

Reproducing the Results, but not the Interpretations: Evidence for Artifacts in Automatic Creativity Scoring with LLMs

Carl Hermany* and Omar Momen* and Sina Zarriß

Computational Linguistics Group, Bielefeld University, Germany

{carl.hermany, omar.hassan, sina.zarriess}@uni-bielefeld.de

Abstract

Recent work has shown that finetuned language models can predict human creativity ratings of metaphors, motivating claims that they can comprehend creativity in non-literal language. We question the interpretation of a previously reported performance of finetuned GPT-2 and RoBERTa and investigate the signals underlying their predictions. Using ridge regression probes, random baselines, token-order ablations, and representation-evolution features, we find that pretrained representations perform comparably to finetuned models. Randomly initialized models retain moderate performance when token and positional information is preserved, and token shuffling does not eliminate the predictive signal. These results suggest that performance is supported largely by surface-level cues, including sentence length, token identity, and token co-occurrence. We therefore reproduce the original results but challenge the interpretation, highlighting the importance of controlled reproducibility studies on LM-related findings and claims from other domains.¹

1 Introduction

Language Models (LMs) have become widely used across research domains and for a variety of purposes (Laskar et al., 2024). One emerging application is the automatic assessment of creativity. Researchers in psychology and creativity research are exploring computational methods that can reduce reliance on the labor-intensive and potentially subjective process of obtaining human creativity ratings. LMs have become a prominent approach in this line of research (Goecke et al., 2024; Bunt et al., 2025; Zarriß et al., 2025).

Recently, DiStefano et al. (2024) investigated whether supervised finetuning could enable LMs

to predict human creativity ratings for human-produced metaphorical sentences. Their experimental setting was designed to resemble a quantifiable creativity task. Participants were asked to produce short metaphorical descriptions in response to prompts representing everyday experiences, such as a “boring class”. Independent human raters then evaluated the creativity of the resulting metaphors. This dataset enabled the authors to assess how closely automatic scoring methods approximated human judgments. Finetuned GPT-2 and RoBERTa achieved the strongest correlations with human ratings among the evaluated methods. The authors interpret these findings as evidence that finetuned LMs offer a promising alternative to human ratings and may learn to understand and evaluate creativity in complex, non-literal language (Organisciak et al., 2023; DiStefano et al., 2024).

From an interpretability perspective, such findings warrant further investigation. When research draws conclusions about LMs’ capabilities, performance-based evidence should generally be complemented by analyses of LMs’ inner workings and the computational signals underlying that performance. This holds in particular for finetuned models, where strong correlations with human ratings do not by themselves reveal which properties of the input the models use, whether they encode generalizable aspects of creativity, or whether their predictions depend on simpler correlates, superficial cues or artifacts in the data that the model picked up during finetuning. We therefore analyze the finetuned models released by DiStefano et al. (2024) using layer-wise representational analyses, diagnostic baselines, and controlled ablation experiments. We aim to characterize the signals that support the creativity predictions by these models.

In this paper, we fit simple linear regression models on different representations extracted from the hidden layers of finetuned and pretrained GPT-2 and RoBERTa checkpoints, and evaluate the corre-

*Equal contribution.

¹Code: https://github.com/OmarMomen14/interpreting_llms_scoring_creativity/

lations of the predictions to human ratings. Then, we reproduce the same experiment with randomly initialized GPT-2 and RoBERTa, as well as positional embeddings alone. Furthermore, we investigate the effect of shuffling the order of input tokens at test time. Additionally, we fit linear regressors on hand-picked features that track the change of token hidden representations across layers. Our results reveal that hidden representations of both pretrained and finetuned LMs encode features that can predict human creativity ratings with Pearson correlations r in the range of 0.6–0.7. At the same time, our additional analyses on these results suggest that the learnt features by these finetuned and pretrained models are mostly surface features related to the length of the input sentence and token identities and co-occurrences. Thereby, we reproduce DiStefano et al. (2024)’s result that LMs finetuned to predict creativity ratings can achieve decent correlations with human ratings, but our additional analyses do not support DiStefano et al. (2024)’s interpretation that LMs are capable of “understanding” creativity in complex language.

2 Related Work

Probing classifiers are used to predict certain properties (e.g., semantic (Yaghoobzadeh et al., 2019) or phonetic (English et al., 2023)) based on the internal representations of another model, in an attempt to analyze the properties embedded in the representations (Köhn, 2015; Gupta et al., 2015). Although limited in their interpretability (Belinkov, 2022), probing classifiers can still offer interesting insights into information retrievable from the internals of LMs. On the specific topic of metaphors, Aghazadeh et al. (2022) train probing classifiers on internal representations of pretrained LMs to detect metaphors in different datasets and across multiple languages. They find that the internal layers encode information that enables the detection of metaphors, with middle layers performing best. Given that not all metaphors are creative, our work extends this research by looking at creativity-related aspects of metaphors, i.e., novelty/creativity scores. With these probes producing strong results in the domain of abstract concepts such as creativity ratings, the question arises whether the representations encode the concept or whether the classifiers merely utilise proxy information correlated with the ratings, e.g., sentence length. As classically shown by Gururangan et al. (2018), LMs often rely on

specific, surface-level artifacts present in the data to increase their performance.

3 Methods

In this Section, we present our experimental methodology. We describe the dataset used in the experiments and illustrate some examples. We present and justify the linear regression models that we use in our analysis, and report their technical details. We also describe our random baselines, hand-picked features and token-order ablation experiments.

3.1 Dataset

The dataset used for this paper is provided by DiStefano et al. (2024). It contains 4,589 samples of human-created metaphors to four different prompts (“boring class”, “gross food”, “bad movie”, “messy room”) by 1,546 participants. Those metaphors were evaluated by trained raters based on their perceived creativity on a scale of 1 (not creative) to 5 (very creative). The ratings for each metaphor were then averaged and z-scored. Examples are shown in Table 1.

To ensure comparability between their results and our approach, we use the same dataset splits (train, test, validation and held-out), with the held-out split exclusively containing the metaphors of the prompt “messy room”, to test if the learned metaphor creativity scoring transfers to unseen prompts.

Prompt	Metaphor	Score
<i>gross food</i>	“Eating a nasty food is like punching your tongue in the taste buds and telling it you hate it for working”	4.12
<i>boring class</i>	“Math class is like a sleep study, the teacher observes you as you slowly fall asleep.”	0.54
<i>messy room</i>	“Living in that room was like a pig pen full of mud and flies; it was so disgusting”	-1.64

Table 1: Examples of metaphorical sentences produced by human participants given specific prompts, together with averaged and z-scored human ratings. We pick examples corresponding to high, medium and low scores.

3.2 Models

We conduct our analyses using the same two LMs as DiStefano et al. (2024), GPT-2 (Radford et al., 2019) and RoBERTa (Liu et al., 2019),

specifically their approximately 125M-parameter variants. The finetuned checkpoints released by the original study allow us to reproduce its reported results and to analyse the hidden representations of the finetuned models. We additionally apply our methods to the corresponding pretrained checkpoints obtained from the Hugging Face Hub (openai-community/gpt2 and FacebookAI/roberta-base).

3.3 Linear Regression Probes

To figure out whether the fairly good predictions of creativity ratings by the finetuned models are linearly recoverable from model-internal representations, we fit simple linear regression models on top of those representations and evaluate the fitted models’ predictions on test and held-out splits. Following Belinkov (2022), we interpret successful regression performance as evidence that information related to human creativity ratings is extractable from the models’ hidden representations, rather than as evidence that the models necessarily use this information during prediction.

Deviating from the method of DiStefano et al. (2024), we don’t prepend context (*A [adjective] metaphor for [topic] is:*) to the metaphors when embedding. Further details regarding preprocessing can be found in Appendix A.4.

3.3.1 Ridge Regression

We use ridge regression, which extends ordinary least squares with an (L_2) penalty that shrinks the regression coefficients (Hoerl and Kennard, 1970). This regularization is suitable for our 768-dimensional LM representations, whose dimensions may be highly correlated, as it reduces overfitting and produces more stable coefficient estimates. We select the regularisation strength separately for each layer through an exhaustive search over 24 logarithmically spaced (α) values between (10^{-5}) and (10^6). Each value is evaluated using shuffled five-fold cross-validation on the training split, with mean Pearson correlation as the selection criterion.

3.3.2 Representations and Pooling

We extract different representations from the finetuned and pretrained model variants, and we test with different pooling methods to prepare the input features for the regression models.

