Dive into the Chasm: Probing the Gap between In- and Cross-Topic Generalization

Anonymous ACL submission

Abstract

Pre-trained language models (PLMs) excel for In-Topic setups where training and evaluation data originate from the same topics. Simultaneously, they struggle with Cross-Topic setups where we withhold instances from distinct topics for evaluations. In this paper, we aim to understand better how and why such generalization gaps emerge by probing various PLMs for different aspects. We show for the first time that these generalization gaps and the fragility of token-level interventions notably vary across PLMs. Further, by evaluating large language models (LLMs), we show how our analysis scales to bigger models. Overall, we observed diverse pre-training objectives and architectural regularization contribute to more robust PLMs and mitigate generalization gaps. Our research attributes to a better understanding of PLMs, selecting appropriate ones, or building more robust ones. [1]

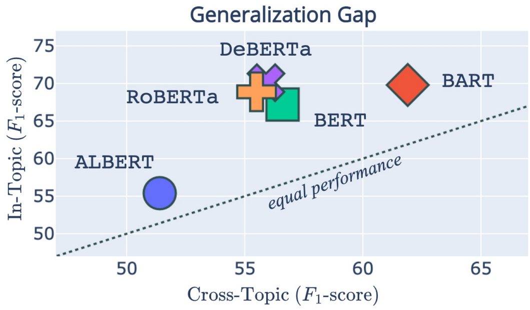

Figure 1: Generalization gap of fine-tuning PLMs on argumentative *stance detection* (Stab et al., 2018) in the In- or Cross-Topic evaluation setup. The dashed line marks the ideal case of equal performance.

1 Introduction

Fine-tuning is widely used to impart pre-trained language models (PLMs) (Devlin et al., 2019; Liu et al., 2019; He et al., 2021; Radford et al., 2019) new tasks and results on remarkable performance gains for general NLP - including GLUE (Wang et al., 2018) or SuperGLUE (Wang et al., 2019). However, such benchmarks are not well aligned to real-world applications where data is limited or unavailable. At the same time, PLMs may not meet expectations when we expect heavy disparities between training and testing data, like Cross-Topic evaluation (Sapkota et al., 2014; Stab et al., 2018; Ren et al., 2021). As a result, apparent generalization gaps exist between the commonly used In-Topic and the more realistic Cross-Topic evaluation setup. These gaps primarily arise when training and testing data originate from the same topics and cover the same vocabulary (In-Topic) or from different topics (Cross-Topic). For Cross-Topic, we see topic-specific tokens encapsulating the semantic distinctions between topics and contributing to distribution shifts. Consequently, it is imperative that PLMs effectively generalize learned tasks across such shifts, particularly for Cross-Topic.

Exemplary, we illustrate in Figure 1 generalization gaps when fine-tuning on the *UKP ArgMin* dataset (Stab et al., 2018) for In- and Cross-Topic. This Argument Mining dataset annotates arguments as either in favor, against, or neutral towards one of eight topics like *Gun Control*. Although we anticipate a better performance of PLMs for In-Topic, we make a crucial observation that In- vs. Cross-Topic performance differences vary considerably across PLMs - like BART performing similarly for In- but outperforming the others for Cross-Topic. As a result, we can not generalize findings or draw practical conclusions from one setup to another - such as choosing a model for new data.

The analysis and comparison of In- vs. Cross-Topic generalization gaps are crucial to building more robust models but remain understudied in the current literature. Mostly general behaviors of PLMs (Belinkov et al., 2017; Peters et al., 2018) are studied, while little research has been done

[1] We provide data and code anonymized online.

on generalization (Aghazadeh et al., 2022; Zhu et al., 2022). To the best of our knowledge, we propose for the first time an in-depth analysis of the In- and Cross-Topic generalization gap across various PLMs (§ 2). More precisely, we propose three probing-based experiments covering three commonly used linguistic tasks (dependency-tree parsing, part-of-speech tagging, and named-entity recognition) and argumentative *stance detection* (*UKP ArgMin*) as a reference.

Ultimately, this work contributes by demonstrating the effectiveness of probing to analyze and compare different generalization scenarios and their gap (like In- vs. Cross-Topic). We conduct three comprehensive experiments to examine generalization capabilities thoroughly:

How do generalization gaps of PLMs differ after pre-training? (§ 4) The probing results showed that generalization gaps differ among the PLMs and are more pronounced for semantic than syntactic probing tasks. Further, we observe apparent probing performance degradation when considering lexical unseen instances - like highly rare entities. In addition, we compare PLMs with large language models (LLMs) and found LLMs have advantages on semantic while PLMs on syntactic probing tasks.

How do PLMs depend on topic-specific tokens? (§ 5) By removing information about topic-specific tokens, PLMs demonstrate apparent differences in their reliance and robustness regarding such vocabulary, which crucially contributes to topical distribution shifts.

How do generalization gaps evolve during fine-tuning? (§ 6) We found fine-tuning significantly impacts the embedding space when we re-probed PLMs tuned on the *UKP ArgMin* dataset for In- or Cross-Topic. We observe that fine-tuning partly erases linguistic properties, which is more pronounced for In- than Cross-Topic fine-tuning.

2 In- and Cross-Topic Probing

The following section formally outlines the used probing setup and tasks before elaborating on the generalization gap, and comparing In- and Cross-Topic probing evaluation.

2.1 Probing Setup and Tasks

We define a probe f_p comprised of a frozen encoder h and linear classifier c without any intermediate layer. This classifier is trained to map instances $X = \{x_1, \ldots, x_n\}$ to targets $Y = \{y_1, \ldots, y_n\}$ for a given probing task. Using a frozen PLM as h, the probe converts x_i into a vector h_i. In detail, we encode the entire sentence, which wraps x_i, and average relevant positions of x_i to find h_i. Relevant positions for the considered probing task are either single tokens for *part-of-speech tagging (POS)*), a span for *named entity recognition (NER)*, or the concatenation of two tokens for *dependency tree parsing (DEP)*. Then, the classifier c utilizes h_i to generate a prediction $\hat{y}_i$, as shown in Equation 1.

$$\hat{y}_i = f_p(x_i) = c(h(x_i)) \qquad (1)$$

2.2 Generalization Gap

Generalization gaps arise when we compare evaluation setups focusing on different capabilities for the same task. This work focuses on gaps occurring when we use data from the same (In-Topic) or different topics (Cross-Topic) for training and evaluation. Such topics $T = \{t_1, \ldots, t_m\}$ are given by a dataset and involve semantically grouping its instances. - i.e., arguments about *Nuclear Energy*. This gap between In- and Cross-Topic is visible in Figure 2, which shows how *NER* instances (in blue) are distributed in the semantic space. For Cross-Topic, entities cover only specific topics and thereby are less broadly spread, while In-Topic ones are spread more broadly since they cover all datasets' topics. Simultaneously, we note more lexically *unseen* entities (in red) during training for Cross-Topic.

In an ideal case, the generalization gaps do not exist because pre-trained language models (PLMs) are robust enough to overcome such distribution shifts between different evaluation setups. However, practically, we saw in Figure 1 these gaps being pronounced on a varying scale for different models.

2.3 Difference between In- and Cross-Topic Evaluation

By evaluating probing tasks for In- and Cross-Topic, we examine the varying generalization gaps between these setups across different PLMs.

Cross-Topic With Cross-Topic evaluation, we investigate how well a probe generalizes when the train, dev, and test instances cover distinct sets of topics $\{T^{(train)}, T^{(dev)}, T^{(test)}\}$. A probe f_p must generalize across the distribution shift in this

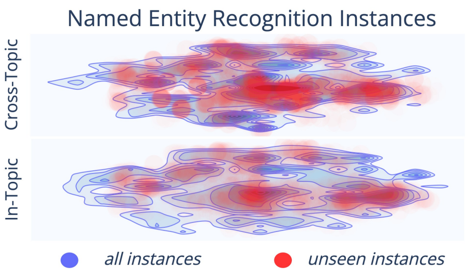

Figure 2: Density plot of the NER test split (blue) for In- and Cross-Topic, encoded with *bert-base-uncased* and reduced with the same t-SNE model (van der Maaten and Hinton, 2008). While both test splits have the same number of instances, the Cross-Topic test split has more instances (a subset of all) with *unseen* vocabulary (red) compared to In-Topic.

setup. This shift originates because distinct topics cover different specific vocabulary Z - i.e., $Z_{(test)}$ for topics in $T^{(test)}$. We formally describe this shift, denoted as ΔZ, as the relative complement between topic-specific vocabulary from train and test instances - $\Delta Z = Z_{(train)} \setminus Z_{(test)}$. For Cross-Topic, we expect ΔZ to be large (Figure 2).

