overview of the LaVA dataset based on verb membership class distributions for each syntactic frame. “Positive” refers to the number of verbs that can participate in the specified syntactic frame, while “Negative” refers to the number of verbs that cannot participate.

## 4 Models

In addition to BERT, we perform experiments on several recent Transformer-based PLMs including RoBERTa (Liu et al., 2019), DeBERTa (He et al., 2021), and ELECTRA (Clark et al., 2020) which vary from BERT in a few ways including modifications to BERT’s tokenization and pre-training procedure and the size of their training corpus. To make comparisons between each model fair, we use the base architectures for each model which have 12 layers, 12 attention heads, and a hidden layer size of 768.<sup>2</sup>

### 4.1 Model differences

For pre-training, BERT uses standard Masked Language Modeling (MLM) wherein tokens from a given input sequence are masked at random and the model attempts to recover the masked tokens from the unmasked tokens and Next Sentence Prediction (NSP), in which the model tries to predict whether one sentence follows another in a given text sequence. The other PLMs drop NSP from their pre-training procedure but make other significant changes to the architecture and the MLM approach. RoBERTa introduces “dynamic” masking, in which different tokens are masked across different training epochs (as opposed to the same training mask being used across epochs). DeBERTa uses a “disentangled attention mechanism” which computes attention weights using distinctly encoded position and context vectors, and also moves absolute position encodings from the input layer to the second-to-last layer. Lastly, instead of randomly masking input tokens, ELECTRA strategically replaces tokens with plausible alternatives using a trained generator network, and separately trains a discriminative model which aims to predict whether each token in an input sequence was replaced by a generator sample.

<sup>2</sup>All further references to these models refer to their *base* architectures.

### 4.2 Training Data

In addition to variations in the pre-training methods, the models are also trained on different datasets. BERT and ELECTRA are both trained on the English Wikipedia Dump and BookCorpus (Zhu et al., 2015). DeBERTa is additionally trained on CC-Stories (Trinh and Le, 2018) and OpenWebText (Gokaslan and Cohen, 2019). Finally, RoBERTa is pretrained on all of the aforementioned datasets as well as the CC-News corpus (Mackenzie et al., 2020).

## 5 Experiment 1: Frame Membership from Word Embeddings

### 5.1 Method

In order to answer the first question: “Do PLM token-level representations encode information about which syntactic frames an individual verb can participate in?”, we build a diagnostic classifier for each syntactic frame which takes a verb’s layer embedding representation as input. For example, to probe the *Spray-Load* alternation, we build two binary classifiers: one that predicts whether a verb can participate in the “locative” frame and one that predicts whether a verb can participate in the “with” frame.

Furthermore, we build a separate classifier for each model layer based on the embedding representations from that particular layer. For the token-embedding layer, the verb embedding is formed by averaging the pretrained token embeddings that correspond to a particular verb. For layers 1–12, the verb embedding is formed by incorporating contextual information from the sentences in FAVA. Specifically, for each verb, we pass the grammatical sentences from FAVA that contain the verb as input to the PLM and average over the token embeddings corresponding to the verb. We choose to only include grammatical examples in the construction of the word-level contextual embeddings since we hypothesize that they represent sentences

	MCC					Accuracy				
	Ref.	BERT	DeBERTa	ELECTRA	RoBERTa	Ref.	BERT	DeBERTa	ELECTRA	RoBERTa
CAUSATIVE-INCHOATIVE										
Inchoative	0.555	0.948 [11]	0.969 [11]	0.959 [5]	0.969 [7]	0.855	0.977	0.986	0.982	0.986
Causative *	0.000	0.000	0.000	0.000	0.000	1.000	1.000	1.000	1.000	1.000
DATIVE										
Preposition	0.320	<b>0.954 [8]</b>	0.937 [12]	0.945 [11]	0.928 [9]	0.850	<b>0.989</b>	0.984	0.986	0.982
Double-Object	0.482	0.976 [10]	0.968 [10]	0.976 [12]	0.936 [9]	0.853	0.994	0.992	0.994	0.984
SPRAY-LOAD										
With	0.645	0.972 [10]	0.972 [12]	<b>0.979 [8]</b>	0.930 [10]	0.839	0.988	0.988	<b>0.991</b>	0.971
Locative	0.253	<b>0.969 [10]</b>	0.961 [12]	0.961 [9]	0.953 [11]	0.734	<b>0.989</b>	0.985	0.985	0.983
THERE										
No-There *	0.000	0.000	0.000	0.000	0.000	1.000	1.000	1.000	1.000	1.000
There	0.459	1.000 [9]	0.987 [7]	1.000 [10]	0.962 [10]	0.858	1.000	1.000	1.000	0.988
UNDERSTOOD OBJECT										
Refl	0.000	0.868 [6]	0.869 [12]	0.860 [5]	<b>0.884 [11]</b>	0.732	0.964	0.964	0.962	<b>0.968</b>
Non-Refl	0.219	0.850 [7]	0.850 [10]	<b>0.855 [11]</b>	0.794 [8]	0.976	0.994	0.994	0.994	0.992