Hidden embeddings We perform layer-wise analysis on the four finetuned and pretrained model

variants. For all 13 layers of each model (1 uncontextualized embedding layer followed by 12 hidden layers), we obtain one single embedded representation for each metaphorical sentence. We experiment with three different methods to compress a metaphorical sentence representation into a single embedding. **1. Averaging:** We average the value for each dimension over all representations of the input content tokens. This results in 768-dimensional representations of each metaphor. **2. Average & Standard Deviation:** We combine the average value per dimension with the standard deviation per dimension, resulting in 1536-dimensional representations. **3. Last- / CLS-Token:** For the GPT-2 embeddings, we use the representation of the last token per metaphor sentence, while for RoBERTa we use the designated classification token (`<cls>`). Both techniques result in 768-dimensional representations.

Positional embeddings Sentence length is one of the surface features that are found to be significantly correlated with the metaphor creativity ratings in (DiStefano et al., 2024). And we hypothesize that both the finetuned models and the fitted regression models are utilizing such surface features to successfully predict the human ratings. To examine this hypothesis, we fit a linear regression model on the encoded positional embeddings of the metaphorical sentences to predict creativity ratings. We apply this method to both the finetuned and pretrained variants of GPT-2 and RoBERTa.

Random baselines To verify whether the linear regression models are learning meaningful features for predicting the creativity ratings, we experiment with three random baselines for both GPT-2 and RoBERTa. **R1:** Fitting regression models on hidden representations of the LM with randomly initialised parameters. **R2:** Same as R1, and additionally zero-out the position encoding. **R3:** Same as R2, but the actual input sequences are replaced with randomly sampled token IDs from the vocabulary.

LLM embeddings To compare the results reproduced from DiStefano et al. (2024)’s study to more recent state-of-the-art LLMs, we fit the same linear regression models to the embeddings of a SOTA model, namely OpenAI’s text-embedding-3-large (OpenAI, 2025). With an embedding dimension of 3072 and a maximum context size of 8192 tokens, it surpasses the capabilities of the smaller GPT-2 and RoBERTa models

by far.

Evolution of embeddings across layers: To examine whether the pretrained and finetuned LMs transform the input tokens across their layers in a way that can predict the creativity of an input metaphoric sentence, we experiment with fitting linear regression models on features that track the change of representations across layers. We test the following five features, on their own and within all possible combinations respectively (31 combinations in total): Token drift, Sentence drift, Norm growth, Euclidean Distance and Distance to final representation. For descriptions of these features, see Appendix (A.1).

3.4 Token-order Ablation

We examine whether the finetuned models and the regression probes rely on the original token order and, consequently, on syntactic and semantic information. At evaluation time, we randomly permute the input token IDs before passing each sentence to the LM. This preserves the set of tokens and the sequence length while disrupting the sentence’s natural order. A substantial decrease in prediction performance would indicate that the models rely on order-sensitive information. Conversely, performance close to that obtained on the original sentences would suggest that the predictions can be supported largely by order-invariant cues, such as token identity, co-occurrence, and sequence length.

4 Results

In this Section, we report the results of fitting the linear regression probes to different sets of LM representations as well as the token order ablation results. We evaluate models based on their predictions of two unseen sets of metaphorical sentences: i) a **test set** of 766 instances that are not seen during training but belong to the same prompt sets used during training, and ii) a **held-out set** of 759 instances that were neither seen during training nor belong to any prompt used during training, which tests generalizability to unseen concepts. We use Pearson correlations between the human creativity ratings and the predictions of either the finetuned models or the fitted regression models. We also remove outliers in accordance with DiStefano et al. (2024), defined as values below or above 3 standard deviations from the mean, in both the true and predicted ratings. An overview of the results can be found in Table 2 and Figures 1 and 2.

Model	Test r	Held-out r
Previously Reported Results DiStefano et al. (2024)		
GPT-2 _{ft}	0.70	0.63
RoBERTa _{ft}	0.72	0.68
DSI	0.42	0.51
Word Count	0.49	0.47
Random Baselines		
GPT-2 - R1 [†]	0.56	0.50
GPT-2 - R2 [†]	0.50	0.39
GPT-2 - R3 [†]	0.08	0.06
RoBERTa - R1 [†]	0.60	0.53
RoBERTa - R2 [†]	0.54	0.48
RoBERTa - R3 [†]	0.18	0.11
Regression on OpenAI’s Embeddings		
Text-embedding-3-large	0.72	0.62
Regression on Hidden Representations		
GPT-2 _{ft} (Layer 11)	0.69	0.58
RoBERTa _{ft} (Layer 10)	0.72	0.63
GPT-2 _{pt} (Layer 6)	0.69	0.64
RoBERTa _{pt} (Layer 8)	0.71	0.64
Regression on Positional Embeddings		
GPT-2 _{pt}	0.53	0.51
RoBERTa _{pt}	0.53	0.51
Token Order Ablation		
GPT-2 _{ft}	0.63	0.57
RoBERTa _{ft}	0.55	0.45
GPT-2 _{pt} [†]	0.65	0.57
RoBERTa _{pt} [†]	0.63	0.55

Table 2: Pearson correlations between predicted and human creativity ratings on the test and held-out splits. Layer and pooling selection based on test- r . GPT-2_{ft} and RoBERTa_{ft} are the checkpoints finetuned on creativity ratings. GPT-2_{pt} and RoBERTa_{pt} are the pretrained checkpoints retrieved from Hugging Face. For models marked with a dagger[†], the reported r values are averaged across all layers. Layer-specific results are plotted in Figures 1 and 2, and reported in Tables 5, 6, 7 and 8.

4.1 Baselines

We could successfully reproduce the results of the original work by predicting the creativity ratings using the finetuned GPT-2 checkpoint (test $r = 0.7$, held-out $r = 0.63$) and the finetuned RoBERTa checkpoint (test $r = 0.72$, held-out $r = 0.68$), even without contextual prefixes (s. 3.3). The difference between test and held-out results points to incomplete generalization. The divergent semantic integration (DSI; Johnson et al. (2023)) is reported in the original work to have weak-to-moderate correlations with creativity ratings (test $r = 0.42$, held-out $r = 0.51$), while word count has slightly stronger correlations (test $r = 0.49$, held-out $r = 0.47$).

We observe that the fitted regression models on representations of random baseline R1 are yielding moderate positive correlations (average test r

$= 0.56$ and $r = 0.60$ for GPT-2 and RoBERTa, respectively). We infer that the regression models are learning patterns related to sentence length (encoded in positional encodings) and token identities and co-occurrences (encoded in the static input embeddings). Disabling the positional encoding in R2 reduces the learnable patterns in the hidden representations (average test $r = 0.50$ and $r = 0.54$ for GPT-2 and RoBERTa, respectively). Additionally, hindering patterns related to token identities in R3 significantly reduces the learnable patterns in the hidden representations (average test $r = 0.08$ and $r = 0.18$ for GPT-2 and RoBERTa, respectively).

On the other hand, an intuitive upper-bound baseline is OpenAI’s output sentence embeddings. The fitted regression model on these embeddings yields a correlation of $r = 0.72$ on the test set and $r = 0.62$ on the held-out set (see Table 2).

4.2 Regression on LM Representations

Finetuned models In Figures 1 and 2, we plot the test Pearson’s correlation r of fitted regression models over average-pooled representations of the finetuned checkpoints of GPT-2 and RoBERTa, respectively. Results of average/std. pooling and Last-/CLS token pooling are reported in Tables 5 and 6. For finetuned GPT-2, the strongest correlation is obtained from the linear regression model fit on layer 11’s pooled sentence representations (average+std.) (test $r = 0.69$, held-out $r = 0.58$). While for finetuned RoBERTa, regression on layer 10’s representations yields $r = 0.72$ (test) and $r = 0.63$ (held-out). Overall, we see the hidden representations of the finetuned RoBERTa to be encoding better features for predicting creativity that systematically improve across layers.