In-Topic In contrast, ΔZ is smaller for the In-Topic setup because instances from every split (train/dev/test) cover the same topics. We expect similar topic distribution and minor semantic differences within these splits compared to Cross-Topic (Figure 2). Thus, we see fewer difficulties for In-Topic because a classifier does not need to generalize across a big distribution shift ΔZ.

Topic-Specific Vocabulary As discussed previously, we see topic-specific vocabulary as one main reason for generalization gaps between In- and Cross-Topic because ΔZ differs for these setups considering a dataset d covering topics $T = t_1, \ldots, t_m$. The topic-specificity of a token z_i is a latently encoded property within the encodings h_i for a token w_i. To capture this property on the token level, we adopt the approach of Kawintiranon and Singh (2021) and use the maximum log-odds-ratio r_i of a token regarding a set of topics T. Firstly, we calculate the odds of finding the token w_i in a topic t_j as $o_{(w_i,t_j)} = \frac{n(w_i,t_j)}{n(\neg w_i,t_j)}$, where $n(w_i, t_j)$ is the number of occurrences of w_i in t_j, and $n(\neg w_i, t_j)$ is the number of occurrences of every other token $\neg w_i$ in t_j. We then compute r as the maximum log-odds ratio of w_i for all topics in T as $r_{(w_i,T)} = max_{t_j \in T}(log(\frac{o(w_i,t_j)}{o_{(w_i,\neg t_j)}}))$.

Model	# Params	Objectives	Data
ALBERT (Lan et al., 2020)	12M	MLM + SOP	16GB
BART (Lewis et al., 2020)	121M	DAE	160GB
BERT (Devlin et al., 2019)	110M	MLM + NSP	16GB
DeBERTa (He et al., 2021)	100M	MLM	80GB
RoBERTa (Liu et al., 2019)	110M	MLM	160GB
ELECTRA (Clark et al., 2020)	110M	MLM+DISC	16GB
GPT-2 (Radford et al., 2019)	117M	LM	40GB

Table 1: Overview of the used PLMs trained on MLM, LM, DISC, NSP, SOP, or DAE objectives.

3 Experimental Setup

We propose three experiments to analyze the varying generalization gap between PLMs after pre-training (§ 4), their dependence on topic-specific vocabulary (§ 5), and the evolution of these gaps during fine-tuning (§ 6). Following, we outline general details about these experiments, while details and results are provided in the subsequent sections.

Models We examine how various PLMs (Table 1) with varying pre-training objectives or architectural designs differ regarding our probing tasks. We cover PLMs pre-trained using masked language modeling (MLM), next sentence prediction (NSP), sentence order prediction (SOP), language modeling (LM), discriminator (DISC), and denoising autoencoder (DAE) objectives. We group them into the ones pre-trained using token- (MLM) and sentence-objectives (NSP, SOP, or DAE) and four purely token-based pre-trained (MLM, LM, DISC). We consider the base-sized variations to compare their specialties in a controlled setup. Apart from these seven contextualized PLMs, we use a static PLM with GloVe (Pennington et al., 2014).

Data We require a dataset with distinguishable topic annotations to evaluate probing tasks in the In- and Cross-Topic evaluation setup. Therefore, we mainly[2] rely on the *UKP ArgMin* dataset (Stab et al., 2018), which provides 25,492 arguments annotated for their argumentative stance (*pro*, *con*, or *neutral*) towards one of eight distinct topics like *Nuclear Energy* or *Gun Control*. Using these instances, we heuristically generate at most 40,000 instances for the three linguistic properties *dependency tree parsing (DEP)*, *part-of-speech tagging (POS)*, or *named entity recognition (NER)* using spaCy.[3] Additionally, we consider the main task

[2]We verified our findings with another dataset in the Appendix § B.1.

[3]We show in the Appendix (§ B.8) that the heuristically generated labels are reliable, and our results are well aligned

of the *UKP ArgMin* dataset (Stab et al., 2018) - *argumentative stance detection (Stance)*. Therefore, we have a topic-dependent reference probe to relate the results of other probes and evaluate the generalization ability of PLMs on real-world tasks after pre-training. We use a three-folded setup for all these four probing tasks to consider the full data variability for both In- and Cross-Topic evaluation. Details about the compositions of these folds and how we ensure a fair comparison between In- and Cross-Topic are provided in the Appendix (§ A.2) as well as examples for probing tasks (Appendix § A.1).

Evaluation We evaluate the three folds of a probing task on three random seeds to get nine measurements per task and calculate the macro averaged F_1 score to consider the variability of labels. Since recent work (Voita and Titov, 2020; Pimentel et al., 2020) questioned whether purely quantitative measures (like F_1) are enough to measure a probe's success, we include the information compression I (Voita and Titov, 2020) for a holistic evaluation. It measures the effectiveness of a probe as the ratio ($\frac{u}{mdl}$) between uniform code length $u = n * log_2(K)$ and minimum description length mdl, where u denotes how many bits are needed to encode n instances with label space of K. We follow *online* variation of mdl and use the same ten-time steps $t_{1:11} = \{\frac{1}{1024}, \frac{1}{512}, ..., \frac{1}{2}\}$, where we train a probe for every t_j with a fraction of instances and evaluate with the same fraction of non-overlapping instances. Exemplary, for, t_9 we use the first fraction of $\frac{1}{4}$ instances to train and another fraction of $\frac{1}{4}$ to evaluate. We find the final mdl as the sum of the evaluation losses of every time step $t_{1:11}$. For Cross-Topic, we group training instances into two groups of distinct topics and sample the same fraction of instances to train and evaluate. Thus, we ensure a similar distribution shift between training and evaluation fractions as in all instances.

4 The Generalization Gap of PLMs

The first experiment shows that the generalization gap already exists after pre-training and varies regarding specific PLMs and probing tasks. We analyze general (Table 2 and Figure 3) and fine-grained (Table 3) results and discuss them for the different evaluating setups, probing tasks, and PLMs. While

with previous work.

	DEP		***POS***		***NER***		***Stance***		*Average*		
	In	*Cross*	*In*	*Cross*	*In*	*Cross*	*In*	*Cross*	*In*	*Cross*	Δ
ALBERT	**43.8**	**39.5**	**80.2**	**78.0**	**48.6**	45.8	54.8	**45.9**	***56.9***	***52.3***	*-4.6*
BART	36.5	36.9	75.4	74.1	**48.7**	**45.3**	**60.8**	44.4	*55.3*	*50.2*	*-5.1*
BERT	25.4	25.6	68.5	67.5	45.4	41.6	56.9	43.0	*49.0*	*44.4*	*-4.6*
DeBERTa	32.8	29.9	73.7	74.6	**48.8**	42.4	**59.8**	**45.8**	*53.4*	*48.2*	*-5.2*
RoBERTa	25.1	23.6	64.0	65.5	**48.4**	42.1	51.8	40.1	*47.3*	*42.8*	*-4.5*
ELECTRA	33.6	33.6	75.3	75.3	41.5	41.2	46.6	43.1	*49.3*	*48.3*	***-1.0***
GPT-2	25.2	23.9	63.5	61.9	45.5	38.6	51.1	38.4	*46.3*	*40.7*	*-5.6*
GloVe	12.1	11.9	26.5	26.2	43.4	37.5	41.6	34.1	*30.9*	*27.4*	*-3.5*
Avg. Δ	*-1.2*		*-0.5*		*-4.5*		*-11.0*		-	-	-

Table 2: In- and Cross-Topic probing results for eight PLMs. We report the macro F_1 over three random seeds, the average difference between the two setups (last row), and their average per PLM (last three columns). Best results within a gap of 1.0 are marked by columns.