Table 2: Results from Word-Level experiments with static embeddings. Reference MCC is from Kann et al. (2019)’s CoLA word-level experiments. The \* symbol indicates syntactic frames which only have positive examples, which trivially achieve 100% accuracy and 0 MCC (see Footnote 1). The best performing model for a given frame is denoted in **bold** (ties are not bolded), and the best performing layer for each model is denoted in brackets ‘[]’.

in the actual training corpora of the PLMs more accurately than the ungrammatical examples.<sup>3</sup> We then average over the verb representations for all input sentences in each layer to form the “layer-embedding” for the verb.

We choose a Logistic Regression classifier without regularization as our diagnostic probe as implemented in `scikit-learn` (Buitinck et al., 2013) and show that it is sufficiently *selective* in Section 6. Following Kann et al. (2019), we use stratified k-fold cross-validation to split the verbs into 4 equally-sized folds: 3 of which are chosen to be the training set and the remaining fold chosen to be the test set.

Also following Kann et al. (2019), we report Matthews correlation coefficient (MCC) (Matthews, 1975) in addition to accuracy for model evaluation. MCC is better suited to data such as ours, in which there is an extreme majority class bias for all syntactic frames.<sup>4</sup>

<sup>3</sup>A potential issue with constructing the embeddings in this manner is that the classifier may simply “memorize” whether there is a corresponding grammatical example for each verb in FAVA to trivially determine frame membership. However, we included ungrammatical examples as well in preliminary experiments and found negligible differences from our final results.

<sup>4</sup>All code and data needed to replicate our analysis can be found at [https://github.com/kvah/analyzing\\_verb\\_alternations\\_plms](https://github.com/kvah/analyzing_verb_alternations_plms)

## 5.2 Results

In Figure 1, we present the layer-by-layer performance of each PLM and in Table 2, we report a comparison between the best-performing layer for each PLM alongside the performance of the “CoLA-style” reference embeddings from Kann et al. (2019). Overall, we find that the contextual PLM embeddings dramatically outperform the reference embeddings in terms of both MCC and accuracy.

Surprisingly, the PLMs perform well even for the more challenging frames; for the “locative” frame, BERT achieves 0.969 MCC compared to 0.253 when using the reference embeddings, and for the “non-reflexive” frame, ELECTRA achieves 0.855 MCC compared to 0.219 when using the reference embeddings. Furthermore, we observe consistent patterns in performance across different layers of each PLM. As shown in Figure 2, the lower layers achieve low-to-moderate correlation on average while the middle-to-upper layers consistently achieve strong correlation.

## 6 Control Task

A control task as described by (Hewitt and Liang, 2019) aims to combat the *Probe Confounder Problem*, which highlights the issue of supervised probe classifiers “learning” a linguistic task by combining signals in the data that are irrelevant to the linguistic property of interest. In the context of our first experiment, a confounding probe would