Pretrained models Similarly, in Figures 1 and 2, we plot the test Pearson’s correlation r of fitted regression models over average-pooled representations of the pretrained checkpoints of GPT-2 and RoBERTa, respectively. Even without any prior information about the task, we find the hidden representations of the pretrained GPT-2 and RoBERTa to be encoding strong features for predicting creativity ratings. A linear regression model fit on layer 6’s (average-pooled) sentence representations of pretrained GPT-2 yields $r = 0.69$ (test) and $r = 0.64$ (held-out). For pretrained RoBERTa, regression on layer 8’s representations (average+std. pooling) yields $r = 0.71$ (test) and $r = 0.64$ (held-out). This shows that fitting regression models on

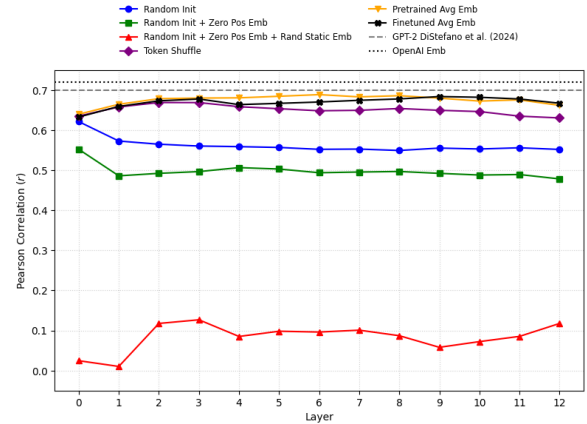


Figure 1: Pearson r correlation of predictions of the test set using different methods with GPT-2 as shown in the legend. Layer 0 is the output of the input embedding layer, and then 1-12 are the outputs of each transformer layer.

hidden representations of the pretrained variants is on par with regression on the representations of the finetuned variants, and close to the scores achieved by finetuning a full model’s weights as in (DiStefano et al., 2024).

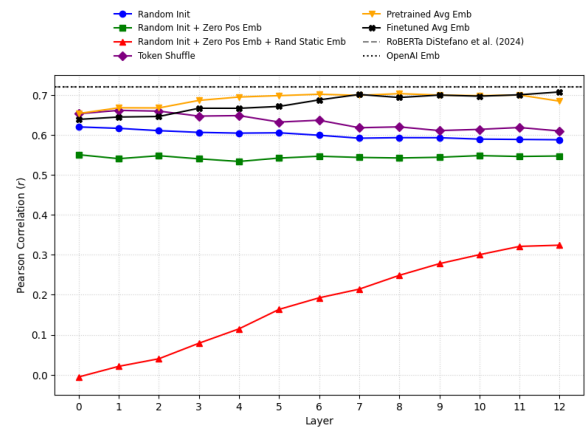


Figure 2: Pearson r correlation of predictions of the test set using different methods with RoBERTa as shown in the legend. Layer 0 is the output of the input embedding layer, and then 1-12 are the outputs of each transformer layer.

Positional embeddings Fitting a linear regression model on the (average-pooled) positional encoding of GPT-2 or RoBERTa (768 dimensions) yields a correlation of $r = 0.53$ on the test set, and $r = 0.51$ on the held-out set (see Table 2 and 4). We also see that the fitted regression on layer 0 (positional encoding + static token embedding) in each of the 4 model variants yields $r \approx 0.65$ on the test set (see Figures 1 and 2).

Feature	d	GPT-2		RoBERTa	
		PT	FT	PT	FT
1. Token Drift	36	0.56	0.57	0.54	0.57
2. Sentence Drift	12	0.55	0.57	0.45	0.57
3. Norm Growth	12	0.54	0.55	0.51	0.55
4. Euclidean Distance	12	0.55	0.53	0.48	0.53
5. Distance to Final	12	0.53	0.48	0.49	0.48
Best combination	–	1,3–5	1–5	1–5	1–4
Correlation	–	0.58	0.63	0.59	0.68

Table 3: Pearson correlations obtained using 5 embedding evolution features and their best-performing combinations. |d| denotes the feature dimensionality. PT and FT denote pretrained and finetuned models, respectively. Numbers in the best-combination row refer to the feature indices listed in the first column. Best combinations picked based on test- r .

This result suggests that encoded sentence length information in the hidden representation has a substantial contribution to the prediction of creativity ratings and that, similarly, the static non-contextual information about tokens can yield on-par results with fully contextualised embeddings. Naturally, the positional encoding and the static token encoding are incorporated in the subsequent hidden and output token representations.

Evolution of representations In Table 3, we report the results of fitting regression models on hand-picked features that track changes in layer representations of pretrained and finetuned checkpoints of GPT-2 and RoBERTa (detailed results are shown in Table 5). We find that all the individual features yield moderate positive correlations with small differences among each other. We also observe that the best correlations are achieved from the finetuned checkpoints in the case of using a combination of 4-5 features together. Fitting a linear regression model over 4 features (72 dimensions) obtained from the hidden representations of finetuned RoBERTa yields a correlation of $r = 0.68$ on the test set and $r = 0.64$ on the heldout set. The transformations of the inputs across the layers of the LMs (especially the finetuned variants) alone, therefore, seem to produce enough information to meaningfully predict the creativity ratings of the metaphors.

4.3 Token-Order Ablation

As shown in Table 2 and 7, with the shuffled inputs, the finetuned GPT-2 model achieves correlations of $r = 0.63$ (test) and $r = 0.57$ (held-out), compared with $r = 0.70$ (test) and $r = 0.63$ (held-out) for the original inputs. Finetuned RoBERTa shows a larger decrease, from $r = 0.72$ (test) and $r = 0.68$ (held-out) to $r = 0.55$ (test) and $r = 0.45$ (held-out). For the pretrained models, we keep the regression probes trained on the layers’ hidden representations of the original, correctly ordered sentences fixed and apply them to representations extracted from the shuffled inputs. Predictive performance remains substantial across all layers, where GPT-2 achieves (on average across layers) $r = 0.65$ and $r = 0.57$ on the test and held-out sets, respectively, while RoBERTa achieves $r = 0.63$ and $r = 0.55$. Overall, shuffling reduces performance but does not destroy the predictive signal in the finetuned models’ weights and the pretrained models’ hidden representations. Generally, GPT-2 is consistently less affected than RoBERTa, suggesting that its predictions rely more strongly on order-invariant cues, such as token identity and sequence length.

5 Discussion

Our results show the possibility of obtaining creativity score predictions with significant moderate-to-strong correlations from training simple linear regressors on sentence representations from hidden layers of GPT-2 and RoBERTa, without the need for extensive fine-tuning. Coming very close to the results obtained by task-specific fine-tuning of the whole models, we find that the information enabling creativity score prediction is decodable from the hidden representations of the pretrained models.

Our promising probing classification results on models’ hidden representation could suggest that even pretrained LMs learn to encode aspects of creativity in language use, without being finetuned for this complex task. However, our additional analyses point towards a very different explanation. Through our random baselines and token order ablation experiments, we find that surface feature encodings like sentence length (encoded in positional embeddings and possibly leaked in other ways) and token identities and co-occurrences (even with random but fixed static embeddings) are contributing strongly to the predictions. This is consistent with (Sanchez-Bayona and Agerri, 2025), who ob-

tain similar findings. On the basis of the discussed findings through our ablations, we argue that it is very likely that the human-generated sentences and rating data used for (DiStefano et al., 2024) contain artefacts and surface cues like sentence length that are easily picked up by any embedding or LM model. This speaks against strong claims from domains like psychology and creativity research about LMs, specifically that they are able to model creativity scoring by “understanding” complex figurative language.

Finally, we find fine-tuning to be useful in restructuring the hidden representations, such that features tracking the embedding evolution (the change of representation across layers) encode better information for predicting creativity ratings than in the case of pretrained models. Our results obtained through these embedding evolution experiments seem to contradict our interpretation regarding surface cues to a certain degree. Future work could further investigate this contradiction.

6 Conclusion

The research presented in this paper aimed to investigate recent work on metaphor creativity score prediction by LMs. Using regression probes, we show that the information necessary for such prediction is already encoded and recoverable in the hidden representations of LMs, even those not fine-tuned for the task. Through random baselines and targeted ablation experiments, we then show that this information mostly consists of surface-level features, thereby supporting our initial hypothesis, that high correlations between predicted and human creativity ratings aren’t necessarily a result of the LMs “understanding” creativity in metaphorical language.

Limitations

As stated in Section 2, probing classifiers are limited in their interpretability, and we cannot reliably say on what information exactly the fine-tuned models rely when predicting creativity scores. Although we tried to isolate candidates through targeted ablation experiments, a definitive analysis is not possible.

Furthermore, the reliance on a single dataset containing a very narrow spectrum of topics, limits the findings. Human creativity, specifically in the domain of metaphor production, is diverse; thus, future work could apply the methodology outlined

in this paper to a broader array of datasets and metaphor topics.

Acknowledgments

This work was funded by the European Regional Development Fund within the project "LLM4KMU - Optimierter Einsatz von Open Source Large Language Modellen in KMU" and by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – CRC-1646, project number 512393437, project A05.