		DEP			***POS***			***NER***		
		all	*Δ seen*	*Δ unseen*	*all*	*Δ seen*	*Δ unseen*	*all*	*Δ seen*	*Δ unseen*
	Instance Ratio	-	85%	15%	-	86%	14%	-	65%	35%
In-Topic	ALBERT	43.8	+0.21	-3.2	80.2	+0.41	-17.7	48.6	+1.1	-5.8
	BART	36.5	+0.13	-3.0	75.4	+0.20	-16.5	48.7	+1.3	-7.0
	BERT	25.4	-0.02	-0.8	68.5	+0.20	-16.5	45.4	+1.0	-5.8
	DeBERTa	32.8	+0.07	-1.5	73.7	+0.09	-12.7	48.8	+1.0	-5.6
	RoBERTa	25.1	-0.01	-0.9	64.0	-0.04	-15.5	48.4	+1.0	-5.7
	Average	-	*-0.08*	*-1.9*	-	*+0.17*	*-15.8*	-	*+1.1*	*-6.0*
	Instance Ratio	-	78%	22%	-	81%	19%	-	51%	49%
Cross-Topic	ALBERT	39.5	+0.03	-2.3	78.0	+0.51	-12.9	45.8	+2.2	-5.3
	BART	36.9	+0.01	-4.0	74.1	+0.24	-16.5	45.3	+2.4	-5.8
	BERT	25.6	-0.09	-0.7	67.5	+0.20	-14.0	41.6	+1.9	-5.1
	DeBERTa	29.9	-0.07	-1.3	74.6	+0.14	-11.7	42.4	+2.0	-5.2
	RoBERTa	23.6	-0.22	-0.3	65.5	+0.00	-14.7	42.1	+1.9	-5.2
	Average	-	*-0.08*	*-1.7*	-	*+0.22*	*-14.0*	-	*+2.1*	*-5.3*

Table 3: Performance difference of *seen* and *unseen* instances compared to the full set (*all*). We report for ALBERT, BART, BERT, DeBERTa, & RoBERTa, and include the ratio of *seen* and *unseen* instances.

we mainly focus on mid-size PLMs usable for fine-tuning, we will close this experiment by comparing them with large language models (LLMs) in § 4.

Design We evaluate eight PLMs using the probe f_p (§ 2.1) on the probing tasks *DEP*, *POS*, *NER*, and *Stance*. We verified these tasks by observing significant performance drains when evaluating them on randomly initialized PLMs (Appendix § B.2). For a holistic evaluation, we provide results by grouping instances into two categories: *seen* and *unseen*. We define *seen* instances as already processed during training but in another context. For example, the pronoun *he* might appear in both training and test data, but in distinct sentences. By evaluating the PLMs on *seen* instances, we gain insights into the influence of token-level lexical information versus context information from surrounding tokens. In contrast, *unseen* instances were not encountered during the training of a probe. They allow assessing whether PLMs generalize to tokens that are similar to some extent (such as *Berlin* and *Washington*) but not seen during training.

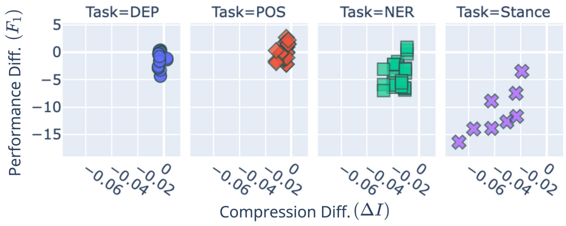

Figure 3: Comparision of the difference in ΔF_1 and ΔI between Cross-Topic and In-Topic for all eight PLMs on the four probing tasks.

Results for Evaluation Setups Upon analyzing Table 2, we observe a clear generalization gap between In- and Cross-Topic evaluation for all tasks and PLMs. As shown in Figure 3, the magnitude of this gap (ΔF_1) correlates with the difference in compression (ΔI). Interestingly, we find a stronger correlation between F_1 and I for Cross-Topic ($\rho = 0.72$) as compared to In-Topic ($\rho = 0.69$). Thus, a higher performance level, like for In-Topic, leaves less room for compression improvements.

Further, we examine the performance of *seen* and *unseen* instances in Table 3. It shows that *seen* performs slightly better than *all*, while *unseen* ones underperform the complete set (*all*) and *seen* instances. Considering the average over PLMs, there are fewer relative gains for *seen* for In-Topic and more loss for *unseen* instances (+1.2, -6.0 for *NER*) compared to Cross-Topic (+2.0, -5.3 for NER). This observation relates to the lower percentage of *unseen* instances (i.e., made of topic-specific terms) for In- compared to Cross-Topic. We see *unseen* instances on In-Topic are harder and cover rare vocabulary, and *seen* instances on Cross-Topic are easier and made of general terms. These results confirm our theoretic and semantic assumptions (§ 2).

Results for Probing Tasks Considering Table 2 and Figure 3, we note higher generalization gaps (*Avg.* Δ of -4.5 and -11.0) for semantic tasks (*NER* and *Stance*) than for syntactic tasks (*DEP* and *POS*) - *Avg.* Δ of -1.2 and -0.5. We verify this trend with results in the Appendix (§ B.5), where we observe a more pronounced gap for semantic *NER* classes (like ORG) than for syntactic ones - like ORDINAL.

Next, we separately compare tasks for *seen* and *unseen* instances. *DEP* shows the slightest performance difference compared to *all*. We assume this is due to the pairwise task nature, which leads to a larger shared vocabulary between *unseen* and training instances. We assume frequent words (like *of*) are part of the *unseen* instances. In contrast, apparent differences between *NER* and *POS* are visible - with less performance drain on *unseen* instances for *NER* than *POS*. Therefore, we assume for *NER* a higher semantic overlap with training instances since they could include - as being an n-gram - words from the training vocabulary. In contrast, tokens of *unseen POS* instances are always single words; thus, we assume a smaller semantic overlap with the training.

Results for Encoding Models We now compare PLMs amongst themselves. The four best-performing PLMs of In-Topic differ up to 7.6 (ALBERT - BERT), while for Cross-Topic, this difference narrows to 4.1 (ALBERT - ELECTRA). These results confirm the varying generalization gap between them and, again, that we can not transfer conclusions from one evaluation setup to another. For example, the probing performance of BART for In-Topic *Stance* is the best and the third best for Cross-Topic.

Generally, we do not see a clear correlation between better average performance and a smaller generalization gap. PLMs like DeBERTa perform better for In- and Cross-Topic but show a bigger gap (-5.1) compared to lower performing PLMs like ELECTRA (-1.0), but there are also worse PLMs with a bigger gap (GPT-2, -5.6) or better ones with a smaller gap (ALBERT, -4.6). Overall, we see the generalization gap being more pronounced for better-performing PLMs.

Considering absolute performance, ALBERT and BART performs the best on average for both evaluation setups, while ELECTRA excels *POS* and *DEP*, and DeBERTa performs for *NER* and *Stance*. In contrast, BERT, RoBERTa, GPT-2, and GloVeunderperform the others. Thus, PLMs with architectural regularization, such as layer-wise parameter sharing (ALBERT), encoder-decoder layers (BART), disentangled attention (DeBERTa), or discriminator (ELECTRA), tend to provide higher Cross-Topic performance. Similarly, regularized PLMs, such as ALBERTor DeBERTa, generally achieve more performance gains for *seen* instances and fewer performance drops for *unseen* ones than models without regularization such as BERT or RoBERTa. We hypothesize that architectural and regularization aspects equip PLMs with a more generalizable and robust

	DEP		***POS***		***NER***		***Stance***		*Average*		
	In	*Cross*	*In*	*Cross*	*In*	*Cross*	*In*	*Cross*	*In*	*Cross*	Δ
ALBERT	**43.8**	**39.5**	**80.2**	**78.0**	48.6	45.8	54.8	**45.9**	*56.9*	***52.3***	*-4.6*
BART	36.5	36.9	75.4	74.1	48.7	45.3	60.8	44.4	*55.3*	*50.2*	*-5.1*
T5 (3B)	33.9	32.5	68.5	68.9	48.3	42.2	53.2	42.1	*51.0*	*46.5*	***-4.5***
FLAN-T5 (3B)	33.1	29.7	66.8	66.9	48.5	43.1	56.0	**45.1**	*51.1*	*46.2*	*-4.9*
GPT-Neo (2.7B)	36.4	33.1	76.4	**77.1**	**52.9**	**49.6**	**62.4**	40.5	***57.0***	*50.1*	*-6.9*

Table 4: Results (macro F_1) of the four probing tasks using the two overall best-performing PLMs (ALBERT and BART) in the In- and Cross-Topic setup based on the *ArgMin* dataset (Table 2) and three LLMs.

encoding space.

Results for Larger Models We compare in Table 4 three relevant and open accessible LLMs with the two best performing models (ALBERT and BART) on the first experiments. In general, we see that the scaling law (Brown et al., 2020) applies to our setting for LLMs with LM-based pre-training. Specifically, GPT-Neo (2.7B) (Black et al., 2021) is more robust and outperforms GPT-2 while performing on par or slightly better than the other PLMs. In contrast, T5 (3B) (Raffel et al., 2022) or FLAN-T5 (3B) (Chung et al., 2022) underperform PLMs on syntactic tasks and perform slightly worse on semantic tasks. We hypothesize that their task-specific pre-training result in less robust and generalizable token encoding space. This is in line with the fact that amongst these two LLMs, FLAN-T5 (3B) performs worse than T5 (3B), which experienced additional instruction-based pre-training.

