Probing BERT for Ranking Abilities

Wallat, Jonas; Beringer, Fabian; Anand, Abhijit; Anand, Avishek

DOI
10.1007/978-3-031-28238-6_17

Publication date
2023

Document Version
Final published version

Published in
Advances in Information Retrieval - 45th European Conference on Information Retrieval, ECIR 2023, Proceedings

Probing BERT for Ranking Abilities

Jonas Wallat[1(✉)], Fabian Beringer[1], Abhijit Anand[1], and Avishek Anand[1,2]

[1] L3S Research Center, Hannover, Germany
{jonas.wallat,fabian.beringer,abhijit.anand}@l3s.de
[2] TU Delft, Delft, Netherlands
avishek.anand@tudelft.nl

Abstract. Contextual models like BERT are highly effective in numerous text-ranking tasks. However, it is still unclear as to whether contextual models understand well-established notions of relevance that are central to IR. In this paper, we use *probing*, a recent approach used to analyze language models, to investigate the ranking abilities of BERT-based rankers. Most of the probing literature has focussed on linguistic and knowledge-aware capabilities of models or axiomatic analysis of ranking models. In this paper, we fill an important gap in the information retrieval literature by conducting a layer-wise probing analysis using four probes based on lexical matching, semantic similarity as well as linguistic properties like coreference resolution and named entity recognition. Our experiments show an interesting trend that BERT-rankers better encode ranking abilities at intermediate layers. Based on our observations, we train a ranking model by augmenting the ranking data with the probe data to show initial yet consistent performance improvements (The code is available at https://github.com/yolomeus/probing-search/).

1 Introduction

Large contextual models such as BERT [14] have delivered impressive and robust performance gains in many NLP and IR tasks. However, these over-parameterized contextual models are still used as functional black boxes with little understanding of what the contextual embedding spaces actually encode. Towards this, *probing* was introduced as a procedure to investigate whether specific linguistic properties or factual information are present in contextual text representations [6], which enable large contextual models to perform well on language tasks. Probes offer insight into otherwise functionally opaque contextual models. Most of the effort in designing probes is to ground the behavior of large contextual models in well-understood linguistic properties and world knowledge. For example, a *part-of-speech* (POS) probe investigates to what degree contextual representations encode POS information in their representations. This innate ability to encode POS is typically investigated by learning a lightweight classifier, called a *probe*, to predict the POS property from the embeddings. The performance of a probe measures the quality of the *contextual representations*.

J. Kamps et al. (Eds.): ECIR 2023, LNCS 13981, pp. 255–273, 2023.
https://doi.org/10.1007/978-3-031-28238-6_17

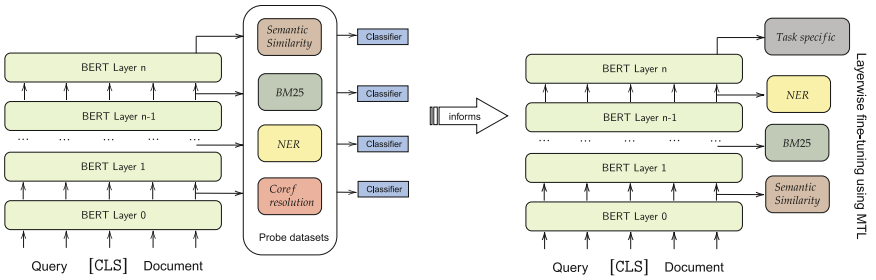

Fig. 1. Procedural overview: in the first set of experiments, we probe for different abilities of neural ranking models (e.g., BM25, semantic similarity). We then utilize the information where the model best captures these properties to give additional training signals to that specific layer during multi-task learning.

Consequently, various task-specific probing tasks have been developed to investigate contextual embeddings for linguistic and factual knowledge [6,36,48,55].

This paper focuses on large contextual models that have been applied with major success in information retrieval tasks. However, there is limited work on probing for IR and, particularly, to *text ranking tasks*. Until now, most studies focused on probing for linguistic [23,48] or factual knowledge [35,36] of pre-trained models, e.g., finding that BERT's layers and their abilities coincide with the classical NLP pipeline [47] or that dependency parse trees can be decoded from BERT's embeddings [23]. There has also been work on investigating the evolution of higher-level factual and linguistic knowledge through the layers of large contextual models [47,52]. Most of the existing work in explaining the behavior of contextual ranking models is through IR axioms [7,41,51]. Although axioms are well-established, formal descriptions of *what makes a good text ranker*, they have limited modeling of semantic similarity and have been shown to have limited applicability to explain neural rankers [7,51].

1.1 Research Questions

We aim to fill the gap of characterizing the performance of neural rankers in terms of IR abilities by proposing probing methods. Through probing, we try to understand the behavior of ranking models by grounding it on well-understood IR properties and best practices for text ranking – *matching, semantic similarity,* in conjugation with essential linguistic properties of *named entity recognition,* and *coreference resolution*. We answer the following research questions:

RQ 1. What abilities do neural rankers acquire to perform the ranking task?

RQ 2. Can we apply the knowledge to build better ranking models?

1.2 Summary of Contributions

First, we construct probing datasets for the probing tasks of lexical matching, semantical similarity, named entity recognition (NER), and coreference

resolution from the MS MARCO dataset [49]. Next, we measure – (a) the degree to which a ranking model understands the IR property (*accuracy*) and, (b) the degree to which the property is extractable from the ranking model (*minimum description length*). Figure 1 depicts and overview of our experiments. We conduct extensive experiments using multiple probes over multiple layers of a BERT-fine-tuned ranking model. Other than existing works that predominantly report only probing results, we operationalize our findings by constructing multi-task learning-based ranking models using auxiliary tasks based on probes.

Results. Our probing study shows that ranking models prioritize lexical and semantical similarity and coreference information over NER abilities. Moreover, we usually find intermediary layers (4–6) to best capture these concepts. We also find that training ranking models in a multi-task learning setup (i.e., ranking and the aforementioned ranking subtasks) can be beneficial - especially when we use our probing results to inform on which layer to train the subtasks.