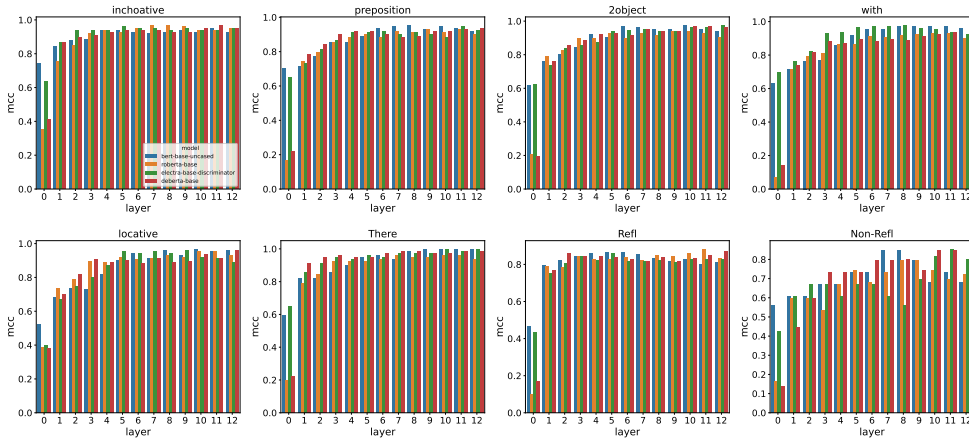


Figure 1: MCC for each model layer across all syntactic frames on LAVA.

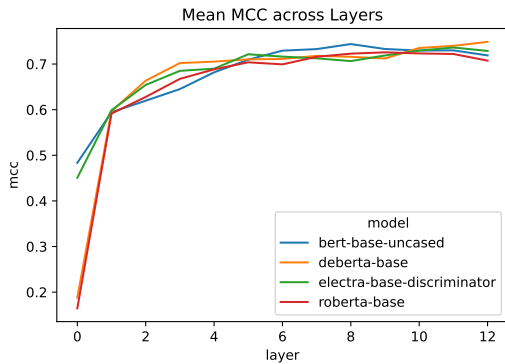


Figure 2: Mean MCC for each model layer across all syntactic frames on LAVA.

be problematic since it suggests that good model performance may be attributed to arbitrary signals picked up by the probe, as opposed to the PLM embeddings actually containing linguistic information about the syntactic frames. To mitigate the Probe Confounder Problem, we implement an example control task for the *Spray-Load* “with” syntactic frame for BERT.

For each verb  $v_i$  in LaVA with a binary label  $y_i$  denoting whether  $v_i$  can participate in the syntax frame SL-WITH, we independently sample a control behavior  $C(v)$  by randomly assigning a binary “membership” value to  $v_i$  based on the empirical membership distribution of verbs that participate in the SL-WITH syntax frame. The control task is the function that maps each verb,  $v_i$ , to the label specified by the control behavior  $C(V_i)$ :

$$f_{control}(v_i) = C(v_i)$$

Following the experiment design of Hewitt and

Liang (2019), we compare the *selectivity* of a linear probe, a Multi-layer Perceptron with 1-hidden layer (MLP-1), and an MLP with 2-hidden layers (MLP-2) where the *selectivity* of a model is defined by the difference between its accuracy on the real task (i.e. predicting verb membership for the SL-WITH frame) and the control task. In addition, we explore several “complexity control” methods including limitation of feature dimensionality, reducing the number of training examples, and increasing regularization.

## 6.1 Complexity Hyperparameters

In this section, we describe the complexity control methods in more detail and enumerate the hyperparameters that we tried for each method. The control parameters were chosen based on the three most effective methods from the experiments of Hewitt and Liang (2019). To isolate the effect of each control method, we only change one of the complexity parameters in each experiment.

### 6.1.1 Limiting Dimensionality

For the Logistic Regression model, we reduce the dimensionality of the feature embeddings by performing a Truncated Singular Value Decomposition and limiting the output matrix to rank  $k$ . For the MLP models, we simply limit the size of the hidden layer(s) to  $k$ .

Considering the input BERT embeddings which have 768 dimensions, we limit  $k$  to the following values:  $\{20, 100, 300, 500\}$ .