5 The Dependence on Topic-Specific Vocabulary

To this point, we saw that the generalization gap varies between different PLMs and probing tasks. Since we see topic-specific vocabulary crucially affects generalization gaps, we analyze the varying dependence on the topic-specific vocabulary of PLMs using *Amnesic Probing* (Elazar et al., 2021). We observe clear differences among PLMs and therefore assume that their embedding space clearly differs beyond single evaluation metrics. Therefore, we emphasize considering various PLMs when using *Amnesic Probing*. Additional insights of comparing *seen* and *unseen* instance and distinct NER classes are provided in the Appendix (§ B.4, § B.6).

Design To measure how PLMs depend on topic-specific vocabulary, we employ *Amnesic Probing* (Elazar et al., 2021) to remove the latently encoded topic-specificity z_i from the embeddings h_i of a token w_i. More precisely, we compare how the performance of a probing task (like *NER*) changes when we remove z_i. A more negative effect indicates a higher dependence on topic-specific vocabulary, while this property is a hurdle when performance improves. We first train a linear model on token-level topic-specificity r (§ 2.3). To shape it as a classification task, we categorize r into three classes (*low*, *medium*, *high*). [4] Next, we find a projection matrix P that projects all embeddings h_i - gathered as H - using the learned weights W_l of l to the null space as $W_l PH = 0$. Using P we update h_i by neutralizing topic-specificity from the input as $h_i^{'} = Ph_i$ before training the probe. Following (Elazar et al., 2021), we verified our results by measuring less effect of removing random information from h_i (see Appendix § B.3).

Results Considering Figure 4, we see ALBERT, BART, and BERT depend less on topic-specific vocabulary. We see their diverse pre-training (token- and sentence-objectives or sentence denoising) results in a more robust embedding space. Surprisingly, they show positive effects (3.2 for *DEP* for BART) when removing topic-specificity. This could remove potentially disturbing parts of the embedding space. Similarly, GPT-2 is less affected by the removal - we assume this is due to its generally lower performance level. Therefore, it has less room for performance drain, and capturing topic-specificity is less powerful.

Comparing In- and Cross-Topic setups shows a narrowing generalization gap for more affected models (like RoBERTa and GloVe on *NER* or *Stance*). Simultaneously, less affected PLMs either maintain the gap or enlarge it slightly - like BART on *DEP*, *NER*, or *Stance*. Further, DeBERTa, RoBERTa, ELECTRA, and GloVe rely more on topic-specific vocabulary since they show significant performance loss (up to 34.6 for GloVe on *POS*) when removing this information. Specifically, GloVe as a static language model, and RoBERTa is affected the highest for all tasks. ELECTRA shows similar behavior, but is less pronounced for *POS*. Thus, its reconstruction pre-training objective provides a more robust embedding space than purely MLM (DeBERTa or RoBERTa). Comparing, DeBERTa and RoBERTa, DeBERTa is less affected by the removal of semantic tasks (*NER* and *Stance*). We hypothesize that distinguishing between token content and token position via disentangled attention makes De-

[4] Please find examples in the Appendix § A.6.

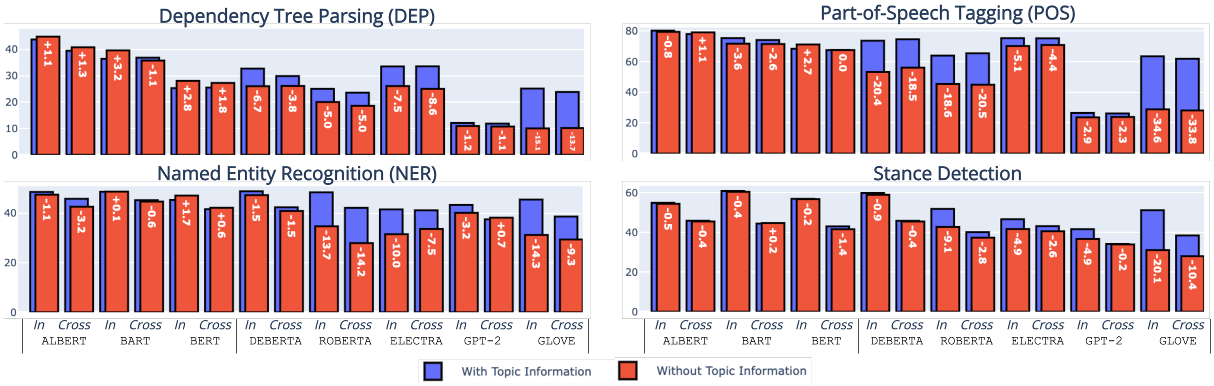

Figure 4: Comparison of the probing results with (blue bars) or without (red bars) topic information. The white text indicates the difference between these two scenarios ($\Delta F_1^{\setminus T}$).

BERTa more robust for the semantic than for syntactic tasks (*DEP* and *POS*).

6 The Evolution of the Generalization Gap during Fine-Tuning

Finally, we re-evaluate fine-tuned PLMs using our proposed probing setups and show that fine-tuning leads to a drain in probing performance. We use these results to retrace apparent differences between evaluation setups and the varying generalization gap between PLMs. This is relevant for a broader understanding of how fine-tuning affects PLMs (Mosbach et al., 2020; Kumar et al., 2022a), and what they learn during fine-tuning (Merendi et al., 2022; Ravichander et al., 2021).

Design We fine-tune the PLMs on an argumentative *stance detection* task and re-evaluate them on the probing tasks *DEP*, *POS*, and *NER*. To be consistent with our probing setup, we used the same folds for fine-tuning. Further details are in the Appendix (§ A.5). We compare these results with the probing performance of their pre-trained counterparts (§ 4 and § 5) and correlate this change with the generalization gap observed on the downstream task. We limit our analysis to ALBERT, BERT, BART, DeBERTa, and RoBERTa.

Results Table 5 shows that fine-tuning clearly boost the performance on *Stance* compared to the probing performance (§ 4) but leads to a clear performance drop (ΔF_1) for both evaluation setups and the probing tasks. Cross-Topic achieved more gains on average (+12.6) and fewer drains (-17.1) on the three linguistic properties than In-Topic (+9.5, -20.4). On average, we assume that In-Topic fine-tuning affects the encoding space of PLMs more heavily than Cross-Topic. Regarding the different probing tasks, the performance drain is more pronounced for syntactic tasks (*DEP* and *POS*) than semantic tasks (*NER*). This hints that PLMs acquire competencies of semantic nature - which holds for *stance detection*. Similarly, removing topic-specificity influences fine-tuned PLMs the least for *NER*. At the same time, this removal is more pronounced for Cross-Topic. This confirms the assumption that the Cross-Topic setup has smaller effects on PLMs internals, since we saw big impacts of this removal (§ 5).

		Stance	***DEP***	***POS***	***NER***	***Avg.***	***DEP***	***POS***	***NER***
		F_1 *fine-tuned*	ΔF_1 *probing*				$\Delta F_1^{\setminus T}$		
In-Topic	ALBERT	55.4 +0.6	-27.3	-40.2	-25.0	-30.8	-0.6	-3.0	-0.1
	BART	69.8 +9.0	-17.3	-32.2	-4.0	-17.8	-0.8	-4.0	+0.3
	BERT	67.2 +10.3	-7.5	-24.8	+1.0	-10.4	+0.4	+0.7	+1.1
	DeBERTa	**70.1 +10.3**	-13.2	-25.3	-8.8	-15.8	-0.8	-3.8	-0.4
	RoBERTa	68.9 +17.1	-19.7	-48.6	-29.7	-27.2	-0.8	-3.0	-0.7
	Avg.	66.3 +9.5	-16.6	-32.6	-12.1	-20.4	-0.5	-2.6	+0.1
Cross-Topic	ALBERT	51.4 +5.5	-14.4	-20.3	-12.6	-15.8	+1.6	-1.3	+2.1
	BART	**61.9 +17.5**	-16.5	-33.9	-5.4	-18.6	-1.0	-3.5	-1.6
	BERT	56.6 +13.6	-5.7	-19.5	+0.6	-8.2	+0.7	+0.6	+1.2
	DeBERTa	55.9 +10.1	-13.4	-33.4	-11.8	-19.5	-1.2	-8.6	+1.6
	RoBERTa	55.5 +15.4	-16.6	-48.3	-23.1	-23.5	-1.9	-4.8	-0.3
	Avg.	56.3 +12.6	-13.0	-29.3	-9.1	-17.1	-0.4	-3.5	+0.6

Table 5: Results of evaluating our probing setup on fine-tuned PLMs on *Stance*. The first column shows these fine-tuned results and the gained improvement compared to probing for *Stance* on pre-trained PLMs (Table 2). Next, we show performance differences between pre-trained and fine-tuned PLMs (ΔF_1 *probing*) and how removing topic-specificity affects the fine-tuned PLMs ($\Delta F_1^{\setminus T}$).