2 Related Work

Probing large and overparameterized contextual models was introduced by Conneau et al. [11] in the NLP community to improve their interpretability. This work aims to probe neural rankers to understand their IR abilities as a step towards explainable IR [4]. Specifically, probing is a posthoc interpretability approach that, instead of optimizing fidelity [40,45,46], tries to ground the knowledge or abilities stored in the parametric memory of neural rankers.

2.1 Probing for Linguistic Properties

Tenney et al. [48] proposed aggregating individual word embeddings to move from word-level probing to subsentences, allowing to probe for coreference and other semantic, long-range concepts. Consequently, many works used this methodology. Zhao et al. [56] investigated how contextualized BERT embeddings are. Tenney et al. [47] probed BERT and found early layers to focus on lower-level concepts, such as *syntax*, and more-involved higher layers on concepts such as *semantics*. Subsequent work improving the probing paradigm either by contextualizing the probing results with suitable baselines [21,54], introducing control tasks [22], or characterizing embedding vs classifier performance [37,50]. For detailed overview of the probing literature until 2019, we refer to the review by Belinkov and Glass [6]. We include many of the best practices in the literature in our work. Many works have investigated task-specific probing [2,52]. Most related to our work is Wallat et al. [52], who also perform a layer-wise probing to check the retention of factual knowledge in BERT. Their layer-wise analysis suggests that most factual knowledge resides in the later layers of the models, with the ranking model outperforming other fine-tuned models in knowledge retention. We instead probe for ranking abilities.

2.2 Probing in IR

In the context of IR, MacAvaney et al. [31] study the ranking models using a large set of diagnostic probes such as *term-frequency*. They also study the effects of shuffling word orders or paraphrasing on the ranking performance. Fan et al. [16] show that the ranking model improved in capturing *synonym detection* information while sacrificing the ability to identify named entities. While both of these works investigated the abilities of IR models, they focus on only the final representation of that is derived from the last layer of the model. We believe that investigating the flow of information through the intermediate layers can yield additional insights. Furthermore, both [16,31] do not contextualize the probing results with standard probing baselines like control tasks, or a measure of *ease of extraction* (e.g., MDL) as recommended in the probing literature [5].

2.3 Axiomatic Interpretability

Similar to probing, neural rankers have also been diagnosed or interpreted using IR axioms [7,41,51]. These works either directly rank documents according to specific axioms such as "if document A contains more query terms than document B, then A should be ranked higher" [20], check whether rankers conform with axioms using diagnostic datasets [41], or try to explain neural rankers with these axioms [7,51]. However, most of these approaches have reported limited success. Völske et al. [51] find that axiomatic explanations frequently fail if models are not confident in their decision and that the existing axioms are insufficient in explaining the complex decisions of ranking models [7]. By investigating the acquired abilities of ranking models, we position our work between the existing high-level investigation into factual knowledge containment [52] and explaining model decisions by shallow features (i.e., axioms) [51].

2.4 Understanding Relevance Factors Without Probing

Apart from probing, the attention patterns of ranking models have been under investigation, finding that redundant attention often focuses on tokens with a high document frequency (e.g., punctuation) [53] and that the attention captures inverse-document frequency information [9]. Furthermore, Qiao et al. [38] investigate the attention and term-matching behavior of BERT and find that it focuses more on query tokens that appear in the document, suggesting attention and lexical matching being deciding factors for BERT's performance gains. Rau and Kamps [39] study the role of NLP abilities in the effectiveness of neural ranking models. By constructing inputs without word order information, they find that while word order seems highly relevant for BERT's pretraining, it is not necessary for relevance estimation.

2.5 Data Augmentation in IR

In the second part of our paper, we use additional training signals from our probe tasks to train ranking models. While there is existing work that utilized information such as BM25 to train rankers either with weak supervision [13] or by

data augmentation [3,44], our work is, to the best of our knowledge, the first to operationalize probing results to build more effective ranking models. We specifically probe the representations of the common early interaction BERT ranker as proposed by Nogueira et al. [33], which applies a linear layer to the [CLS] token in order to estimate relevance. Besides early interaction methods, there have been recent works on late interaction models [18,26,29], where independent document and query representations only interact in the last layer.

3 Probing BERT Ranking Models

The ability of a text ranker to effectively rank documents given an underspecified query is based on many well-understood principles in IR like term matching, document frequency, and length normalization, among others [32]. In this work, we are interested in BERT rankers, but our analysis can naturally be extended to other overparameterized contextual rankers with multiple transformer layers.

3.1 Problem Statement

Given a trained (or fine-tuned) text ranking model $\mathcal{M}$, we are interested in measuring the degree to which output representations of $\mathcal{M}$ satisfy or adhere to well-understood ranker properties. For each ranking property i, a *probing dataset* P_i is constructed. To measure if a property i is well-captured in $\mathcal{M}$, our objective is to train a probing classifier or simple a *probe* g_i given the output representation/s or embedding from $\mathcal{M}$ to generalize on the probing dataset P_i.

3.2 Layerwise Probing

We conduct probing analysis on multiple layers of $\mathcal{M}$ to assess the evolution of ranking properties across layers of the ranking model. For each of our ranking subtasks (Sect. 3.4) and each layer of the model, we train a simple MLP classifier over the model's output representations or embeddings.

We follow the probing paradigm that is based on the general assumption that an *above-chance performance* on the probing tasks indicates the presence of task knowledge in the embeddings. These probing performances need to be put into context by how hard the task is (e.g., by comparing performance with suitable baselines [54]) and how much of the performance can actually be attributed to the classifier [37,50]. Towards addressing these concerns, we first carefully select random and pre-trained baselines to compare against (refer Sect. 4.2) and secondly use the *minimum description length* (MDL) to measure attributability. Next, we detail our probing setup with MDL.