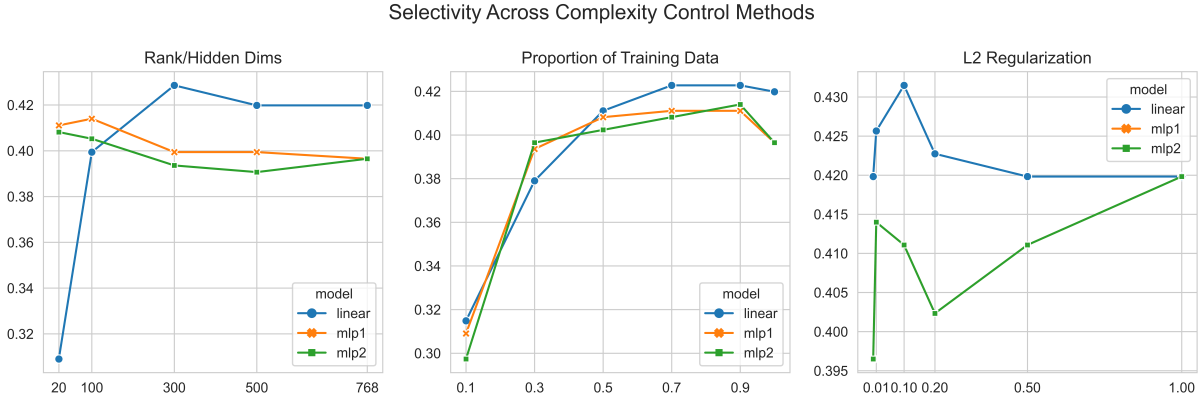


Figure 3: Linguistic Task selectivities for the three complexity control methods.

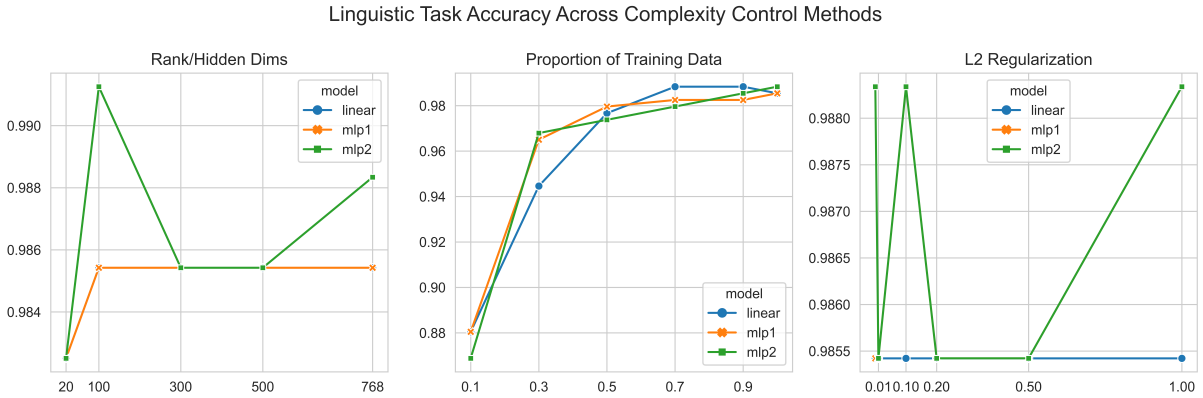


Figure 4: Linguistic Task accuracies for the three complexity control methods.

### 6.1.2 Reducing Proportion of Training Data

Because LaVA is not split into train and test sets, we use 4-fold cross validation as done in Kann et al. (2019) with 3 training folds and one test fold for evaluation on the control task. As an additional constraint, we reduce the number of training samples in each training fold by randomly sampling a proportion  $p$  of the samples and discarding the rest.

Although Zhang and Bowman (2018) recommend training on 1%, 10% and, 100% of the training data, our training data is relatively small and imbalanced (71% of the train set verbs do not participate in the SL-WITH frame). Hence, we experiment with larger values of  $p$ :  $\{0.1, 0.3, 0.5, 0.7, 0.9\}$

### 6.1.3 $L_2$ Regularization

For both the linear and MLP models, we add  $L_2$  regularization with the following strength values:  $\{0.01, 0.1, 0.2, 0.5, 1\}$

## 6.2 Results

Figure 3 shows the high-level trends across experiment configurations for model selectivity. We observe that the linear model with default parameters ( $k = 768, p = 1, L_2 = 0$ ) outperforms both the

MLP-1 and MLP-2 model in selectivity (0.429 v.s. 0.397) with no significant decrease in linguistic task accuracy (0.985 for the linear and MLP-1 models v.s. 0.988 for the MLP-2 model).