Considering the single PLMs, we see apparent differences. For example, ALBERT, with its shared architecture and priorly best-performing PLM, experiences big probing performance drains and the smallest fine-tuning gains (+0.6, +5.5). In contrast, we note effective fine-tuning of BERTwith +10.3 for In- and +13.6 for Cross-Topic, and that it lost the least probing performance. Comparing RoBERTa and DeBERTa reveals again the effectiveness of architectural regularization of DeBERTa. RoBERTa shows the most gains when fine-tuning on *Stance* and almost catching up with DeBERTa. However, it experiences a more clear performance drain (-27.2, -23.5) regarding the probing tasks for In- and Cross-Topic compared to DeBERTa (-15.8, -19.5). Next, we focus on BART and its superior Cross-Topic performance on *Stance*. It seems already well-equipped for this downstream task due to its high In-Topic probing performance on *Stance*. Therefore, it can learn the task more robustly during fine-tuning.

7 Related Work

The rise of PLMs (Devlin et al., 2019; Liu et al., 2019; Radford et al., 2019; He et al., 2021) enabled big success on a wide range of tasks (Wang et al., 2018, 2019). Nevertheless, they still fall behind on more realistic Cross-Topic, like generalizing towards unseen topics (Stab et al., 2018; Gulrajani and Lopez-Paz, 2021; Allaway and McKeown, 2020). One primary reason is that PLMs often rely on unwanted spurious correlations. Despite PLMs seeing such vocabulary during pre-training, they failed to consider test vocabulary in the required fine-grained way (Thorn Jakobsen et al., 2021; Reuver et al., 2021). Further, Kumar et al. (2022b) found linear models can outperform fine-tuning PLMs when considering out-of-distribution data. Thus, a broader understanding of PLMs in challenging evaluation setups is crucial.

Probing (Belinkov et al., 2017; Conneau et al., 2018; Peters et al., 2018) helps to analyze inners of PLMs. This includes to examine how linguistic (Tenney et al., 2019a,b), numeric (Wallace et al., 2019), reasoning (Talmor et al., 2020), or discourse (Koto et al., 2021) properties are encoded. Other works focus on specific properties used for other tasks (Elazar et al., 2021; Lasri et al., 2022), or fine-tuning dynamics (Merchant et al., 2020; Zhou and Srikumar, 2022; Kumar et al., 2022b). However, these works target the commonly used *In-Topic* setup and less work considering Cross-Topic setups. Aghazadeh et al. (2022) analyzed metaphors across domains and language, or Zhu et al. (2022) cross-distribution probing for visual tasks. They found that models generalize to some extent across distribution shifts in probing-based evaluation. Nevertheless, these works focus on specialized tasks and consider the generalizations across distributions in isolation. In contrast, we propose with our experiments a more holistic probing-based evaluation of PLMs, covering different generalization aspects after pre-training and fine-tuning.

8 Conclusion

Discussion We demonstrated the practical usefulness of probing to analyze and compare PLMs on different generalization setups. Thereby, we show that generalization gaps vary regarding PLM and probing tasks. Further, we provide preliminary insights into how LLMs differ from PLMs using our proposed setup and found they tend to have strong performance for semantic tasks. By re-evaluating fine-tuned PLMs, we found that generalization gaps arise differently and linguistic properties partly disappear during training - being more prominent for In- than Cross-Topic fine-tuning. Overall, we found architectural regularization and diverse pre-training objectives positively affect the generalizability and robustness of PLMs - like, being less influenced by removing the topic-specificity of tokens. We verified our results using a second dataset from the social media domain (Conforti et al., 2020) - details in the Appendix § B.1.

To conclude, we analyzed and compared PLMs on different generalization setups and shed light on why generalization gaps evolve differently across PLMs. We emphasized the importance of different pre-training or architectural specialties to improve the robustness of PLMs. Further, we demonstrated how probing could help to identify promising PLMs like BART, which seems to overcome semantic difficulties for Cross-Topic more quickly due to its high In-Topic probing performance on the downstream task.

Outlook We extended the probing focus to analyze and compare In- and Cross-Topic generalization capabilities and their varying generalization gap of PLMs. With our findings in mind, we see regularly probing PLMs and LLMs on new tasks and considering forthcoming learning paradigms as indispensable for a holistic evaluation of their verity and multiplicity. Therefore, we will continue to analyze language models, including a broader set of tasks to increase our understanding of how, why, and where they differ.

Ethical Considerations and Limitations

Automatic Annotations for Linguistic Properties Our experiments require all instances origin in the same datasets with topic annotations. Thanks to this condition, we align all our experiments, like probing PLMs, with the same data as they got pre-trained. Therefore, we minimize other influences like semantic shifts of other datasets. However, there are no corresponding annotations for linguistic properties, which forces us to rely on automatically gathered annotations. This work addresses this issue by transparently stating the libraries and models we used to derive these annotations and providing the source code and the extracted labels in our repository. We compared our results (§ B.8) with previous work (Tenney et al., 2019a,b; Hewitt and Liang, 2019) and found our results well aligned. Further, we verify the probing task results on the different PLMs with randomly initialized counter-parts (§ B.2) and confirm our findings with a second dataset (§ B.1).

Definition of Topic-Specific Vocabulary This work considers a topic as a semantic grouping provided by a given dataset. As previously mentioned, this focus on the context of one dataset allows in-depth and controlled analysis, like examining the change of PLMs during fine-tuning. On the other hand, we need to thoroughly re-evaluate other datasets, since the semantic space and granularity of the topic are different in almost every other dataset. Nevertheless, results in the Appendix (§ B.1) let us assume that our findings correlate with other datasets and domains. Further, we consider only token-level specific vocabulary, as done previously in literature (Kawintiranon and Singh, 2021). We think that considering n-grams could give a better approximation of topic-specific terms. Still, we do not take them into account because *Amnesic Probing* (Elazar et al., 2021) require token-level properties to apply resulting intervention on token-level tasks like *POS*.

Impact of PLMs Design choices This work analyzes PLMs regarding a set of different properties like pre-training objectives or architectural regularization. However, we do not claim the completeness of these aspects nor a clear causal relationship. Making such a final causal statement would require significant computational resources to pre-train models to verify single properties with full certainty. Instead, we use same-sized model variations, evaluate all probes on three folds and three random seeds to account for data variability and random processes, and verify our results on a second dataset. Nevertheless, we use them to correlate results on aggregated properties (like having diverse pre-training objectives or not) and not on single aspects like the usefulness of the *Sentence-Order* objective.

References

Ehsan Aghazadeh, Mohsen Fayyaz, and Yadollah Yaghoobzadeh. 2022. Metaphors in pre-trained language models: Probing and generalization across datasets and languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2037–2050, Dublin, Ireland. Association for Computational Linguistics.

Emily Allaway and Kathleen McKeown. 2020. Zero-Shot Stance Detection: A Dataset and Model using Generalized Topic Representations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8913–8931, Online. Association for Computational Linguistics.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017. What do neural machine translation models learn about morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 861–872, Vancouver, Canada. Association for Computational Linguistics.

Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. 2021. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow. If you use this software, please cite it using these metadata.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

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*.

Costanza Conforti, Jakob Berndt, Mohammad Taher Pilehvar, Chryssi Giannitsarou, Flavio Toxvaerd, and Nigel Collier. 2020. Will-they-won't-they: A very large dataset for stance detection on Twitter. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1724, Online. Association for Computational Linguistics.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. 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.

Ishaan Gulrajani and David Lopez-Paz. 2021. In search of lost domain generalization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

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, Hong Kong, China. Association for Computational Linguistics.

Kornraphop Kawintiranon and Lisa Singh. 2021. Knowledge enhanced masked language model for stance detection. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4725–4735, Online. Association for Computational Linguistics.

Fajri Koto, Jey Han Lau, and Timothy Baldwin. 2021. Discourse probing of pretrained language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 3849–3864. Association for Computational Linguistics.

Ananya Kumar, Aditi Raghunathan, Robbie Matthew Jones, Tengyu Ma, and Percy Liang. 2022a. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Ananya Kumar, Aditi Raghunathan, Robbie Matthew Jones, Tengyu Ma, and Percy Liang. 2022b. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations.