3.3 Probing with Minimum Description Length

By applying the information-theoretic concept of minimum description length to the probing paradigm, Voita and Titov [50] address the question: *how well the*

model encodes certain information? If the embedding encodes a concept such as named entities more efficiently, it can describe this information more precisely. In that case, the minimum description length will be shorter than in embeddings that do not capture named entity information.

To compute MDL, we use the online code definition [42]. For this, the dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$ is divided into timesteps $1 = t_0 < t_1 < \ldots < t_S = N$. After encoding block t_0 with a uniform code, for each following timestep, a probing model p_{θ_i} is trained on the samples $(1, \ldots, t_i)$ and used to predict over data points $(t_i + 1, \ldots, t_{i+1})$. The full MDL is then computed as a sum over the codelengths of each p_{θ_i} and the uniform encoding of the first block:

$$L(y_{1:n}|x_{1:n}) = t_1 \log_2 C - \sum_{i=1}^{S-1} \log_2 p_{\theta_i}(y_{t_i+1:t_{i+1}}|x_{t_i+1:t_{i+1}}) \tag{1}$$

where C is the number of target classes. Following Voita and Titov [50], we choose timesteps at 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.25, 12.5, 25, 50 and 100% of the dataset.

Similarly to Fayyaz et al. [17], we reformulate MDL to *compression*. For this, MDL is scaled in relation to the codelength of a uniform encoding:

$$\texttt{compression} = \frac{N \log_2(C)}{\text{MDL}} \tag{2}$$

where N is the number of targets, and C is the number of target classes. Since MDL depends on the total number of targets, a relative measure, like compression, is more practical for comparing tasks. Furthermore, both accuracy and compression are to be maximized, while MDL is to be minimized.

MDL is only defined for classification tasks as it requires the number of target classes. Therefore, we reformulate regression tasks to classification tasks by binning target scores into $k = 10$ equally sized class bins.

3.4 Probing Tasks

For a selection of principled ranking abilities, we utilize well-known abilities of ranking models from the information retrieval (IR) literature: Arguably, one of the most fundamental ranking subtasks is a model's ability to match text, which has been widely used either for classical ranking models [43] or to inform the pre-finetuning of neural rankers [27]. Furthermore, we probe for the ranking model's ability to match according to the semantic meaning [30]. Given that a large part of queries focus on entities and that named entity recognition (NER) can have a positive impact on IR [25], we include NER as one of our tasks. Lastly, we include coreference resolution, which is not canonically associated with principled ranking. With the established importance of entity recognition, we wonder how well ranking models can perform the matching of entity surface forms between queries and documents.

For our experiments, we compile a list of abilities that neural ranking models might employ for predicting document relevance. We choose our tasks as follows:

BM25 Prediction. The BM25 algorithm [43] uses lexical matching to estimate relevance and is widely used in ranking. We ask whether neural rankers encode the necessary information to perform well at measuring lexical similarity. The BM25 formula includes inverted document frequencies of the terms; therefore, to accurately predict BM25, the ranker needs to implicitly learn term distributions in the dataset. We use query document pairs from the MS MARCO test set and predicted BM25 scores as labels to create the probing dataset.

Semantic Similarity. Like lexical matching, it seems very probable that part of the ranking model's performance can be attributed to semantic matching. We test whether semantic similarity information resides in the embeddings of our rankers. Similar to existing work in axiomatic IR [51], we estimate the semantic similarity between query and document pairs by the cosine similarity between the average GloVe [34] query and document embeddings (after stop-word removal).

Named Entity Recognition. Since user queries usually ask for some information about entities, we test the models' ability to identify entities. To do so, we use the Spacy [24] named entity recognizer and tag all named entities in MS MARCO query-document pairs.

Coreference Resolution. Queries are often underspecified [10]. We, therefore, include the probing task of coreference resolution between entity mentions in the query and surface form occurrences in the document into our suite of tasks. Given a query "trump birthplace", the task is to match an entity from the query ("trump") to surface forms in a document (e.g., "Donald Trump", "the former president"). To find coreference pairs, we use Huggingface's neuralcoref[1].

4 Experimental Setup

4.1 Datasets

MS MARCO: We use the TREC Deep Learning track (2019) dataset (TREC-DL) for evaluation. Our models are evaluated on the TREC-DL test split which contains 200 queries. For creating training and development splits we use MS MARCO, containing 532k queries. To retrieve documents from the corpus of $\sim$ 8.8mio passages, we use BM25.

Probing: Since our (contextual) ranking models are trained on MS MARCO, we explicitly use the MS MARCO test set to create our probing datasets. For this, we uniformly sample 60k query-passage pairs, where 40k are used for training, and 10k for validation and testing, respectively.

4.2 Models

We conduct our probing experiments on BERT [14], using three different base models throughout our experiments:

[1] https://github.com/huggingface/neuralcoref.

1. **BERT-BASE-UNCASED** - the publicly available[2] pre-trained BERT model consisting of 12-layer, 768 dimensions, 12-heads, 110M parameters. The length of the input is restricted to 512 tokens.
2. **BERT-MSM-PASSAGE** - bert-base model, fine-tuned on MS MARCO for the TREC-DL 2019 *passage* level ranking task [12]
3. **BERT-MSM-DOC** - the bert-base model, fine-tuned on MS MARCO for the TREC-DL 2019 *document* level ranking task.

The ranking models were trained with a similar setup as Nogueira et al. [33] for up to 20 epochs on using the binary cross-entropy objective.

4.3 Training Probe Models

For all tasks, we train a 2-layer MLP probe model with self-attention pooling (similar to [48]) for up to a maximum of 50 epochs and perform early stopping after 10 epochs if no improvement in validation loss has been measured. As an optimization algorithm, we use Adam [28] with a batch size of 32 and clip gradients with an L2-norm greater than 5. We start with a learning rate of 1e–4 and half it at the end of an epoch if the validation loss does not improve.