Looking at the effect of complexity control methods on model accuracy in Figure 4, we find that limiting dimensionality and  $L_2$  regularization has little impact across all configurations, with the worst model (linear:  $k = 20$ ) achieving an accuracy of 0.983 and the best model (MLP-2:  $k = 100$ ) achieving only a slightly higher accuracy of 0.991. On the other hand, reducing the proportion of data in each training fold appears to have significant impact on model performance. For the linear model, there is a huge discrepancy in accuracy between training on 10% of the data (0.869) and the full training set (0.988). A nearly identical pattern can be observed for both of the MLP models as well.

Comparing selectivity, the linear models outperform both MLPs across all complexity control methods. For dimensionality control, we see a lower selectivity in the linear model for lower values of  $k$  ( $k = 20, 100$ ) but the best linear model ( $k = 300$ ) achieved a higher selectivity (0.429)

than the best MLP model (0.414). Similarly, the best performing configuration for reduced training samples and  $L_2$  regularization are linear models with  $p = 0.9$  (0.423) and  $L_2 = 0.1$  (0.431) respectively.

We arrive at two major conclusions from the control task experiments. The first is that a linear probe is a good choice for our linguistic task since it achieves higher selectivity than the MLP models without substantial loss in model accuracy across a wide range of complexity control methods. The second is that limiting dimensionality, reducing training samples, and  $L_2$  regularization are all effective methods for increasing model selectivity for both the linear and MLP models. However, the best configurations are not significantly better ( $> 0.01$  improvement in selectivity) than the default linear model so we did not make any modifications to our classification probe. As we only performed these experiments for BERT and the SL-WITH syntactic frame specifically, a great avenue for future work is to test whether our results extend to other PLMs and syntactic frames.

## 7 Experiment 2: Grammar Judgments from Sentence-embeddings

### 7.1 Method

In the second experiment, we investigate the extent to which PLMs encode frame-selectional properties of their main verb. For each PLM and embedding layer, we fit a binary Logistic Regression classifier on the FAVA training set for a given alternation class which predicts whether a given sentence is grammatical. We ignore the held out development set because the probe hyperparameters do not need to be tuned and directly evaluate each model on the test set. The whole process can be described by the following equation:

$$c_{s_i} = f(\mathbf{W}s_i + \mathbf{b})$$

where  $s_i$  refers to the embedding of the whole sentence for layer  $i$  (by averaging all  $i$  layer’s hidden states of words in the sentence  $s$ ),  $f$  refers to the logistic regression classifier,  $\mathbf{W}$  and  $\mathbf{b}$  are the parameters of  $f$ , and  $c_{s_i}$  is a binary value corresponding to whether the sentence is grammatical. We then extract the best performing layer for each model and compare the results with the reference acceptability judgment model proposed by Kann et al. (2019).

### 7.2 Results

The MCC and accuracy scores for each model and layer are shown in Figure 5. From the figure, we can see that there is significant variation in layer performance aside from the *Understood-object* alternation. Generally, we observe a trend in which performance increases substantially from the lower (1-4) layers to the middle layers (5-9), with some models, most notably ELECTRA, continuing to improve through the upper layers (10-12). This can be seen more clearly in Figure 6, which shows the mean layer performance across each category. Furthermore, ELECTRA achieves the best MCC on 5 of the 6 categories: *Combined* (0.818), *Inchoative* (0.864), *Spray\_Load* (0.830), *There* (0.828), and *Understood-Object* (0.869). The outlier frame is RoBERTa, which achieves the best MCC (0.802) on the *Dative* frame.

Table 3 provides a comparison between the best performing layer from each PLM and the reference embeddings from Kann et al. (2019) for each alternation class. As defined by Kann et al. (2019) an MCC value between 0.5 and 0.7 demonstrates a moderate correlation between predicted and true labels while an MCC greater than 0.7 implies strong correlation. From the table, we see all models are able to obtain strong correlation for the *Understood-Object* alternation, the *There* frame, and the *Causative-Inchoative* frame. In contrast, BERT and RoBERTa are only able to achieve moderate correlation on the *Spray-Load* frame, while all models except RoBERTa only achieve moderate correlation on the *Dative* alternation. Consistent with the CoLA-style embeddings, we find that the PLMs achieve the best performance on average for sentences from the *Understood-Object* alternation class. This is surprising since frames from the *Understood-Object* alternation were the hardest to predict for the word-level task for both the CoLA-style embeddings and the PLMs. Nevertheless, all PLM outperform the reference model across all alternation categories for the sentence acceptability judgment task.