Karim Lasri, Tiago Pimentel, Alessandro Lenci, Thierry Poibeau, and Ryan Cotterell. 2022. Probing for the usage of grammatical number. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8818–8831, Dublin, Ireland. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for 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*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

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

Federica Merendi, Felice Dell'Orletta, and Giulia Venturi. 2022. On the nature of BERT: correlating fine-tuning and linguistic competence. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pages 3109–3119. International Committee on Computational Linguistics.

Marius Mosbach, Anna Khokhlova, Michael A. Hedderich, and Dietrich Klakow. 2020. On the Interplay Between Fine-tuning and Sentence-level Probing for Linguistic Knowledge in Pre-trained Transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2502–2516, Online. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher 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, Doha, Qatar. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018. Dissecting contextual word embeddings: Architecture and representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, Brussels, Belgium. Association for Computational Linguistics.

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. 2020. Information-theoretic probing for linguistic structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4609–4622, Online. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2022. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Abhilasha Ravichander, Yonatan Belinkov, and Eduard H. Hovy. 2021. Probing the probing paradigm: Does probing accuracy entail task relevance? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 3363–3377. Association for Computational Linguistics.

Xiaoying Ren, Jing Jiang, Ling Min Serena Khoo, and Hai Leong Chieu. 2021. Cross-topic rumor detection using topic-mixtures. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1534–1538, Online. Association for Computational Linguistics.

Myrthe Reuver, Suzan Verberne, Roser Morante, and Antske Fokkens. 2021. Is stance detection topic-independent and cross-topic generalizable? - a reproduction study. In *Proceedings of the 8th Workshop on Argument Mining*, pages 46–56, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Upendra Sapkota, Thamar Solorio, Manuel Montes, Steven Bethard, and Paolo Rosso. 2014. Cross-topic authorship attribution: Will out-of-topic data help? In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1228–1237, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Christian Stab, Tristan Miller, Benjamin Schiller, Pranav Rai, and Iryna Gurevych. 2018. Cross-topic argument mining from heterogeneous sources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3664–3674, Brussels, Belgium. Association for Computational Linguistics.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2020. oLMpics-On What Language Model Pre-training Captures. *Transactions of the Association for Computational Linguistics*, 8:743–758.

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 *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Terne Sasha Thorn Jakobsen, Maria Barrett, and Anders Søgaard. 2021. Spurious correlations in cross-topic argument mining. In *Proceedings of *SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics*, pages 263–277, Online. Association for Computational Linguistics.

Laurens van der Maaten and Geoffrey E. Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

Elena Voita and Ivan Titov. 2020. Information-theoretic probing with minimum description length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 183–196, Online. Association for Computational Linguistics.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. 2019. Do NLP models know numbers? probing numeracy in embeddings. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 5306–5314. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

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

Yichu Zhou and Vivek Srikumar. 2022. A closer look at how fine-tuning changes BERT. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1046–1061, Dublin, Ireland. Association for Computational Linguistics.

Zining Zhu, Soroosh Shahtalebi, and Frank Rudzicz. 2022. OOD-probe: A neural interpretation of out-of-domain generalization. In *ICML 2022: Workshop on Spurious Correlations, Invariance and Stability*.

A Additional Details of the Experiments

A.1 Probing Tasks

Table 6 shows examples and additional details of the different probing tasks.

A.2 Fold Composition

We rely on a three-folded evaluation for In- and Cross-Topic for a generalized performance measure. These folds cover every instance exactly once in a test split. In addition, we require that In- and Cross-Topic train/dev/test splits have the same number of instances for a fair comparison, as visualized in Figure 5. For Cross-Topic, we make sure that every topic $\{t_1, ..., t_m\}$ is covered precisely once by one of the three test splits $X^{(test)}_{cross}$. To compose $X^{(train)}_{cross}$ and $X^{(dev)}_{cross}$, we randomly distribute the remaining topics for every fold. For In-Topic, we randomly[5] form subsequent test splits $X^{(test)}_{in}$ for every fold from all instances $\{x_1, ..., x_m\}$. $X^{(train)}_{in}$ and $X^{(dev)}_{in}$ are then randomly composed for every fold using the remaining instance set following the dimension of $X^{(train)}_{cross}$ and $X^{(dev)}_{cross}$.

A.3 Training Setup

For all our experiments, we use NVIDIA RTX A6000 GPUs, python (3.8.10), transformers (4.9.12), and PyTorch (1.11.0).

A.4 Probing Hyperparameters

Further, we use for the training of the probes the following fixed hyperparameters: 20 epochs, where we find the best one using dev instances; AdamW (Loshchilov and Hutter, 2019) as optimizer; a batch size of 64; a learning rate of 0.0005; a dropout rate of 0.2; a warmup rate of 10% of the steps; random seeds: $[0, 1, 2]$

In addition, we use the following tags from the huggingface model hub:

- `albert-base-v2`
- `bert-base-uncased`
- `facebook/bart-base`
- `microsoft/deberta-base`
- `roberta-base`

[5]We expect that all folds cover all topics given the small number of topics (8) and the big number of instances.

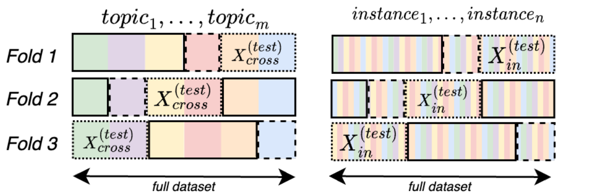

Figure 5: Overview of the In- and Cross-Topic setup using three folds. The colour indicates a topic; solid lines train-, dotted lines dev-, and dashed lines test-splits.

- `google/electra-base-discriminator`
- `gpt2`
- `t5-3b`
- `google/flan-t5-xl`
- `EleutherAI/gpt-neo-2.7B`

A.5 Fine-Tuning Hyperparameters

To fine-tune on *stance detection*, we use the following setup: 5 epochs, where we find the best one using dev instances; AdamW (Loshchilov and Hutter, 2019) as optimizer; a batch size of 16; a learning rate of 0.00002; a warmup rate of 10% of the steps; random seeds: $[0, 1, 2]$.

A.6 Token-Level Examples for Topic Relevance

In § 5, we use the binned topic-specificity (§ 5) for each token. We show in Table 7 examples for three bins *low*, *medium*, and *high*. The first bin (*low*) is made of tokens, which barely occur in the dataset. The second one (*medium*) consists of tokens which are part of most topics. Finally, the last bin (*high*) includes tokens with a high topic relevance for ones like *Cloning* or *Minimum Wage*.

B Further Results

B.1 Generalization Across Datasets

With Table 8, Figure 6, and Table 9, we verify the results of § 4, § 5, and § 4 using another *stance detecion* dataset. Namely, we use the *wtwt* (*will-they-wont-they*) (Conforti et al., 2020) dataset which covers 51.284 tweets annotated either *support*, *refute*, *comment*, or *unrelated* towards five financial topics. For the overall performance comparison between In- and Cross-Topic, the results show the

Task	Example	Label	# Instances	# Labels
`DEP`	I think there is a lot we can learn from Colorado and Washington State.	*nsubj*	40,000	41
`POS`	I think there is a lot we can learn from Colorado and Washington State.	*PRON*	40,000	17
`NER`	I think there is a lot we can learn from Colorado and Washington State.	*PERS*	25,892	17
`Stance`	I think there is a lot we can learn from Colorado and Washington State.	*PRO*	25,492	3

Table 6: Overview and examples of the different probing tasks.

low	*medium*	*high*
fianc, joking, validate, latitude, poignantly, informative ameliorate, bonding, mentors brigade, emancipation, deriving, ignatius, 505, nominations, electorate, SWPS, 731	as, on, take, some, like, how, so, one, these, instead, while, ago where, came, still, many, come, engage, seems	cloning, uniform, wage, marijuana, minimum, gun, cloned, wear, clone, nuclear, energy, penalty, uranium, legalization, cannabis, execution, wast, employment

Table 7: Examples of tokens with a *low*, *medium*, or *high* token relevance following § 4.