5 Results

To establish which ranking ability is learned by fine-tuning on ranking datasets (**RQ 1**), we compare the performance of a fine-tuned passage ranking (BERT-MSM-PASSAGE) and a document ranking model (BERT-MSM-DOC) to two baselines: 1) a pretrained model without fine-tuning, and 2) model with random weight initialization. For a pre-trained model, we use a BERT model (BERT-BASE-UNCASED). Furthermore, we use BERT input embeddings with random weight initialization as a source of random embeddings [54].

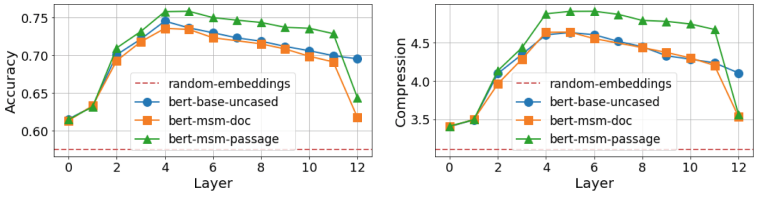

Fig. 2. Probing results over the layers for the BM25 task.

[2] https://huggingface.co/bert-base-uncased.

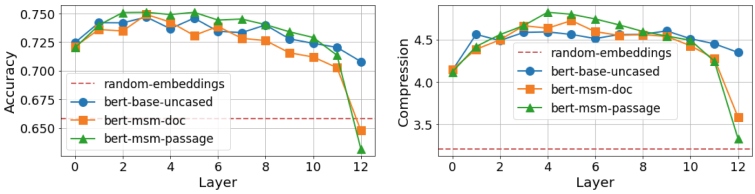

Fig. 3. Probing results over the layers for the semantic similarity task.

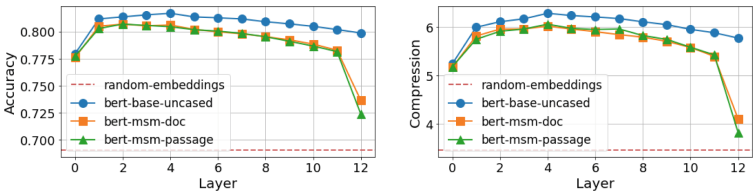

Fig. 4. Probing results over the layers for the NER task.

5.1 Matching Ability of Ranking Models

Figure 2 presents the degree to which fine-tuned BERT models have learned to predict BM25 or, in other words, exhibit the ability to perform term matching. The plot on the left shows *task accuracy* and the plot on the right shows *compression* over the layers (metric introduced in Sect. 3.2). First of all, and expectedly, we can see that all three models capture more BM25 information than random embeddings to a large degree. While the accuracy seems to differ only slightly, we can observe that the compression of BERT-MSM-PASSAGE is markedly higher than for the other two models. A higher compression score means the BM25 information is more easily decodable from the ranking models' embeddings. By probing all layers of our models, we can also understand in which layer the matching ability is best captured. It is evident that the BM25 knowledge increases until layer 5 or 6 and then slowly decreases until layer 11. In layer 12, the performance decreases starkly - a result that is in line with multiple works finding that the last layer is the most task-specific and therefore performs worse in other tasks ([2,52] inter alia). Additionally, recent work by Ghasemi et al. [19] suggests that BERT rankers do not fully rely on lexical matching, which is also indicated by BM25 knowledge decreasing in the later layers.

5.2 Ability to Capture Semantic Similarity

The probing results for semantic similarity are shown in Fig. 3. Again, we can observe similar trends. Semantic similarity appears to be best captured in layer 4 (compared to layers 5 or 6 for BM25). Like with BM25, we can see the ranking models' compressions to be slightly improved over the pre-trained model – suggesting that training the models on ranking emphasizes understanding and capturing semantic similarity.

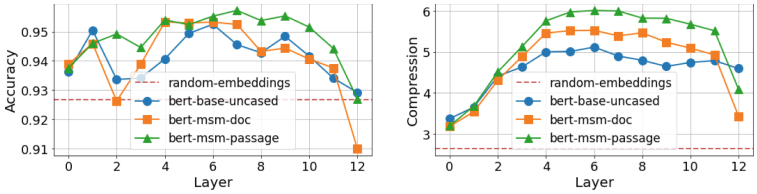

Fig. 5. Probing results over the layers for the coreference resolution task.

5.3 Other Abilities

Figures 4 and 5 show the probing performance on NER and coreference resolution, respectively. Interestingly, we find that the ranking models do not spot entities better than a pre-trained model (Fig. 4). Although the results suggest that the identification of entities is not a priority, matching surface forms of entities is better encoded after fine-tuning on the ranking task.

5.4 Insights and Summary

Our first insight is that, compared to BERT-MSM-DOC, BERT-MSM-PASSAGE shows a better accuracy-compression trade-off in all the auxiliary tasks considered. In other words, not only does BERT-MSM-PASSAGE exhibit primitive ranking abilities, but these abilities are easily extractable for text ranking tasks. Second, all considered auxiliary tasks are best encoded at intermediary layers and slowly decrease towards the final layer. This shows that deep contextual models used as rankers extract features that are in some sense compositional in nature, with lower-level abilities being exhibited in the lower layers. We believe that the abilities we deal with are intermediate abilities. Existing layerwise studies have shown that ranking models exhibit higher-level abilities in the last few layers [52]. Finally, we observe that BM25, semantic similarity, and coreference resolution are better encoded in ranking models. NER, on the other hand, seems to be deprioritized by the re-ranking models in our study, confirming earlier results [16].

6 Can the Probing Results Be Used for Building Better Rankers?

Until now, we have established that fine-tuned ranking models exhibit basic linguistic and information retrieval abilities. To answer **RQ 2**, we operationalize our findings. Towards this, along with the ranking training set, we construct three task datasets (BM25, NER, semantic similarity). As in this setting, we aim for learning ranking on MS MARCO, we only use queries from the train set to prevent test overlap. For each task, we sample $100k$ queries and, using BM25, retrieve 10 documents each. This results in 1 million samples per task which is approximately the size of our pointwise MS MARCO training set.