## 8 Discussion

On the word level prediction task, all PLMs achieve strong correlation ( $> 0.7$  MCC) across all syntactic frames with the strongest performance in the “there” frame (1.00, achieved by BERT and ELECTRA) and the weakest performance on the “non-reflexive” frame (0.794, achieved by RoBERTa). When look-

	COMBINED	CAUSATIVE-INCHOATIVE	DATIVE	SPRAY-LOAD	THERE	UNDERSTOOD
MCC						
REF.	0.290	0.603	0.413	0.323	0.528	0.753
BERT	0.642 (10)	0.760 (8)	0.678 (6)	0.625 (10)	0.716 (10)	0.842 (9)
DeBERTa	0.653 (9)	0.776 (5)	0.633 (8)	0.704 (12)	0.744 (6)	0.826 (1)
ELECTRA	<b>0.818 (11)</b>	<b>0.864 (11)</b>	0.670 (8)	<b>0.830 (12)</b>	<b>0.828 (12)</b>	<b>0.869 (11)</b>
RoBERTa	0.496 (8)	0.725 (8)	<b>0.802 (2)</b>	0.470 (5)	0.725 (11)	0.793 (1)
ACCURACY						
REF.	0.646	0.854	0.760	0.662	0.729	0.874
BERT	0.840 (10)	0.920 (8)	0.880 (6)	0.820 (10)	0.890 (10)	0.921 (9)
DeBERTa	0.847 (9)	0.924 (5)	0.902 (8)	0.858 (12)	0.912 (6)	0.909 (1)
ELECTRA	<b>0.920 (11)</b>	<b>0.954 (11)</b>	0.897 (8)	<b>0.918 (12)</b>	<b>0.937 (12)</b>	<b>0.934 (11)</b>
RoBERTa	0.787 (8)	0.909 (8)	<b>0.944 (2)</b>	0.747 (5)	0.899 (11)	0.893 (1)

Table 3: Results from Sentence-Level experiments. REF refers to the reference probing model in (Kann et al., 2019). Bolded values show the best result for each alternation class. ‘()’ indicates the best performing layer for each model.

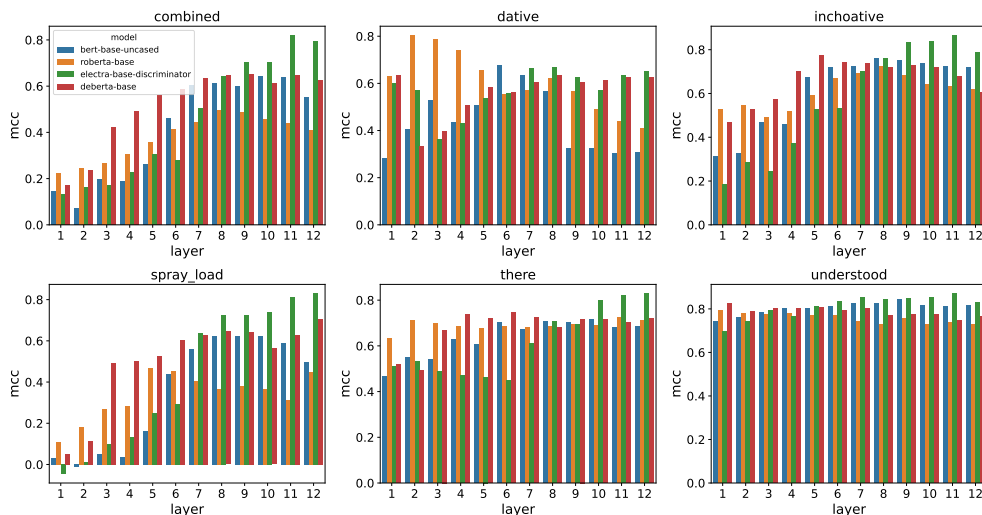


Figure 5: Layer-by-layer MCC score for each alternation class on FAVA.

ing at accuracies, each model is able to predict whether a verb belongs to a particular syntactic frame with excellent accuracy ( $> 0.95$  across all alternation frames). Moreover, looking at Table 2, we see that the middle (5-9) and upper (10-12) layers consistently achieve the highest MCC, which is reinforced by the trend shown in Figure 2.