	DEP		***POS***		***NER***		***Stance***		*Average*		
	In	*Cross*	*In*	*Cross*	*In*	*Cross*	*In*	*Cross*	*In*	*Cross*	Δ
ALBERT	**33.5**	**32.9**	**75.1**	**74.2**	30.9	28.6	**57.3**	32.8	***49.1***	*42.1*	*-7.0*
BART	**32.9**	**33.1**	63.2	62.1	**32.4**	**30.5**	51.9	**47.2**	*45.1*	***43.2***	*-1.9*
BERT	21.6	21.2	54.8	55.9	27.2	27.8	47.4	32.1	*37.8*	*34.2*	*-3.6*
DeBERTa	26.9	27.6	69.6	67.9	29.4	28.5	49.5	35.7	*43.9*	*40.0*	*-3.9*
RoBERTa	20.4	19.9	54.7	53.5	26.1	25.5	37.0	37.8	*35.6*	*34.2*	*-1.4*
ELECTRA	26.6	26.6	69.6	68.6	21.7	24.1	35.1	36.7	*38.2*	*39.0*	***+0.8***
GPT-22	16.9	16.5	42.2	42.2	25.1	24.0	40.8	32.6	*31.2*	*28.8*	*-2.4*
GloVe	12.9	12.2	23.5	22.6	28.1	24.6	45.2	34.2	*27.4*	*23.4*	*-4.0*
Avg. Δ	*-0.3*		*-0.7*		*-0.9*		*-9.5*		-	-	-

Table 8: Results of the four probing tasks using eight PLMs in the In- and Cross-Topic setup. We report the mean F_1 (macro averaged) over three random seeds, the average difference between the two evaluation setups per task (last row), and their average per PLM (last two columns). Best-performing results within a margin of 1pp are marked for every task and setup.

	DEP		***POS***		***NER***		***Stance***		*Average*		
	In	*Cross*	*In*	*Cross*	*In*	*Cross*	*In*	*Cross*	*In*	*Cross*	Δ
ALBERT	**33.5**	32.9	**75.1**	**74.2**	30.9	28.6	57.3	32.8	*49.1*	*42.1*	*-7.0*
BART	**32.9**	**33.1**	63.2	62.1	32.4	30.5	51.9	**47.2**	*45.1*	*43.2*	*-1.9*
T5 (3B)	25.5	26.3	59.7	59.3	34.9	**36.4**	53.4	38.7	*43.4*	*40.2*	*-3.2*
FLAN-T5 (3B)	25.5	26.3	59.7	59.3	34.9	**36.4**	53.4	38.7	*43.4*	*40.2*	*-3.2*
GPT-Neo (2.7B)	29.5	29.7	69.4	68.4	**37.4**	34.3	**74.9**	43.9	***52.8***	***44.1***	*-8.7*

Table 9: Results (macro F_1) of the four probing tasks using the overall best PLMs (ALBERT and BART) in the In- and Cross-Topic setup based on the *wtwt* dataset (Table 8) and three LLMs.

	DEP		***POS***		***NER***	
	Random	Δ	*Random*	Δ	*Random*	Δ
ALBERT	1.4	-42.4	6.8	-41.8	3.4	-76.8
BART	1.4	-35.1	5.0	-43.7	2.7	-72.7
BERT	2.7	-22.7	9.4	-36.0	4.6	-63.9
DeBERTa	7.0	-25.8	16.3	-32.5	16.1	-57.6
RoBERTa	2.2	-22.9	11.0	-37.4	4.7	-59.3
ELECTRA	1.7	-31.9	8.4	-33.1	3.8	-71.5
GPT-2	5.8	-19.4	12.3	-33.2	12.5	-51.0

Table 10: Results of evaluating *DEP*, *POS*, and *NER* using the seven contextual PLMs (random initialized) for In-Topic and the difference to their pre-trained counterparts in Table 2.

same trend as we already saw in § 4, but on a lower level. We assume that this is mainly due to this dataset's more specific domain (twitter) compared to *UKP ArgMin*. Focusing on the influence of topic-specific vocabulary verifies the previously presented results (§ 5) again. PLMs pre-trained with purely token-based objectives highly depend on topic-specific vocabulary. Considering LLMs (Table 9), we see again similar behavior as on the *ArgMin* dataset (§ 4).

B.2 Comparison of Probing Tasks against Random Initialized PLMs

We show in Table 10 and Table 11 the results of running the three linguistic probes on the seven contextualized PLMs in their random initialized version. For In- and Cross-Topic, there is a clear performance drop of having random initialized models.

B.3 The Effect of Removing Random Information

We saw in § 5 that removing topic-specificity has a big impact for some models (like RoBERTa or ELECTRA) but at the same time can even boost the performance of others like BERT. As suggested in Elazar et al. (2021), we apply a sanity check by removing random information from the encodings of PLMs. Following the results in Figure 7, removing random information (green bars) performs in between the scenarios with (blue bars) or without (red bars) topic information for cases where we see a clear negative effect when removing topic information. In contrast, removing random information can produce a more pronounced effect when we see performance improvements. This observation backs our assumption that removing information can have a regularization effect.

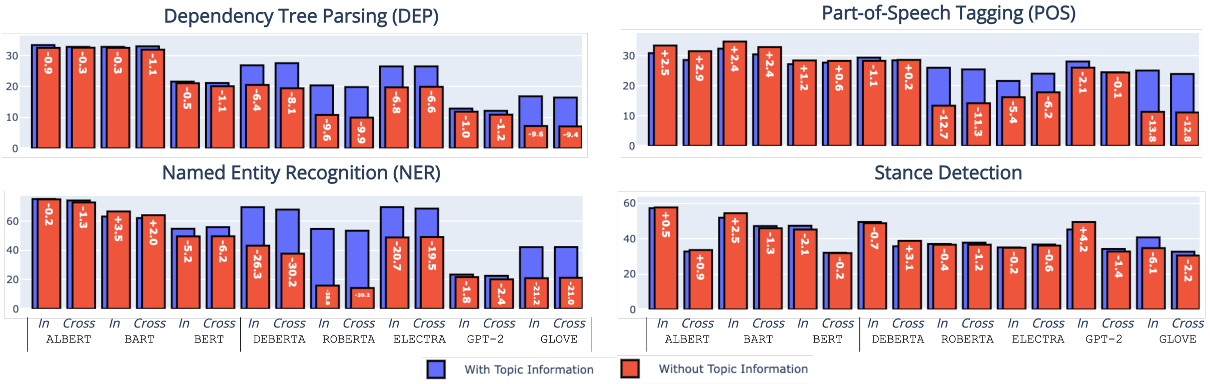

Figure 6: Comparison of the probing results with (blue bars) or without (red bars) topic-specificity for the *will-they-wont-they* dataset (Conforti et al., 2020). The white text indicates the difference between these two scenarios.

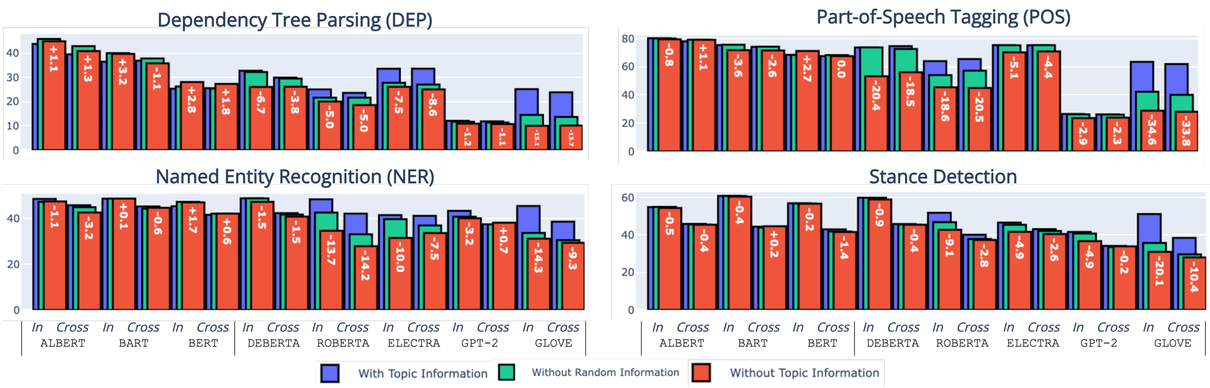

Figure 7: Comparison of the probing results with (blue bars) and without (red bars) topic information, or without random information (green bars). The white text indicates the difference between the blue and red bars.

	DEP		*POS*		*NER*	
	Random	Δ	*Random*	Δ	*Random*	Δ
ALBERT	1.4	-38.1	6.2	-39.6	3.4	-74.6
BART	1.5	-35.4	5.0	-40.3	2.9	-71.2
BERT	2.1	-23.5	9.6	-32.0	4.5	-63.0
DeBERTa	6.8	-23.1	14.0	-28.4	17.2	-57.4
RoBERTa	2.6	-21.0	10.0	-32.1	5.2	-60.3
ELECTRA	3.0	-30.6	9.8	-31.4	4.1	-71.2
GPT-2	5.8	-18.1	13.6	-25.0	11.0	-50.9

Table 11: Results of evaluating *DEP*, *POS*, and *NER* using the seven contextual PLMs (random initialized) for Cross-Topic and the difference to their pre-trained counterparts in Table 2.