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.
- Yonatan Belinkov. 2022. [Probing classifiers: Promises, shortcomings, and advances](#). *Computational Linguistics*, 48(1):207–219.
- Hannah L. Bunt, Alex Goddard, Tom W. W. Reader, and Alex Gillespie. 2025. [Validating the use of large language models for psychological text classification](#). *Frontiers in Social Psychology*, Volume 3 - 2025.
- Paul V. DiStefano, John D. Patterson, and Roger E. Beaty. 2024. [Automatic Scoring of Metaphor Creativity with Large Language Models](#). 37(4):555–569.
- Patrick Cormac English, John D. Kelleher, and Julie Carson-Berndsen. 2023. [Discovering Phonetic Feature Event Patterns in Transformer Embeddings](#). In *INTERSPEECH 2023*, pages 4733–4737. ISCA.
- Benjamin Goecke, Paul V. DiStefano, Wolfgang Aschauer, Kurt Haim, Roger Beaty, and Boris Forthmann. 2024. [Automated Scoring of Scientific Creativity in German](#). *The Journal of Creative Behavior*, 58(3):321–327.
- Abhijeet Gupta, Gemma Boleda, Marco Baroni, and Sebastian Padó. 2015. [Distributional vectors encode referential attributes](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 12–21, Lisbon, Portugal. Association for Computational Linguistics.
- Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. 2018. [Annotation Artifacts in Natural Language Inference Data](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*,

pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Arthur E. Hoerl and Robert W. Kennard. 1970. [Ridge regression: Biased estimation for nonorthogonal problems](#). *Technometrics*, 12(1):55–67.

Dan R. Johnson, James C. Kaufman, Brendan S. Baker, John D. Patterson, Baptiste Barbot, Adam E. Green, Janet van Hell, Evan Kennedy, Grace F. Sullivan, Christa L. Taylor, Thomas Ward, and Roger E. Beaty. 2023. [Divergent semantic integration \(DSI\): Extracting creativity from narratives with distributional semantic modeling](#). *Behavior Research Methods*, 55:3726–3759.

Arne Köhn. 2015. [What’s in an embedding? analyzing word embeddings through multilingual evaluation](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2067–2073, Lisbon, Portugal. Association for Computational Linguistics.

Md Tahmid Rahman Laskar, Sawsan Alqahtani, M Saiful Bari, Mizanur Rahman, Mohammad Abdullah Matin Khan, Haidar Khan, Israt Jahan, Amran Bhuiyan, Chee Wei Tan, Md Rizwan Parvez, Enamul Hoque, Shafiq Joty, and Jimmy Xiangji Huang. 2024. [A systematic survey and critical review on evaluating large language models: Challenges, limitations, and recommendations](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13785–13816, Miami, Florida, USA. 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](#). *Preprint*, arXiv:1907.11692.

OpenAI. 2025. [New embedding models and api updates](#), <https://openai.com/index/new-embedding-models-and-api-updates/>. Accessed: 20.07.2026.

Peter Organisciak, Selcuk Acar, Denis Dumas, and Kelly Berthiaume. 2023. [Beyond semantic distance: Automated scoring of divergent thinking greatly improves with large language models](#). *Thinking Skills and Creativity*, 49:101356.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. [Language models are unsupervised multitask learners](#). *OpenAI*.

Elisa Sanchez-Bayona and Rodrigo Agerri. 2025. [Metaphor and large language models: When surface features matter more than deep understanding](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 17462–17477, Vienna, Austria. Association for Computational Linguistics.

Yadollah Yaghoobzadeh, Katharina Kann, T. J. Hazen, Eneko Agirre, and Hinrich Schütze. 2019. [Probing](#)

[for semantic classes: Diagnosing the meaning content of word embeddings](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5740–5753, Florence, Italy. Association for Computational Linguistics.

Sina Zarrieß, Simeon Junker, Judith Sieker, and Özge Alacam. 2025. [Components of Creativity: Language Model-based Predictors for Clustering and Switching in Verbal Fluency](#). In *Proceedings of the 29th Conference on Computational Natural Language Learning*, pages 216–232, Vienna, Austria. Association for Computational Linguistics.

A Appendix

A.1 Embeddings Evolution Features

We construct multiple different features that track the evolution of the hidden representations across the model layers. The results of training linear models on those features are reported in Table 3.

1. **Token-wise contextualization drift.** We calculate the directional change d_t of each token representation h_t across two consecutive ℓ layers as $d_t^{(\ell)} = 1 - \cos(h_t^{(\ell-1)}, h_t^{(\ell)})$ and summarize its distribution across the sentence using the mean, standard deviation, and maximum token drift. This method obtains 36 features as input for the regression model: 12 layer-pairs x 3 estimates.
2. **Mean-pooled sentence drift.** We quantify the directional change in the mean-pooled sentence representation m_i between consecutive layers ℓ as $d_i^{(\ell)} = 1 - \cos(m_i^{(\ell-1)}, m_i^{(\ell)})$, where larger values indicate stronger redirection of the representation by layer ℓ . This method obtains 12 features as input for the regression model: 12 layer-pairs x 1 estimate.
3. **Norm growth or contraction.** We track changes in a sentence’s representation magnitude $\|m_i\|_2$ using the log norm change $g_i^{(\ell)} = \log(\|m_i^{(\ell)}\|_2 + \epsilon) - \log(\|m_i^{(\ell-1)}\|_2 + \epsilon)$, where positive and negative values indicate growth and contraction, respectively. This method obtains 12 features as input for the regression model: 12 layer-pairs x 1 estimate.
4. **Relative Euclidean update size.** We measure the magnitude of the layer update relative to the incoming representation as $r_i^{(\ell)} = \|\Delta_i^{(\ell)}\|_2 / (\|m_i^{(\ell-1)}\|_2 + \epsilon)$, thereby capturing both directional and magnitude changes.

This method obtains 12 features as input for the regression model: 12 layer-pairs x 1 estimate.

5. **Convergence toward the final representation.** We measure how far each intermediate mean-pooled representation remains from the final-layer representation as $c_i^{(\ell)} = 1 - \cos(m_i^{(\ell)}, m_i^{(L)})$, producing a layer-wise trajectory of convergence toward the model’s final representation. This method obtains 12 features as input for the regression model: 12 layer-pairs x 1 estimate.

A.2 Detailed Results

The detailed results of all experiments are reported in Table 4.

A.3 Difference between Pooling methods

Figures 3-6 illustrate the results of training linear regressors on the embeddings obtained using the three different pooling methods: using the last (GPT-2), respectively the CLS-token (RoBERTa), averaging all embeddings in the input, and taking the average together with the standard deviation. Detailed results are also reported in Table 5 and 6.

A.4 Tokenization Details

For most of our experiments, tokenization was done deviating from DiStefano et al. (2024), with no padding and using truncation to a maximum length of 256 tokens. In the case of experiments using the RoBERTa model, the metaphors were wrapped in the special tokens <s> and </s>. However, when using pooling methods based on the mean, those special tokens were omitted. Further deviating from DiStefano et al. (2024), we did not prepend a contextual prefix (“A [adjective] metaphor for [topic] is:”) to the metaphors. The token-order ablation experiments are an exception to these tokenization procedures. For those experiments, we padded the input and, in the case of the finetuned models, used prefixes with the best-performing adjectives as reported by DiStefano et al. (2024) (“novel” for GPT-2, “surprising” for RoBERTa). However, all special tokens, padding tokens and elicitation prompts were kept fixed in their original positions. Only the pure metaphor tokens were isolated using boolean masks and then shuffled.

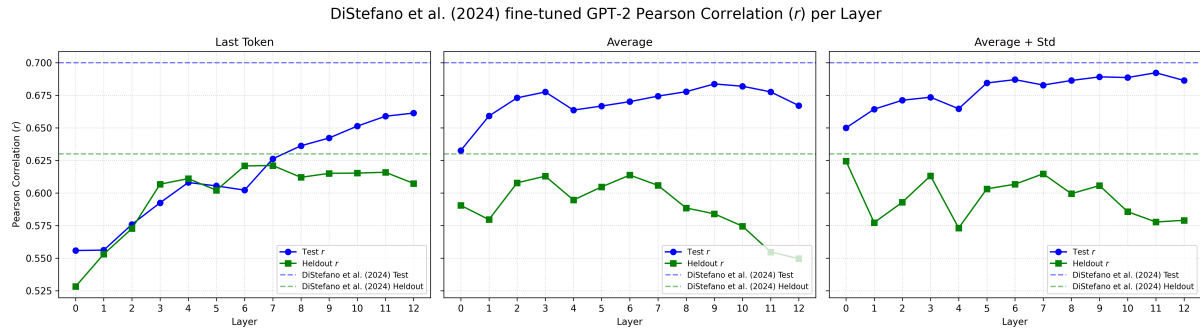


Figure 3: Per-layer Pearson Correlation between Ridge Regressor predictions based on the three different DiStefano et al. (2024) finetuned GPT-2 embeddings and human-rated creativity. DiStefano et al. (2024) fine-tuning results as reference.