For the sentence-level experiments, we see a similar outcome wherein the upper-middle PLM layers achieve the best performance on average. However, we observe that there is much more variation in performance between each PLM. ELECTRA and BERT are relatively consistent, since their best performing layer for all alternation classes either come from the middle or upper layers. In contrast, the lower layers of RoBERTa achieve the best performance on the *Dative* alternation, and both RoBERTa and DeBERTa achieve the best perfor-

mance on the *Understood-Object* alternation from the first layer. These anomalies can potentially be explained by the claim that different alternation classes require different types of linguistic knowledge (i.e. syntactic v.s. semantic) which are encoded in different PLM layers. However, the consistently strong performance of the upper layers for BERT and ELECTRA across all alternation classes provides counter evidence against the claim.

ELECTRA is the best performing model overall on the sentence-level acceptability task, achieving the best MCC and accuracy on four of the five alternation classes (all except *Dative*). Unsurprisingly, ELECTRA also excels on the combined dataset compared to the other models (0.165 MCC over the second-best performing model, DeBERTa). While it is difficult to attribute the model’s success to a specific property, one hypothesis is that its gener-



ator/discriminator architecture closely resembles the FAVA task of identifying acceptable sentences from linguistic minimal pairs. This idea is reinforced by the authors as well, who note that the model’s relatively strong performance on CoLA potentially stems from the fact that the acceptability judgment task of CoLA “closely matches ELECTRA’s pre-training task of identifying fake tokens” (Clark et al., 2020, p.15).

While we are optimistic about our results, there are several limitations to our experiments. First, we only analyze five different alternation classes which is a small subset of the 83 classes presented in Levin (1993). In addition – although our control task ensures that our classifier probe is relatively *selective* for the first experiment and BERT, it may not necessarily generalize well to the second experiment, other syntactic frames, and other models. In the future, we hope to expand our selectivity experiments to a wider array of syntactic frames and models.

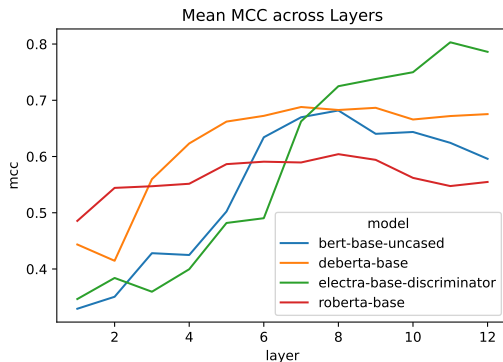


Figure 6: Mean layer MCC score across all alternation classes on FAVA.

## 9 Conclusion and Future Work

Overall, our results support the hypothesis that PLM contextual embeddings encode linguistic information about verb alternation classes at both the word and sentence level. For the frame-selectional verb classification task, all PLMs achieve significant improvement upon the reference CoLA-style embeddings from Kann et al. (2019), especially for frames in which the CoLA-style embeddings obtain weak correlation (i.e. “locative”, “reflexive”, and “non-reflexive”). Also, it is clear that model performance tends to improve from lower to upper layers, which can be seen the most easily from the mean performance across layer figures. For the sentence acceptability task, we arrive at similar conclusions,

albeit with greater distinction in results between different models and layers. While there are numerous factors that may be responsible for the improved performance from PLMs, we hypothesize that the improvement can largely be attributed to the attention-based encodings of transformer models since we only saw modest improvements in performance from the reference embeddings when using the bottom “static” layers for each PLM.

In terms of future work, there are several interesting avenues that we hope to explore. From the data perspective, it would certainly be worthwhile to test whether our insights and conclusions extends to the dozens of alternations described in Levin (1993) that are not present in the LAVA and FaVA datasets. There are also several interesting adaptations that can be made to our experiment methodology. For example, instead of just analyzing the base architecture for each PLM, we could also analyze *small* and *large* variants to directly evaluate the effect of scaling training data and model size within the same model. Moreover, while we attempt to control the *Probe Confounder Problem* by building a selective probe, there is no guarantee that the classifier probes do not pick up on arbitrary signals in the training data that lead to non-meaningful improvements in performance. Two promising alternative approaches that mitigate this risk include unsupervised evaluation of minimal pairs as shown in Warstadt et al. (2020) and “amnesic probing”, which tests whether a property that can be extracted from a probe is actually relevant to task importance (Elazar et al., 2021).