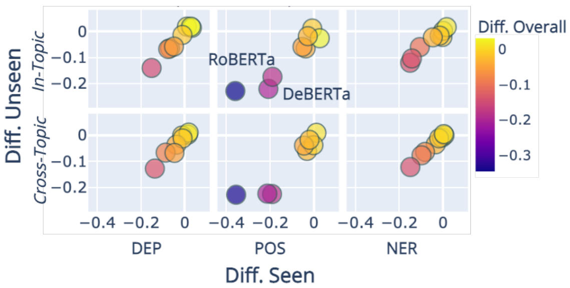

Figure 8: Performance difference for *seen* (x-axis) and *unseen* (y-axis) instances when removing topic information or not. One dot represents one PLM.

B.4 The Effect of Removing Topic Information on *Seen* and *Unseen* Instances

We show in Figure 8 that a performance drop affects *seen* and *unseen* instances for In- and Cross-Topic equally. Exceptionally, we see *unseen* ones are more affected on *POS* for DeBERTa and RoBERTa. This result indicates that these PLMs fall short of generalizing towards rare vocabularies - like *unseen* instances of *POS*.

B.5 Analysis of Per-Class Results for NER

When considering the per-class results of *NER* in Table 12, we see the classes CARDINAL, MONEY, ORG, and PERSON show the biggest differences between In- and Cross-Topic. For ORG and PERSON, we see their topic-specific terms as the main reason for the performance gap. In contrast, we were surprised about the high difference for CARDINAL. We think this is mainly because this class embodies all numbers belonging to no other class. For MONEY, we see its uneven distribution over topics as the main reason for the performance difference - one topic covers more than 50% of the instances. These entities are highly topic-specific from a statistical point of view.

		CARDINAL	DATE	GPE	MONEY	NORP	ORDINAL	ORG	PERCENT	PERSON
In	ALBERT	95.0	95.3	89.4	95.0	91.3	97.8	80.2	99.2	82.7
	BART	94.8	94.6	89.7	95.6	91.6	97.3	81.0	99.4	83.5
	DeBERTa	95.3	95.6	90.0	96.5	91.5	97.4	81.1	99.2	83.7
Cross	ALBERT	91.2	95.0	88.6	55.6	90.8	98.1	78.8	98.9	81.7
	BART	90.1	94.2	88.9	35.0	90.7	97.6	79.1	98.8	81.8
	DeBERTa	88.3	95.3	88.6	0.0	90.5	97.5	79.8	98.6	81.8

Table 12: Per-class results of ALBERT, BART, and DeBERTa on *NER* for In- and Cross-Topic.

		CARDINAL	DATE	GPE	MONEY	NORP	ORDINAL	ORG	PERCENT	PERSON
In	BART	-0.23	0.04	0.15	0.15	0.02	-0.04	0.08	-0.13	0.20
	BERT	1.65	-0.15	-0.04	28.00	-0.14	-0.58	0.06	0.00	0.22
	DEBERTA	-1.14	-0.13	-1.48	-7.74	-14.40	-0.30	-0.82	-0.12	-0.10
	ROBERTA	-6.00	-3.00	-7.82	-24.09	-90.61	-98.06	-2.66	-0.51	-0.58
Cross	BART	-0.48	0.01	-0.13	2.45	-0.06	-0.52	-0.38	-0.09	-0.03
	BERT	-0.05	-0.05	1.00	0.00	8.95	-0.60	0.29	0.00	0.00
	DEBERTA	-0.07	-0.16	-2.52	0.00	-21.88	-0.35	-0.91	-0.01	0.07
	ROBERTA	-9.04	-2.63	-7.45	0.00	-85.23	-98.07	-2.99	-35.97	-0.46

Table 13: Class-wise effect on the performance when removing topic information of BART, BERT, DeBERTa, and RoBERTa on NER for In- and Cross-Topic.

Despite having almost the same performance for In-Topic, BART and DeBERTa tend to outperform ALBERT on classes with more semantic complexities - like GPE, ORG or PERSON. For Cross-Topic, we see ALBERT performing better in classes unevenly distributed instances over topics - like MONEY. Further, it outperforms BART and DeBERTa on less semantical classes (CARDINAL, ORDINAL, PERCENT).

B.6 Effect of Removing Token-Level Topic Information of Per-Class Results for NER

Similar to the previous analysis, there are apparent effects of removing topic information when considering NER classes separately. Table 13 shows these results for BART, BERT, DeBERTa, and RoBERTa. Like the overall result, BART, DeBERTa, and RoBERTa perform less when removing topic information. Whereby the effect is the most pronounced for RoBERTa with the highest performance drop for In- and Cross-Topic on classes like NORP or ORDINAL. In addition, these results show that the performance gain from removing topic information within BERT happens on MONEY for In-Topic and NORP for Cross-Topic.

B.7 The Effect of Fine-Tuning on NER Classes

Analysing the results (Table B.7) for every NER class gives additional insights into where the fine-tuning had the most significant effect. We generally see the biggest effect on classes with less semantic meaning, like ORDINAL, PERCENT, or MONEY. At the same time, GPE, PERSON, and ORG are

		CARDINAL	DATE	GPE	MONEY	NORP	ORDINAL	ORG	PERCENT	PERSON
In	ALBERT	-34.2	-25.4	-26.9	-95.0	-51.9	-60.3	-22.4	-99.2	-21.8
	BART	-8.5	-7.2	-7.5	-7.2	-10.4	-36.6	-4.1	-3.8	-2.7
	BERT	-1.9	-2.0	-2.0	34.8	-4.4	-17.9	-0.8	-3.9	-1.1
	DEBERTA	-15.1	-6.8	-8.7	-19.5	-43.7	-60.8	-8.8	-24.8	-8.3
Cross	ALBERT	-21.5	-10.4	-19.1	-55.6	-34.4	-13.1	-10.7	-81.0	-9.2
	BART	-9.2	-7.4	-7.0	-16.3	-11.2	-24.4	-3.9	-4.5	-2.1
	BERT	-2.5	-1.2	-1.2	3.6	-2.2	-9.7	-0.8	-2.6	-0.5
	DEBERTA	-18.2	-6.2	-12.7	0.0	-50.6	-76.0	-11.7	-73.5	-6.8

Table 14: Per-class difference before and after fine-tuning on *stance detection* of ALBERT, BART, BERT, and DeBERTa on NER for In- and Cross-Topic.

less affected as classes with more attached semantics. Regarding the different PLMs, ALBERT and DeBERTa show the most performance training, while BERT gains performance for the MONEY class.

	DEP		***POS***		***NER***	
	In	*Cross*	*In*	*Cross*	*In*	*Cross*
ALBERT	85.2	83.9	93.8	93.6	86.9	85.0
BART	80.9	81.0	92.6	92.0	87.1	84.5
BERT	76.1	76.1	89.2	88.6	85.2	82.9
DeBERTa	81.2	79.9	92.8	93.1	87.5	84.0
RoBERTa	75.9	75.5	89.6	90.1	86.3	83.2
ELECTRA	81.1	80.7	92.3	92.2	82.8	82.2
GPT-2	69.8	69.1	85.8	85.7	84.6	81.1
GloVe	39.5	38.5	46.6	45.9	78.8	77.2
Average	73.7	73.1	85.3	85.2	84.9	82.5
BERT (Tenney et al., 2019b)	93.0		97.0		96.1	
BERT (Tenney et al., 2019a)	95.2		96.5		96.0	
BERT (Hewitt and Liang, 2019)	89.0		97.2		-	

Table 15: Accuracy results for In- and Cross-Topic probing results for eight PLMs, across three random seeds.

B.8 Annotation Verification

To evaluate probing tasks in the In- and Cross-Topic setup, we rely on data with topic annotations on the instance level - like the *UKP ArgMin* (Stab et al., 2018) or the *wtwt* (Conforti et al., 2020) dataset. Since these datasets do not include linguistic annotations, we rely on spaCy to automatically derive the labels for *dependency tree parsing (DEP)*, *part-of-speech tagging (POS)*, or *named entity recognition (NER)*. We used the `en_core_web_sm` model, which provides reliable labels with an accuracy of 97.0 for *POS*, 90.0-92.0 for *DEP*, and an F1 score of 85.0 for *NER* (details available online).In addition, we see our results (§ 4) well aligned (DEP $<$ NER $<$ POS) with previous work (Tenney et al., 2019b), even though we mainly report F_1 score. This finding is also supported by considering the accuracy evaluation (Table 15), which corresponds to previous results. Note that we can expect a generally lower performance level since we trained the probes on fewer instances than related work.