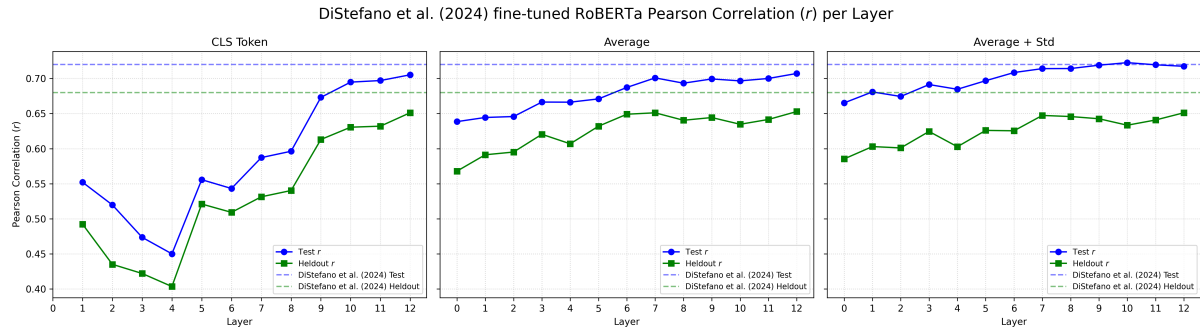


Figure 4: Per-layer Pearson Correlation between Ridge Regressor predictions based on the three different DiStefano et al. (2024) finetuned RoBERTa embeddings and human-rated creativity. DiStefano et al. (2024) fine-tuning results as reference.

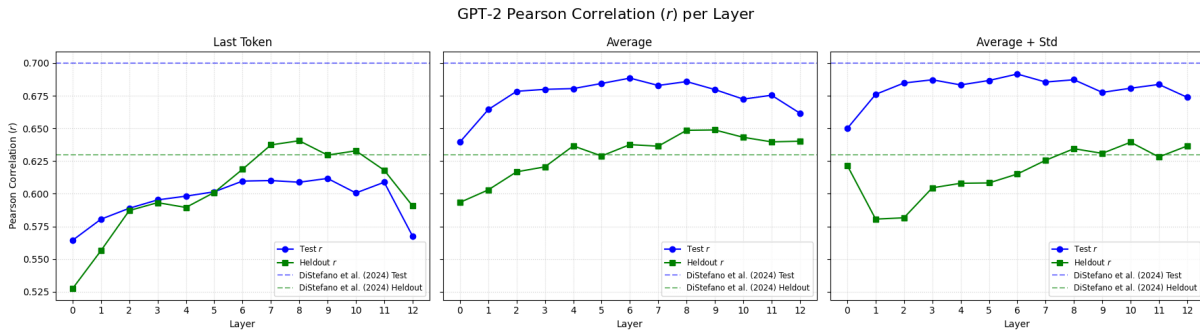


Figure 5: Per Layer Pearson Correlation between Ridge Regressor predictions based on the three different pretrained RoBERTa embeddings and human-rated creativity. DiStefano et al. (2024) fine-tuning results as reference.

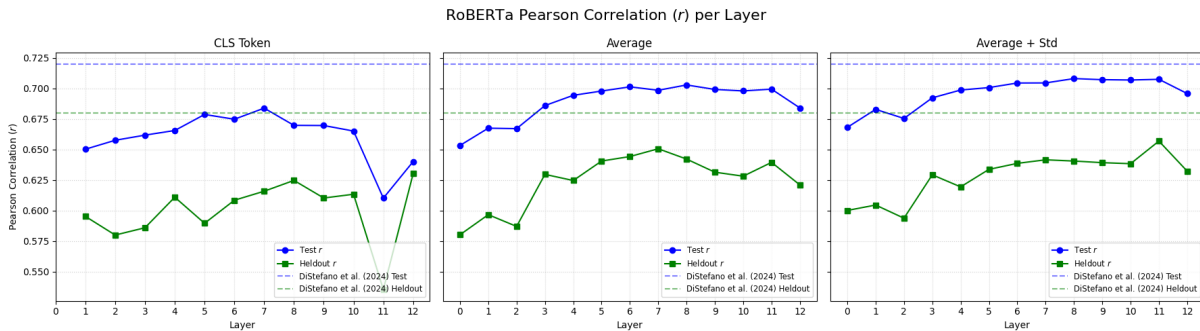


Figure 6: Per Layer Pearson Correlation between Ridge Regressor predictions based on the three different pretrained GPT-2 embeddings and human-rated creativity. DiStefano et al. (2024) fine-tuning results as reference.

Model		Layer	Test Split			Held-out Split		
			r	95% CI	Removed	r	95% CI	Removed
Previously Reported Results DiStefano et al. (2024)								
GPT-2 _{ft}		–	0.70	[0.66, 0.73]	2	0.63	[0.59, 0.67]	1
RoBERTa _{ft}		–	0.72	[0.69, 0.76]	1	0.68	[0.63, 0.71]	1
DSI		–	0.42	[0.36, 0.47]		0.51	[0.46, 0.56]	
Word Count		–	0.49	[0.43, 0.54]		0.47	[0.42, 0.53]	
Regression on OpenAI’s Embeddings								
text-embedding-3-large		–	0.72	[0.68, 0.75]	2	0.62	[0.57, 0.66]	0
Regression on Hidden Representations								
GPT-2 _{ft}	Last Token	12	0.66	[0.62, 0.70]	3	0.61	[0.56, 0.65]	0
	Average	9	0.68	[0.64, 0.72]	3	0.58	[0.53, 0.63]	4
	Average + Std	11	0.69	[0.65, 0.73]	2	0.58	[0.53, 0.62]	3
RoBERTa _{ft}	CLS Token	12	0.71	[0.67, 0.74]	1	0.65	[0.61, 0.69]	1
	Average	12	0.71	[0.67, 0.74]	1	0.65	[0.61, 0.69]	1
	Average + Std	10	0.72	[0.69, 0.75]	1	0.63	[0.59, 0.67]	1
GPT-2 _{pt}	Last Token	9	0.61	[0.56, 0.64]	2	0.63	[0.58, 0.67]	4
	Average	6	0.69	[0.65, 0.72]	1	0.64	[0.59, 0.68]	2
	Average + Std	6	0.69	[0.65, 0.73]	1	0.62	[0.57, 0.66]	1
RoBERTa _{pt}	CLS Token	7	0.68	[0.64, 0.72]	3	0.62	[0.57, 0.66]	2
	Average	8	0.70	[0.67, 0.74]	1	0.64	[0.60, 0.68]	1
	Average + Std	8	0.71	[0.67, 0.74]	1	0.64	[0.60, 0.68]	1
Regression on Positional Embeddings								
GPT-2 _{pt}		–	0.53	[0.47, 0.58]	1	0.51	[0.45, 0.56]	0
RoBERTa _{pt}		–	0.53	[0.47, 0.58]	1	0.51	[0.45, 0.56]	1
Regression on Embeddings Evolution								
GPT-2 _{pt}	1. Token Drift	–	0.56	[0.51, 0.61]	3	0.47	[0.42, 0.53]	8
	2. Sentence Drift	–	0.55	[0.50, 0.60]	1	0.49	[0.44, 0.55]	2
	3. Norm Growth	–	0.54	[0.49, 0.59]	2	0.51	[0.46, 0.56]	1
	4. Euclidean Distance	–	0.55	[0.50, 0.60]	3	0.52	[0.46, 0.57]	1
	5. Distance to Final Layer	–	0.53	[0.48, 0.58]	1	0.50	[0.44, 0.55]	0
	Best Combination	–	0.58	[0.54, 0.63]	4	0.51	[0.45, 0.56]	1
RoBERTa _{pt}	1. Token Drift	–	0.54	[0.49, 0.59]	3	0.52	[0.47, 0.57]	2
	2. Sentence Drift	–	0.45	[0.39, 0.51]	5	0.36	[0.30, 0.42]	9
	3. Norm Growth	–	0.51	[0.46, 0.56]	2	0.45	[0.39, 0.51]	8
	4. Euclidean Distance	–	0.48	[0.42, 0.53]	6	0.38	[0.31, 0.44]	11
	5. Distance to Final Layer	–	0.49	[0.43, 0.54]	3	0.40	[0.33, 0.45]	3
	Best Combination	–	0.59	[0.54, 0.63]	1	0.53	[0.47, 0.57]	5
Token Order Ablation for Finetuned and Pretrained models								
GPT-2 _{ft}		–	0.63	[0.58, 0.67]	1	0.57	[0.52, 0.62]	1
RoBERTa _{ft}		–	0.55	[0.50, 0.60]	5	0.45	[0.39, 0.50]	3
GPT-2 _{pt} (average across 12 layers)		–	0.65	–	–	0.57	–	–
RoBERTa _{pt} (average across 12 layers)		–	0.63	–	–	0.55	–	–

Table 4: Pearson correlations between predicted and human creativity ratings for all employed methodologies. Zero-based indexing for the layers. Reported layers selected based on test- r .