We employ a multi-task learning (MTL) setup where we train the ranking task together with individual ranking subtasks (Sect. 3.4). To support the model's learning process, we directly funnel the subtask signal into the model at the corresponding layer where it was best captured (as identified during the probing experiments) and supply the ranking signal in the last layer.

6.1 MTL Training

Multi-task learning is an approach of training multiple tasks in parallel with shared representations to share knowledge across tasks [8]. This has been shown to improve generalization. To train on multiple tasks simultaneously, we uniformly draw samples from the pool of both datasets until the batch size is reached. We then pass the resulting mixed batch through the language model and retrieve the intermediate output representations at each layer. For simplicity, average pooling over the sequence dimension is performed at the desired layers, and a task-specific 2-layer MLP is applied, which takes the following form:

$$\mathrm{MLP}(x) = W_1\sigma(W_0x + b_0) + b_1 \tag{3}$$

with $W_0 \in \mathbb{R}^{m\times n}$, $W_1 \in \mathbb{R}^{n\times k}$ and $b_0 \in \mathbb{R}^n$, $b_1 \in \mathbb{R}^k$ as learnable parameters and σ as the RELU activation. Analogously to our probing experiments, we cast regression to classification tasks by binning the targets into $k = 10$ categories. For our loss function, we use the simple scaling scheme proposed in [1]

$$\mathcal{L}(y_i, \hat{y}_i) = \frac{\mathrm{CE}(y_i, \hat{y}_i)}{\log k_i} \tag{4}$$

where y_i and $\hat{y}_i$ are target and prediction for datapoint i respectively, CE is the cross-entropy loss and k_i denotes the number of target classes for point i, e.g. for a binary target $k_i = 2$. For experiments with the pairwise objective, we similarly use margin loss with $\lambda = 0.2$.

Table 1. Effect of different loss objectives on ranking with BM25 as auxiliary task on the TREC-DL 2019 dataset. pt and pr refer to the pointwise and pairwise training objectives. * marks a significant improvement (p-value < 0.1).

Model	Layer	MAP	MRR	nDCG@10	nDCG@20	P@10	P@20
Ranking (pt-baseline)	12	0.436	0.926	0.678	**0.653**	0.784	**0.685**
Ranking + BM25 (pt)	5	**0.437**	**0.947**	**0.682**	0.652	**0.791**	0.680
Ranking (pr-baseline)	12	0.433	**0.965**	0.681	0.652	0.772	0.670
Ranking + BM25 (pr)	5	**0.452***	**0.965**	**0.685**	**0.673***	**0.786**	**0.708***

Table 2. Effect of layers on MTL performance on the TREC-DL 2019 and 2020 dataset. While we train the ranking task (pointwise loss) always on the final layer, we experiment with different layers for the auxiliary tasks (BM25, named entity recognition, semantic similarity). Bold values indicate the best performance out of all configurations of that specific model (e.g., for all Ranking+BM25 models). */** mark a significant improvement (p-value $< 0.1/0.05$ respectively).

Model	Layer	TREC 19			TREC 20		
		MAP	MRR	nDCG@10	MAP	MRR	nDCG@10
Ranking (baseline)	12	0.436	0.926	0.678	0.446	0.875	0.674
Ranking + BM25	5	0.437	0.947	0.682	0.454	0.900	0.680
	6	**0.439**	**0.953***	**0.690**	**0.460****	**0.932***	**0.689**
	12	0.420	0.912	0.659	0.450	0.927	0.668
Ranking + NER	4	**0.447***	**0.950**	0.685	**0.466****	**0.922****	**0.705****
	5	0.444	0.934	0.680	0.451	0.859	0.679
	12	**0.447**	0.944	**0.688**	0.464**	0.912	0.691
Ranking + Sem	1	0.436	**0.934***	**0.682**	0.451	0.910	**0.687**
	4	**0.440**	0.928	**0.682**	0.453	0.897	0.677
	12	0.436	0.928	0.669	**0.458***	**0.913**	0.683

6.2 MTL Results

First, we train both ranking-only and MTL (Ranking + BM25) models in pairwise and pointwise fashion. Table 1 presents these results.

The experiment suggests that the multi-task training setup with training BM25 on layer 5, as well as ranking on layer 12, improves the overall task performance. While there is an improvement in the pointwise training, we observe larger improvements in the pairwise setting.

Insight. Combining the ranking task with auxiliary tasks can improve the overall ranking performance.

6.3 Effect of MTL Layers on Performance

Next, we investigate if selecting the layer with the best probing performance does hold a benefit over choosing the last layer in our MTL setup.

Table 2 presents the results of multi-task training setups with the ranking task on layer 12 and auxiliary tasks on varying layers. Given significantly higher training times in the pairwise setting, we trained these models with a pointwise objective. It is evident that for the BM25 task, there is a benefit to selecting the MTL layer according to the probing results. Using the 12th layer for training both ranking and BM25 leads to a degradation in ranking performance (compared to the baseline model). Adding semantic-similarity based data augmentation, however, yields no clear trend on the TREC-DL 2019 and 2020 datasets. We hypothesize that BERT embeddings and the self-attention mechanism are

sufficient in estimating query document similarity for the re-ranking task. Also, the construction of gold labels by using GloVe embeddings might not capture semantic similarity as it is used by BERT. For NER, we see all chosen layers to be beneficial. This might be the result of ranking models dropping NER to some capacity (see Fig. 4) and directly forcing the model to include NER information being helpful for the ranking task and specifically the entity-driven MS MARCO dataset. The probing study results suggest that NER information is not prioritized while acquiring the ability to rank passages.

Insight. Choosing the MTL layer according to the probing results can outperform choosing the last layer.