## References

- Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.
- Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. **ELECTRA: Pre-training text encoders as discriminators rather than generators**. In *ICLR*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. **BERT: pre-training of deep bidirectional transformers for language understanding**. *CoRR*, abs/1810.04805.

- 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.
- Aaron Gokaslan and Vanya Cohen. 2019. [Openwebtext corpus](#).
- Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. [Deberta: Decoding-enhanced bert with disentangled attention](#). In *International Conference on Learning Representations*.
- John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. 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 2733–2743.
- Katharina Kann, Alex Warstadt, Adina Williams, and Samuel R. Bowman. 2019. [Verb argument structure alternations in word and sentence embeddings](#). In *Proceedings of the Society for Computation in Linguistics (SCiL) 2019*, pages 287–297.
- Beth Levin. 1993. *English verb classes and alternations: A preliminary investigation*. University of Chicago press.
- Beth Levin, Malka Rappaport Hovav, and Samuel Jay Keyser. 1995. *Unaccusativity: At the syntax-lexical semantics interface*, volume 26. MIT press.
- 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 preprint arXiv:1907.11692*.
- Joel Mackenzie, Rodger Benham, Matthias Petri, Johanne R. Trippas, J. Shane Culpepper, and Alistair Moffat. 2020. [Cc-news-en: A large english news corpus](#). *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*.
- Brian W Matthews. 1975. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, 405(2):442–451.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. [Glove: Global vectors for word representation](#). In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.
- Steven Pinker. 1989. *Learnability and cognition: The acquisition of argument structure*. Cambridge, MA: MIT Press.
- XiPeng Qiu, TianXiang Sun, YiGe Xu, YunFan Shao, Ning Dai, and XuanJing Huang. 2020. [Pre-trained models for natural language processing: A survey](#). *Science China Technological Sciences*, 63(10):1872–1897.
- Florian Schafer. 2009. The causative alternation. *Language and linguistics compass*, 3(2):641–681.
- Trieu H. Trinh and Quoc V. Le. 2018. [A simple method for commonsense reasoning](#).
- Alex Warstadt and Samuel R. Bowman. 2019. [Linguistic analysis of pretrained sentence encoders with acceptability judgments](#).
- Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. [BLiMP: A benchmark of linguistic minimal pairs for English](#). In *Proceedings of the Society for Computation in Linguistics 2020*, pages 409–410, New York, New York. Association for Computational Linguistics.
- Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.
- Kelly Zhang and Samuel Bowman. 2018. [Language modeling teaches you more than translation does: Lessons learned through auxiliary syntactic task analysis](#). In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 359–361, Brussels, Belgium. Association for Computational Linguistics.
- Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. [Aligning books and movies: Towards story-like visual explanations by watching movies and reading books](#). In *The IEEE International Conference on Computer Vision (ICCV)*.

## A Complexity Control Results

	Dimensions ( $k$ )	Training Prop. ( $p$ )	$L_2$ Reg.	Accuracy	Selectivity
DEFAULT PARAMS					
Linear	768	1.0	0.0	0.985	<b>0.420</b>
MLP-1	768	1.0	0.0	0.985	0.397
MLP-2	768	1.0	0.0	0.988	0.397
LIMITING DIMENSIONS					
Linear	300	1.0	0.0	0.985	<b>0.429</b>
MLP-1	100	1.0	0.0	0.985	0.414
MLP-2	20	1.0	0.0	0.983	0.408
REDUCING TRAINING SAMPLES					
Linear	768	0.9	0.0	0.988	<b>0.423</b>
MLP-1	768	0.9	0.0	0.983	0.411
MLP-2	768	0.9	0.0	0.985	0.414
$L_2$ REGULARIZATION					
Linear	768	1.0	0.1	0.985	<b>0.431</b>
MLP-1	768	1.0	1.0	0.988	0.420
MLP-2	768	1.0	1.0	0.988	0.420

Table A1: Results from the Complexity Control Experiments. For each experiment, only the best performing configuration for each model is reported.