Layer	Last / CLS Token				Average				Average + Std			
	GPT-2 _{ft}	GPT-2 _{pt}	RoBERTa _{ft}	RoBERTa _{pt}	GPT-2 _{ft}	GPT-2 _{pt}	RoBERTa _{ft}	RoBERTa _{pt}	GPT-2 _{ft}	GPT-2 _{pt}	RoBERTa _{ft}	RoBERTa _{pt}
0	0.56 [0.51, 0.61] (1)	0.56 [0.50, 0.60] (1)	—	—	0.64 [0.60, 0.68] (2)	0.63 [0.59, 0.67] (3)	0.65 [0.61, 0.69] (2)	0.64 [0.59, 0.68] (2)	0.65 [0.61, 0.69] (1)	0.65 [0.61, 0.69] (1)	0.67 [0.63, 0.71] (1)	0.67 [0.62, 0.70] (2)
1	0.58 [0.53, 0.63] (2)	0.56 [0.51, 0.60] (3)	0.65 [0.61, 0.69] (1)	0.55 [0.50, 0.60] (6)	0.66 [0.62, 0.70] (3)	0.66 [0.62, 0.70] (2)	0.67 [0.63, 0.71] (2)	0.64 [0.60, 0.68] (2)	0.68 [0.64, 0.71] (1)	0.66 [0.62, 0.70] (2)	0.68 [0.64, 0.72] (1)	0.68 [0.64, 0.72] (1)
2	0.59 [0.54, 0.63] (2)	0.58 [0.53, 0.62] (1)	0.66 [0.62, 0.70] (1)	0.52 [0.47, 0.57] (7)	0.68 [0.64, 0.71] (3)	0.67 [0.63, 0.71] (1)	0.67 [0.63, 0.70] (1)	0.65 [0.60, 0.69] (1)	0.68 [0.65, 0.72] (1)	0.67 [0.63, 0.71] (1)	0.68 [0.65, 0.71] (1)	0.67 [0.63, 0.71] (1)
3	0.60 [0.55, 0.64] (1)	0.59 [0.54, 0.64] (3)	0.66 [0.62, 0.70] (1)	0.47 [0.42, 0.53] (6)	0.68 [0.64, 0.71] (3)	0.68 [0.64, 0.71] (2)	0.69 [0.65, 0.72] (1)	0.67 [0.63, 0.70] (1)	0.69 [0.65, 0.72] (1)	0.67 [0.63, 0.71] (2)	0.69 [0.65, 0.73] (1)	0.69 [0.65, 0.73] (1)
4	0.60 [0.55, 0.64] (2)	0.61 [0.56, 0.65] (1)	0.67 [0.62, 0.70] (1)	0.45 [0.39, 0.51] (7)	0.68 [0.64, 0.71] (1)	0.66 [0.62, 0.70] (3)	0.69 [0.66, 0.73] (1)	0.67 [0.62, 0.70] (1)	0.68 [0.64, 0.72] (1)	0.66 [0.62, 0.70] (2)	0.70 [0.66, 0.73] (1)	0.68 [0.65, 0.72] (1)
5	0.60 [0.55, 0.64] (1)	0.61 [0.56, 0.65] (1)	0.68 [0.64, 0.72] (2)	0.56 [0.50, 0.60] (3)	0.68 [0.64, 0.71] (1)	0.67 [0.63, 0.70] (2)	0.70 [0.66, 0.73] (1)	0.67 [0.63, 0.71] (1)	0.69 [0.65, 0.72] (1)	0.68 [0.64, 0.72] (2)	0.70 [0.66, 0.73] (1)	0.70 [0.66, 0.73] (2)
6	0.61 [0.56, 0.65] (2)	0.60 [0.56, 0.65] (1)	0.67 [0.63, 0.71] (3)	0.54 [0.49, 0.59] (6)	0.69 [0.65, 0.72] (1)	0.67 [0.63, 0.71] (2)	0.70 [0.66, 0.74] (1)	0.69 [0.65, 0.72] (1)	0.69 [0.65, 0.73] (1)	0.69 [0.65, 0.72] (2)	0.70 [0.67, 0.74] (1)	0.71 [0.67, 0.74] (1)
7	0.61 [0.56, 0.65] (3)	0.63 [0.58, 0.67] (1)	0.68 [0.64, 0.72] (3)	0.59 [0.54, 0.63] (8)	0.68 [0.64, 0.72] (1)	0.67 [0.63, 0.71] (3)	0.70 [0.66, 0.73] (1)	0.70 [0.66, 0.74] (1)	0.69 [0.65, 0.72] (1)	0.68 [0.64, 0.72] (3)	0.70 [0.67, 0.74] (1)	0.71 [0.68, 0.75] (1)
8	0.61 [0.56, 0.65] (2)	0.64 [0.59, 0.68] (1)	0.67 [0.63, 0.71] (4)	0.60 [0.55, 0.64] (7)	0.69 [0.65, 0.72] (1)	0.68 [0.64, 0.71] (3)	0.70 [0.67, 0.74] (1)	0.69 [0.65, 0.73] (1)	0.69 [0.65, 0.72] (1)	0.69 [0.65, 0.72] (2)	0.71 [0.67, 0.74] (1)	0.71 [0.68, 0.75] (1)
9	0.61 [0.57, 0.65] (2)	0.64 [0.60, 0.68] (1)	0.67 [0.63, 0.71] (4)	0.67 [0.63, 0.71] (4)	0.68 [0.64, 0.72] (1)	0.68 [0.64, 0.72] (3)	0.70 [0.66, 0.73] (1)	0.70 [0.66, 0.73] (2)	0.68 [0.64, 0.71] (2)	0.69 [0.65, 0.72] (3)	0.71 [0.67, 0.74] (1)	0.72 [0.68, 0.75] (1)
10	0.60 [0.55, 0.64] (2)	0.65 [0.61, 0.69] (2)	0.67 [0.62, 0.70] (1)	0.69 [0.66, 0.73] (1)	0.67 [0.63, 0.71] (3)	0.68 [0.64, 0.72] (4)	0.70 [0.66, 0.73] (1)	0.70 [0.66, 0.73] (2)	0.68 [0.64, 0.72] (2)	0.69 [0.65, 0.72] (3)	0.71 [0.67, 0.74] (1)	0.72 [0.69, 0.75] (1)
11	0.61 [0.56, 0.65] (1)	0.66 [0.62, 0.70] (3)	0.61 [0.56, 0.65] (5)	0.70 [0.66, 0.73] (2)	0.68 [0.63, 0.71] (3)	0.68 [0.64, 0.71] (4)	0.70 [0.66, 0.73] (1)	0.70 [0.66, 0.73] (1)	0.68 [0.64, 0.72] (2)	0.69 [0.65, 0.73] (2)	0.71 [0.67, 0.74] (1)	0.72 [0.68, 0.75] (2)
12	0.57 [0.52, 0.61] (2)	0.66 [0.62, 0.70] (3)	0.64 [0.60, 0.68] (4)	0.71 [0.67, 0.74] (1)	0.66 [0.62, 0.70] (2)	0.67 [0.63, 0.70] (5)	0.68 [0.64, 0.72] (2)	0.71 [0.67, 0.74] (1)	0.67 [0.63, 0.71] (1)	0.69 [0.65, 0.72] (3)	0.70 [0.66, 0.73] (2)	0.72 [0.68, 0.75] (1)

Table 5: Pearson correlations (r) for the test split across all layers (0-12) utilizing different pooling methods and models, together with the 95% CI. Numbers in parenthesis indicate count of removed test items.