6.4 Threats to Validity

The general shortcoming of probing studies is that a high probing accuracy is not a causal reason for applicability during inference [5,15,50]. Secondly, the decrease in probing task performance over the later layers suggests that the model prioritizes other, potentially compositional, information over our considered IR abilities. At this point, we do not fully understand what information is *used* for relevance estimation. The MTL experiments are a first step towards applying the information gathered from probing studies and are able to show some statistically significant improvements using a very simple MTL setup. The question of how much performance improvement is possible by augmenting additional training signal at intermediary layers will require additional research on the optimal *location*, *tasks*, as well as the right *amount* of training signal to be supplied.

7 Discussion and Conclusion

In this paper, we study the abilities acquired by neural ranking models. To do so, we construct probing datasets from MS MARCO and study how well ranking models encode lexical and semantic similarity, named entity recognition, and coreference resolution. We find ranking models to better encode lexical and semantic similarity as well as coreference resolution. Unlike previous work, which only investigated the final layer, we find these abilities to be best captured at an intermediary layer and to drop toward the final layer, posing the question of what information ranking models utilize for relevance estimation. We later use this information on which layers best encode the tasks to inform our multi-task learning setup. Our experiments show that training the ranking task on the final, and the auxiliary task (e.g., lexical similarity) on the layer with the best probing performance can outperform training both tasks on the final layer. More work, exceeding our naive MTL setup, has to be done to see how much improvement really is possible. Nevertheless, we see potential in adding ranking subtasks to the training setup for improving generalization and data efficiency. To the best of our knowledge, this is the first work to show that the probing results are not purely informational and can be used to improve the model-building process.

Acknowledgements. This research was (partially) funded by the Federal Ministry of Education and Research (BMBF), Germany under the project LeibnizKILabor with grant No. 01DD20003.

References

1. Aghajanyan, A., Gupta, A., Shrivastava, A., Chen, X., Zettlemoyer, L., Gupta, S.: Muppet: Massive multi-task representations with pre-finetuning. In: Moens, M., Huang, X., Specia, L., Yih, S.W. (eds.) Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7–11 Nov 2021, pp. 5799–5811. Association for Computational Linguistics (2021). https://doi.org/10.18653/v1/2021.emnlp-main.468
2. van Aken, B., Winter, B., Löser, A., Gers, F.A.: How does BERT answer questions?: A layer-wise analysis of transformer representations. In: Zhu, W., et al. (eds.) Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China,3–7 Nov 2019, pp. 1823–1832. ACM (2019). https://doi.org/10.1145/3357384.3358028
3. Anand, A., Leonhardt, J., Rudra, K., Anand, A.: Supervised contrastive learning approach for contextual ranking. In: Crestani, F., Pasi, G., Gaussier, É. (eds.) ICTIR 2022: The 2022 ACM SIGIR International Conference on the Theory of Information Retrieval, Madrid, Spain, 11–12 July 2022, pp. 61–71. ACM (2022). https://doi.org/10.1145/3539813.3545139
4. Anand, A., Lyu, L., Idahl, M., Wang, Y., Wallat, J., Zhang, Z.: Explainable information retrieval: a survey. CoRR abs/2211.02405 (2022). https://arxiv.org/abs/2211.02405
5. Belinkov, Y.: Probing classifiers: promises, shortcomings, and advances. Comput. Linguist. **48**(1), 207–219 (2022). https://doi.org/10.1162/coli_a_00422
6. Belinkov, Y., Glass, J.: Analysis methods in neural language processing: a survey. Trans. Assoc. Comput. Linguist. **7**, 49–72 (2019). https://doi.org/10.1162/tacl_a_00254. https://www.aclweb.org/anthology/Q19-1004
7. Câmara, A., Hauff, C.: Diagnosing bert with retrieval heuristics. In: Jose, J.M., et al. (eds.) ECIR 2020. LNCS, vol. 12035, pp. 605–618. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-45439-5_40
8. Caruana, R.: Multitask learning. Mach. Learn. **28**(1), 41–75 (1997). https://doi.org/10.1023/A:1007379606734
9. Choi, J., Jung, E., Lim, S., Rhee, W.: Finding inverse document frequency information in BERT. CoRR abs/2202.12191 (2022). https://arxiv.org/abs/2202.12191
10. Clarke, C.L.A., Kolla, M., Vechtomova, O.: An effectiveness measure for ambiguous and underspecified queries. In: Azzopardi, L., et al. (eds.) ICTIR 2009. LNCS, vol. 5766, pp. 188–199. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04417-5_17
11. Conneau, A., Kruszewski, G., Lample, G., Barrault, L., Baroni, M.: What you can cram into a single $&!#* vector: probing sentence embeddings for linguistic properties. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, 15–20 July 2018, Volume 1: Long Papers, pp. 2126–2136. Association for Computational Linguistics (2018). https://doi.org/10.18653/v1/P18-1198. https://www.aclweb.org/anthology/P18-1198/