Layer	Last / CLS Token				Average				Average + Sid			
	GPT-2 _{out}	GPT-2 _{in}	RoBERTa _{out}	RoBERTa _{in}	GPT-2 _{out}	GPT-2 _{in}	RoBERTa _{out}	RoBERTa _{in}	GPT-2 _{out}	GPT-2 _{in}	RoBERTa _{out}	RoBERTa _{in}
0	0.53 [0.47, 0.58] (0)	0.53 [0.47, 0.58] (0)	—	—	0.59 [0.55, 0.64] (2)	0.59 [0.54, 0.64] (3)	0.58 [0.53, 0.63] (3)	0.57 [0.52, 0.61] (1)	0.62 [0.58, 0.66] (1)	0.62 [0.58, 0.67] (1)	0.60 [0.55, 0.64] (2)	0.59 [0.54, 0.63] (2)
1	0.56 [0.51, 0.60] (1)	0.55 [0.50, 0.60] (0)	0.60 [0.55, 0.64] (4)	0.49 [0.44, 0.54] (3)	0.60 [0.56, 0.65] (2)	0.58 [0.53, 0.63] (1)	0.60 [0.55, 0.64] (1)	0.59 [0.54, 0.64] (3)	0.58 [0.53, 0.63] (2)	0.58 [0.53, 0.62] (0)	0.60 [0.56, 0.65] (1)	0.60 [0.56, 0.65] (1)
2	0.59 [0.54, 0.63] (1)	0.57 [0.52, 0.62] (0)	0.58 [0.53, 0.63] (3)	0.44 [0.38, 0.49] (7)	0.62 [0.57, 0.66] (1)	0.61 [0.56, 0.65] (1)	0.59 [0.54, 0.63] (1)	0.60 [0.55, 0.64] (1)	0.58 [0.53, 0.63] (1)	0.59 [0.54, 0.64] (1)	0.59 [0.55, 0.64] (1)	0.60 [0.55, 0.64] (1)
3	0.59 [0.55, 0.64] (1)	0.61 [0.56, 0.65] (0)	0.59 [0.54, 0.63] (4)	0.42 [0.36, 0.48] (4)	0.62 [0.57, 0.66] (2)	0.61 [0.57, 0.66] (1)	0.63 [0.58, 0.67] (1)	0.62 [0.57, 0.66] (1)	0.60 [0.56, 0.65] (1)	0.61 [0.57, 0.66] (1)	0.63 [0.58, 0.67] (1)	0.62 [0.58, 0.67] (0)
4	0.59 [0.54, 0.63] (1)	0.61 [0.56, 0.65] (0)	0.61 [0.56, 0.65] (5)	0.40 [0.34, 0.46] (4)	0.64 [0.59, 0.68] (1)	0.59 [0.55, 0.64] (1)	0.62 [0.58, 0.67] (1)	0.61 [0.56, 0.65] (1)	0.61 [0.56, 0.65] (1)	0.57 [0.52, 0.62] (1)	0.62 [0.57, 0.66] (1)	0.60 [0.56, 0.65] (1)
5	0.60 [0.55, 0.64] (3)	0.60 [0.55, 0.65] (0)	0.59 [0.54, 0.63] (1)	0.52 [0.47, 0.57] (7)	0.63 [0.58, 0.67] (2)	0.60 [0.56, 0.65] (2)	0.64 [0.60, 0.68] (1)	0.63 [0.59, 0.67] (0)	0.61 [0.56, 0.65] (1)	0.60 [0.56, 0.65] (1)	0.63 [0.59, 0.67] (1)	0.63 [0.58, 0.67] (0)
6	0.62 [0.57, 0.66] (2)	0.62 [0.58, 0.66] (0)	0.61 [0.56, 0.65] (3)	0.51 [0.45, 0.56] (6)	0.64 [0.59, 0.68] (2)	0.61 [0.57, 0.66] (1)	0.64 [0.60, 0.68] (1)	0.65 [0.61, 0.69] (0)	0.62 [0.57, 0.66] (1)	0.61 [0.56, 0.65] (1)	0.64 [0.60, 0.68] (1)	0.63 [0.58, 0.67] (1)
7	0.64 [0.59, 0.68] (3)	0.62 [0.58, 0.66] (0)	0.62 [0.57, 0.66] (2)	0.53 [0.48, 0.58] (12)	0.64 [0.59, 0.68] (2)	0.61 [0.56, 0.65] (1)	0.65 [0.61, 0.69] (1)	0.65 [0.61, 0.69] (1)	0.63 [0.58, 0.67] (2)	0.61 [0.57, 0.66] (1)	0.64 [0.60, 0.68] (1)	0.65 [0.60, 0.69] (1)
8	0.64 [0.60, 0.68] (3)	0.61 [0.57, 0.65] (0)	0.62 [0.58, 0.67] (1)	0.54 [0.49, 0.59] (7)	0.65 [0.61, 0.69] (2)	0.59 [0.54, 0.63] (3)	0.64 [0.60, 0.68] (1)	0.64 [0.60, 0.68] (1)	0.63 [0.59, 0.68] (4)	0.60 [0.55, 0.64] (1)	0.64 [0.60, 0.68] (1)	0.65 [0.60, 0.69] (1)
9	0.63 [0.58, 0.67] (4)	0.62 [0.57, 0.66] (0)	0.61 [0.56, 0.65] (2)	0.61 [0.57, 0.66] (3)	0.65 [0.61, 0.69] (2)	0.58 [0.53, 0.63] (4)	0.63 [0.59, 0.67] (2)	0.64 [0.60, 0.68] (1)	0.63 [0.59, 0.67] (2)	0.61 [0.56, 0.65] (3)	0.64 [0.60, 0.68] (1)	0.64 [0.60, 0.68] (1)
10	0.63 [0.59, 0.67] (4)	0.62 [0.57, 0.66] (0)	0.61 [0.57, 0.66] (5)	0.63 [0.59, 0.67] (1)	0.64 [0.60, 0.68] (3)	0.57 [0.52, 0.62] (5)	0.63 [0.58, 0.67] (2)	0.63 [0.59, 0.68] (1)	0.64 [0.60, 0.68] (4)	0.59 [0.54, 0.63] (3)	0.64 [0.59, 0.68] (1)	0.63 [0.59, 0.67] (1)
11	0.62 [0.57, 0.66] (5)	0.62 [0.57, 0.66] (0)	0.54 [0.48, 0.58] (2)	0.63 [0.59, 0.67] (1)	0.64 [0.60, 0.68] (2)	0.55 [0.50, 0.60] (4)	0.64 [0.60, 0.68] (3)	0.64 [0.60, 0.68] (1)	0.63 [0.58, 0.67] (2)	0.58 [0.53, 0.62] (3)	0.66 [0.61, 0.70] (2)	0.64 [0.60, 0.68] (1)
12	0.59 [0.54, 0.64] (2)	0.61 [0.56, 0.65] (0)	0.63 [0.59, 0.67] (3)	0.65 [0.61, 0.69] (1)	0.64 [0.60, 0.68] (3)	0.55 [0.50, 0.60] (5)	0.62 [0.58, 0.66] (3)	0.65 [0.61, 0.69] (1)	0.64 [0.59, 0.68] (3)	0.58 [0.53, 0.62] (1)	0.63 [0.59, 0.67] (2)	0.65 [0.61, 0.69] (1)

Table 6: Pearson correlations (r) for the held-out split across all layers (0-12) utilizing different pooling methods and models, together with the 95% CI. Numbers in parenthesis indicate count of removed test items.

Layer	Random Baseline						Token Order Ablation	
	R1		R2		R3		GPT-2	RoBERTa
	GPT-2	RoBERTa	GPT-2	RoBERTa	GPT-2	RoBERTa	GPT-2	RoBERTa
0	0.62 [0.58, 0.66] (3)	0.62 [0.57, 0.66] (2)	0.55 [0.50, 0.60] (3)	0.55 [0.50, 0.60] (3)	0.02 [-0.05, 0.10] (8)	-0.01 [-0.08, 0.07] (10)	0.63 [0.59, 0.68] (2)	0.65 [0.61, 0.69] (2)
1	0.57 [0.52, 0.62] (2)	0.62 [0.57, 0.66] (2)	0.49 [0.43, 0.54] (4)	0.54 [0.49, 0.59] (2)	0.01 [-0.06, 0.08] (3)	0.02 [-0.05, 0.09] (3)	0.66 [0.62, 0.70] (1)	0.66 [0.62, 0.70] (1)
2	0.56 [0.51, 0.61] (2)	0.61 [0.56, 0.65] (2)	0.49 [0.44, 0.54] (4)	0.55 [0.50, 0.60] (2)	0.12 [0.05, 0.19] (2)	0.04 [-0.03, 0.11] (6)	0.67 [0.63, 0.71] (1)	0.66 [0.62, 0.70] (1)
3	0.56 [0.51, 0.61] (1)	0.61 [0.56, 0.65] (1)	0.50 [0.44, 0.55] (2)	0.54 [0.49, 0.59] (2)	0.13 [0.06, 0.20] (3)	0.08 [0.01, 0.15] (9)	0.67 [0.63, 0.71] (1)	0.65 [0.60, 0.69] (2)
4	0.56 [0.51, 0.61] (1)	0.60 [0.56, 0.65] (1)	0.51 [0.45, 0.56] (3)	0.53 [0.48, 0.58] (5)	0.09 [0.01, 0.16] (2)	0.11 [0.04, 0.18] (10)	0.66 [0.62, 0.70] (1)	0.65 [0.60, 0.69] (1)
5	0.56 [0.51, 0.60] (3)	0.60 [0.56, 0.65] (1)	0.50 [0.45, 0.55] (4)	0.54 [0.49, 0.59] (5)	0.10 [0.03, 0.17] (5)	0.16 [0.09, 0.23] (10)	0.65 [0.61, 0.69] (1)	0.63 [0.59, 0.67] (2)
6	0.55 [0.50, 0.60] (1)	0.60 [0.55, 0.64] (1)	0.49 [0.44, 0.55] (2)	0.55 [0.49, 0.59] (5)	0.10 [0.03, 0.17] (3)	0.19 [0.12, 0.26] (8)	0.65 [0.61, 0.69] (3)	0.64 [0.59, 0.68] (2)
7	0.55 [0.50, 0.60] (1)	0.59 [0.54, 0.64] (2)	0.50 [0.44, 0.55] (3)	0.54 [0.49, 0.59] (6)	0.10 [0.03, 0.17] (5)	0.21 [0.14, 0.28] (10)	0.65 [0.61, 0.69] (2)	0.62 [0.57, 0.66] (4)
8	0.55 [0.50, 0.60] (2)	0.59 [0.54, 0.64] (1)	0.50 [0.44, 0.55] (3)	0.54 [0.49, 0.59] (7)	0.09 [0.02, 0.16] (5)	0.25 [0.18, 0.31] (9)	0.65 [0.61, 0.69] (2)	0.62 [0.57, 0.66] (4)
9	0.56 [0.50, 0.60] (2)	0.59 [0.54, 0.64] (1)	0.49 [0.44, 0.54] (3)	0.54 [0.49, 0.59] (5)	0.06 [-0.01, 0.13] (3)	0.28 [0.21, 0.34] (9)	0.65 [0.61, 0.69] (1)	0.61 [0.56, 0.65] (3)
10	0.55 [0.50, 0.60] (2)	0.59 [0.54, 0.63] (2)	0.49 [0.43, 0.54] (3)	0.55 [0.50, 0.60] (4)	0.07 [0.00, 0.14] (5)	0.30 [0.23, 0.36] (9)	0.65 [0.60, 0.69] (2)	0.61 [0.57, 0.66] (3)
11	0.56 [0.50, 0.60] (1)	0.59 [0.54, 0.63] (2)	0.49 [0.43, 0.54] (3)	0.55 [0.49, 0.59] (6)	0.09 [0.01, 0.16] (5)	0.32 [0.26, 0.38] (10)	0.63 [0.59, 0.68] (3)	0.62 [0.57, 0.66] (4)
12	0.55 [0.50, 0.60] (1)	0.59 [0.54, 0.63] (2)	0.48 [0.42, 0.53] (3)	0.55 [0.50, 0.59] (6)	0.12 [0.05, 0.19] (4)	0.32 [0.26, 0.39] (10)	0.63 [0.59, 0.67] (6)	0.61 [0.56, 0.65] (3)

Table 7: Pearson correlations (r) for the random baselines and the token order ablation on the test split across all layers (0-12), together with the 95% CI. Numbers in parenthesis indicate count of removed test items.

Layer	Random Baseline						Token Order Ablation	
	R1		R2		R3		GPT-2	RoBERTa
	GPT-2	RoBERTa	GPT-2	RoBERTa	GPT-2	RoBERTa	GPT-2	RoBERTa
0	0.53 [0.47, 0.58] (0)	0.54 [0.49, 0.59] (0)	0.44 [0.38, 0.50] (4)	0.46 [0.40, 0.52] (6)	0.01 [-0.06, 0.08] (5)	0.01 [-0.06, 0.08] (6)	0.59 [0.54, 0.63] (1)	0.58 [0.53, 0.62] (2)
1	0.52 [0.47, 0.57] (0)	0.54 [0.48, 0.58] (0)	0.39 [0.33, 0.45] (3)	0.45 [0.39, 0.50] (5)	-0.03 [-0.10, 0.04] (2)	-0.02 [-0.09, 0.05] (5)	0.57 [0.52, 0.62] (3)	0.57 [0.52, 0.62] (4)
2	0.51 [0.45, 0.56] (2)	0.53 [0.47, 0.58] (0)	0.38 [0.31, 0.44] (3)	0.43 [0.38, 0.49] (5)	0.04 [-0.03, 0.11] (2)	-0.02 [-0.09, 0.06] (3)	0.58 [0.53, 0.62] (3)	0.54 [0.49, 0.59] (2)
3	0.50 [0.44, 0.55] (2)	0.53 [0.47, 0.58] (0)	0.36 [0.29, 0.42] (2)	0.45 [0.39, 0.51] (4)	0.10 [0.02, 0.17] (4)	0.04 [-0.03, 0.11] (3)	0.57 [0.52, 0.62] (2)	0.56 [0.51, 0.60] (1)
4	0.50 [0.44, 0.55] (3)	0.52 [0.47, 0.57] (0)	0.37 [0.30, 0.43] (3)	0.47 [0.41, 0.52] (3)	0.06 [-0.01, 0.13] (4)	0.06 [-0.01, 0.13] (6)	0.56 [0.51, 0.61] (2)	0.56 [0.51, 0.61] (1)
5	0.49 [0.44, 0.55] (2)	0.52 [0.46, 0.57] (0)	0.37 [0.31, 0.43] (3)	0.48 [0.42, 0.53] (4)	0.04 [-0.03, 0.11] (5)	0.10 [0.02, 0.17] (7)	0.55 [0.50, 0.60] (4)	0.56 [0.51, 0.61] (2)
6	0.50 [0.44, 0.55] (2)	0.53 [0.47, 0.58] (0)	0.38 [0.32, 0.44] (2)	0.47 [0.42, 0.53] (2)	0.09 [0.02, 0.16] (3)	0.11 [0.04, 0.18] (9)	0.56 [0.51, 0.61] (4)	0.54 [0.49, 0.59] (3)
7	0.50 [0.44, 0.55] (4)	0.53 [0.47, 0.58] (0)	0.39 [0.33, 0.45] (2)	0.49 [0.44, 0.55] (3)	0.09 [0.01, 0.16] (3)	0.15 [0.08, 0.22] (7)	0.57 [0.52, 0.62] (4)	0.54 [0.49, 0.59] (3)
8	0.49 [0.43, 0.54] (2)	0.52 [0.47, 0.57] (0)	0.40 [0.34, 0.46] (2)	0.50 [0.45, 0.56] (4)	0.08 [0.01, 0.15] (0)	0.17 [0.10, 0.23] (6)	0.58 [0.53, 0.62] (4)	0.55 [0.50, 0.60] (4)
9	0.50 [0.45, 0.55] (0)	0.52 [0.47, 0.57] (0)	0.40 [0.34, 0.46] (3)	0.51 [0.45, 0.56] (3)	0.09 [0.02, 0.16] (1)	0.19 [0.12, 0.26] (6)	0.57 [0.52, 0.62] (5)	0.55 [0.50, 0.60] (4)
10	0.51 [0.45, 0.56] (0)	0.53 [0.48, 0.58] (0)	0.41 [0.34, 0.46] (3)	0.51 [0.46, 0.56] (3)	0.10 [0.02, 0.17] (3)	0.21 [0.14, 0.28] (6)	0.56 [0.51, 0.60] (5)	0.53 [0.48, 0.58] (4)
11	0.51 [0.45, 0.56] (1)	0.53 [0.48, 0.58] (0)	0.38 [0.32, 0.44] (2)	0.53 [0.47, 0.58] (5)	0.08 [0.01, 0.15] (3)	0.23 [0.16, 0.29] (6)	0.54 [0.49, 0.59] (5)	0.53 [0.47, 0.58] (4)
12	0.48 [0.42, 0.53] (2)	0.54 [0.49, 0.59] (1)	0.38 [0.32, 0.44] (2)	0.51 [0.46, 0.56] (3)	0.10 [0.03, 0.17] (3)	0.25 [0.18, 0.32] (5)	0.52 [0.47, 0.57] (2)	0.52 [0.47, 0.57] (3)

Table 8: Pearson correlations (r) for the random baselines and token order ablation on the held-out split across all layers (0-12), together with the 95% CI. Numbers in parenthesis indicate count of removed test items.