12. Craswell, N., Mitra, B., Yilmaz, E., Campos, D., Voorhees, E.M.: Overview of the TREC 2019 deep learning track. CoRR abs/2003.07820 (2020). https://arxiv.org/abs/2003.07820
13. Dehghani, M., Zamani, H., Severyn, A., Kamps, J., Croft, W.B.: Neural ranking models with weak supervision. In: Kando, N., Sakai, T., Joho, H., Li, H., de Vries, A.P., White, R.W. (eds.) Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, 7–11 August 2017, pp. 65–74. ACM (2017). https://doi.org/10.1145/3077136.3080832
14. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: 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), pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (2019). https://doi.org/10.18653/v1/N19-1423. https://www.aclweb.org/anthology/N19-1423
15. Elazar, Y., Ravfogel, S., Jacovi, A., Goldberg, Y.: Amnesic probing: behavioral explanation with amnesic counterfactuals. Trans. Assoc. Comput. Linguistics **9**, 160–175 (2021). https://doi.org/10.1162/tacl_a_00359. https://doi.org/10.1162/tacl_a_00359
16. Fan, Y., Guo, J., Ma, X., Zhang, R., Lan, Y., Cheng, X.: A linguistic study on relevance modeling in information retrieval. In: Leskovec, J., Grobelnik, M., Najork, M., Tang, J., Zia, L. (eds.) WWW 2021: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, 19–23 Apr 2021, pp. 1053–1064. ACM / IW3C2 (2021). https://doi.org/10.1145/3442381.3450009
17. Fayyaz, M., Aghazadeh, E., Modarressi, A., Mohebbi, H., Pilehvar, M.T.: Not all models localize linguistic knowledge in the same place: a layer-wise probing on bertoids' representations. In: Bastings, J., et al. (eds.) Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP, BlackboxNLP@EMNLP 2021, Punta Cana, Dominican Republic, 11 Nov 2021, pp. 375–388. Association for Computational Linguistics (2021). https://aclanthology.org/2021.blackboxnlp-1.29
18. Formal, T., Piwowarski, B., Clinchant, S.: SPLADE: sparse lexical and expansion model for first stage ranking. In: Diaz, F., Shah, C., Suel, T., Castells, P., Jones, R., Sakai, T. (eds.) SIGIR 2021: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, 11–15 July 2021, pp. 2288–2292. ACM (2021). https://doi.org/10.1145/3404835.3463098
19. Ghasemi, N., Hiemstra, D.: BERT meets cranfield: uncovering the properties of full ranking on fully labeled data. In: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop, pp. 58–64. Association for Computational Linguistics, Online (2021). https://aclanthology.org/2021.eacl-srw.9
20. Hagen, M., Völske, M., Göring, S., Stein, B.: Axiomatic result re-ranking. In: Mukhopadhyay, S., et al.(eds.) Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, 24–28 October 2016, pp. 721–730. ACM (2016). https://doi.org/10.1145/2983323.2983704

21. Hewitt, J., Ethayarajh, K., Liang, P., Manning, C.D.: Conditional probing: measuring usable information beyond a baseline. In: Moens, M., Huang, X., Specia, L., Yih, S.W. (eds.) Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7–11 November 2021, pp. 1626–1639. Association for Computational Linguistics (2021). https://doi.org/10.18653/v1/2021.emnlp-main.122
22. Hewitt, J., Liang, P.: 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), pp. 2733–2743. Association for Computational Linguistics, Hong Kong, China (2019). https://doi.org/10.18653/v1/D19-1275. https://aclanthology.org/D19-1275
23. Hewitt, J., Manning, C.D.: A structural probe for finding syntax in word representations. 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), pp. 4129–4138. Association for Computational Linguistics, Minneapolis, Minnesota (2019). https://doi.org/10.18653/v1/N19-1419. https://www.aclweb.org/anthology/N19-1419
24. Honnibal, M., Montani, I.: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. Unpublished Software Application. https://spacy.io (2017)
25. Khalid, M.A., Jijkoun, V., de Rijke, M.: The impact of named entity normalization on information retrieval for question answering. In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 705–710. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78646-7_83
26. Khattab, O., Zaharia, M.: Colbert: efficient and effective passage search via contextualized late interaction over BERT. In: Huang, J., et al. (eds.) Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, 25–30 July 2020, pp. 39–48. ACM (2020). https://doi.org/10.1145/3397271.3401075
27. Kim, M., Ko, Y.: Multitask fine-tuning for passage re-ranking using BM25 and pseudo relevance feedback. IEEE Access **10**, 54254–54262 (2022). https://doi.org/10.1109/ACCESS.2022.3176894
28. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015, Conference Track Proceedings (2015). https://arxiv.org/abs/1412.6980
29. Leonhardt, J., Rudra, K., Khosla, M., Anand, A., Anand, A.: Efficient neural ranking using forward indexes. In: Laforest, F., et al. (eds.) WWW 2022: The ACM Web Conference 2022, Virtual Event, Lyon, France, 25–29 April 2022, pp. 266–276. ACM (2022). https://doi.org/10.1145/3485447.3511955
30. Li, H., Xu, J.: Semantic matching in search. Found. Trends Inf. Retr. **7**(5), 343–469 (2014). https://doi.org/10.1561/1500000035
31. MacAvaney, S., Feldman, S., Goharian, N., Downey, D., Cohan, A.: ABNIRML: analyzing the behavior of neural IR models. Trans. Assoc. Comput. Linguistics **10**, 224–239 (2022). https://doi.org/10.1162/tacl_a_00457
32. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to information retrieval. Cambridge University Press (2008). https://doi.org/10.1017/CBO9780511809071. https://nlp.stanford.edu/IR-book/pdf/irbookprint.pdf

33. Nogueira, R.F., Cho, K.: Passage re-ranking with BERT. CoRR abs/1901.04085 (2019). https://arxiv.org/abs/1901.04085
34. Pennington, J., Socher, R., Manning, C.: GloVe: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543. Association for Computational Linguistics, Doha, Qatar (2014). https://doi.org/10.3115/v1/D14-1162. https://aclanthology.org/D14-1162
35. Petroni, F., et al.: KILT: a benchmark for knowledge intensive language tasks. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 2523–2544. Association for Computational Linguistics, Online (2021). https://doi.org/10.18653/v1/2021.naacl-main.200. https://aclanthology.org/2021.naacl-main.200
36. Petroni, F., et al.: Language models as knowledge bases? 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), pp. 2463–2473. Association for Computational Linguistics, Hong Kong, China (2019). https://doi.org/10.18653/v1/D19-1250. https://aclanthology.org/D19-1250
37. Pimentel, T., Saphra, N., Williams, A., Cotterell, R.: Pareto probing: trading off accuracy for complexity. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 3138–3153. Association for Computational Linguistics, Online (2020). https://doi.org/10.18653/v1/2020.emnlp-main.254. https://www.aclweb.org/anthology/2020.emnlp-main.254
38. Qiao, Y., Xiong, C., Liu, Z., Liu, Z.: Understanding the behaviors of BERT in ranking. CoRR abs/1904.07531 (2019). https://arxiv.org/abs/1904.07531
39. Rau, D., Kamps, J.: The role of complex NLP in transformers for text ranking. In: Crestani, F., Pasi, G., Gaussier, É. (eds.) ICTIR 2022: The 2022 ACM SIGIR International Conference on the Theory of Information Retrieval, Madrid, Spain, 11–12 July 2022, pp. 153–160. ACM (2022). https://doi.org/10.1145/3539813.3545144
40. Rennings, D., Lyu, L., Anand, A.: Listwise explanations for ranking models using multiple explainers. In: Advances in Information Retrieval - 45th European Conference on IR Research, ECIR 2023, Dublin, Ireland, Proceedings, Part I. Lecture Notes in Computer Science, Springer (2023)
41. Rennings, D., Moraes, F., Hauff, C.: An axiomatic approach to diagnosing neural IR models. In: Azzopardi, L., Stein, B., Fuhr, N., Mayr, P., Hauff, C., Hiemstra, D. (eds.) ECIR 2019. LNCS, vol. 11437, pp. 489–503. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-15712-8_32
42. Rissanen, J.: Universal coding, information, prediction, and estimation. IEEE Trans. Inf. Theory **30**(4), 629–636 (1984). https://doi.org/10.1109/TIT.1984.1056936
43. Robertson, S.E., Walker, S., Jones, S., Hancock-Beaulieu, M., Gatford, M.: Okapi at TREC-3. In: Harman, D.K. (ed.) Proceedings of The Third Text REtrieval Conference, TREC 1994, Gaithersburg, Maryland, USA, 2–4 Nov 1994. NIST Special Publication, vol. 500–225, pp. 109–126. National Institute of Standards and Technology (NIST) (1994). https://trec.nist.gov/pubs/trec3/papers/city.ps.gz
44. Rudra, K., Anand, A.: Distant supervision in BERT-based Adhoc document retrieval. In: d'Aquin, M., Dietze, S., Hauff, C., Curry, E., Cudré-Mauroux, P. (eds.) CIKM 2020: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, 19–23 October 2020, pp. 2197–2200. ACM (2020). https://doi.org/10.1145/3340531.3412124

45. Singh, J., Anand, A.: EXS: explainable search using local model agnostic interpretability. In: Culpepper, J.S., Moffat, A., Bennett, P.N., Lerman, K. (eds.) Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM 2019, Melbourne, VIC, Australia, 11–15 Feb 2019, pp. 770–773. ACM (2019). https://doi.org/10.1145/3289600.3290620
46. Singh, J., Anand, A.: Model agnostic interpretability of rankers via intent modelling. In: Hildebrandt, M., Castillo, C., Celis, L.E., Ruggieri, S., Taylor, L., Zanfir-Fortuna, G. (eds.) FAT* 2020: Conference on Fairness, Accountability, and Transparency, Barcelona, Spain, 27–30 Jan 2020, pp. 618–628. ACM (2020). https://doi.org/10.1145/3351095.3375234
47. Tenney, I., Das, D., Pavlick, E.: BERT rediscovers the classical NLP pipeline. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 4593–4601. Association for Computational Linguistics, Florence, Italy (2019). https://doi.org/10.18653/v1/P19-1452. https://www.aclweb.org/anthology/P19-1452
48. Tenney, I., et al.: 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, 6–9 May 2019. OpenReview.net (2019). https://openreview.net/forum?id=SJzSgnRcKX
49. Thorne, J., Vlachos, A., Christodoulopoulos, C., Mittal, A.: FEVER: a large-scale dataset for fact extraction and Verification. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pp. 809–819. Association for Computational Linguistics, New Orleans, Louisiana (2018). https://doi.org/10.18653/v1/N18-1074. https://aclanthology.org/N18-1074
50. Voita, E., Titov, I.: Information-theoretic probing with minimum description length. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 183–196. Association for Computational Linguistics, Online (2020). https://doi.org/10.18653/v1/2020.emnlp-main.14. https://www.aclweb.org/anthology/2020.emnlp-main.14
51. Völske, M., et al.: Towards axiomatic explanations for neural ranking models. In: Hasibi, F., Fang, Y., Aizawa, A. (eds.) ICTIR 2021: The 2021 ACM SIGIR International Conference on the Theory of Information Retrieval, Virtual Event, Canada, 11 July 2021, pp. 13–22. ACM (2021). https://doi.org/10.1145/3471158.3472256
52. Wallat, J., Singh, J., Anand, A.: BERTnesia: investigating the capture and forgetting of knowledge in BERT. In: Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP, pp. 174–183. Association for Computational Linguistics, Online (2020). https://doi.org/10.18653/v1/2020.blackboxnlp-1.17. https://aclanthology.org/2020.blackboxnlp-1.17
53. Zhan, J., Mao, J., Liu, Y., Zhang, M., Ma, S.: An analysis of BERT in document ranking. In: Huang, J., et al. (eds.) Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, 25–30 July 2020, pp. 1941–1944. ACM (2020). https://doi.org/10.1145/3397271.3401325
54. Zhang, K., Bowman, S.: Language modeling teaches you more than translation does: Lessons learned through auxiliary syntactic task analysis. In: Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pp. 359–361. Association for Computational Linguistics, Brussels, Belgium (2018). https://doi.org/10.18653/v1/W18-5448. https://www.aclweb.org/anthology/W18-5448

55. Zhang, X., Ramachandran, D., Tenney, I., Elazar, Y., Roth, D.: Do language embeddings capture scales? In: Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP, pp. 292–299. Association for Computational Linguistics, Online (2020). https://doi.org/10.18653/v1/2020.blackboxnlp-1.27. https://aclanthology.org/2020.blackboxnlp-1.27
56. Zhao, M., Dufter, P., Yaghoobzadeh, Y., Schütze, H.: Quantifying the contextualization of word representations with semantic class probing. In: Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 1219–1234. Association for Computational Linguistics, Online (2020). https://doi.org/10.18653/v1/2020.findings-emnlp.109. https://www.aclweb.org/anthology/2020.findings-emnlp.109