How Does BERT Answer Questions? A Layer-Wise Analysis of Transformer Representations

Betty van Aken, Benjamin Winter, Alexander Löser, Felix A. Gers

CIKM 2019

February 14, 2020

Presenter: Rishab Bamrara

https://qdata.github.io/deep2Read/

Motivation:

- Bidirectional Encoder Representations from Transformers (BERT) reach state-of-the-art results in a variety of Natural Language Processing tasks.

- However, understanding of their internal functioning is still insufficient and unsatisfactory.

- Hence most of the times these deep learning models are treated as black box as they lack transparency, reliability and prediction guaranty.

- Transformers are moderately interpretable by their attention values, however this may not always be the case.

Related Work:

1. Tenney et al. : a novel **"edge-probing"** framework (9 tasks)

2. Yoav Goldberg. 2019. Assessing BERT’s Syntactic Abilities (**More tasks**)

3. Qiao et al. : focus specifically on analysing BERT as a **Ranking model**.

4. Zhang and Zhu, **Visual interpretability** for deep learning: limited to CNNs

5. Liu et al. :perform a **layer-wise analysis** of BERT’s token representations.

Background:

- **Edge Probing:** Translates core NLP tasks into classification tasks by focusing solely on their labeling part.

- **Named Entity Labeling (NEL):** Given a span of tokens the model has to predict the correct entity category.

- **Coreference Resolution:** Predict whether two mentions within a text refer to the same entity.

- **Relation Classification:** Predict which relation type connects two known entities.

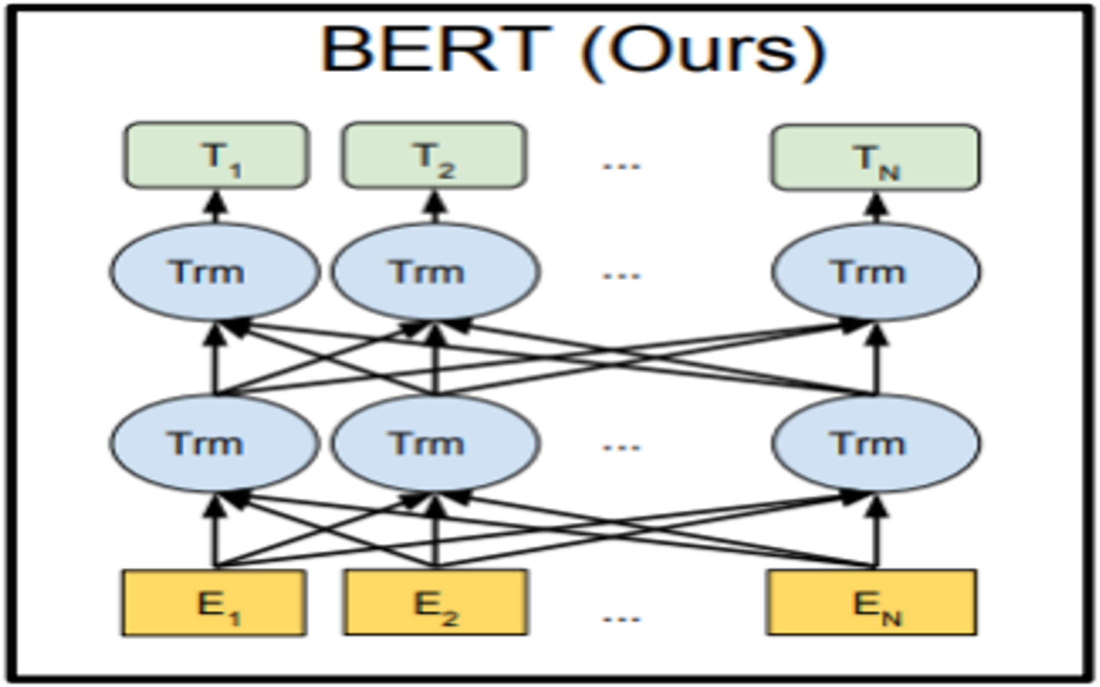
- **Question Type Classification:** Correctly identify question type.

- **Supporting Facts:** Predict whether a sentence contains supporting facts regarding a specific question or whether it is irrelevant.

- **Dimensionality Reduction:** Process of reducing the number of random variables under consideration. (t-SNE, PCA, ICA).

Background:

- **K-Means Clustering:** Clustering based on mean.
- **BERT:** A method of pre-training language representations.



BERT (Bidirectional Encoder Representation for Transformers): Pre-training of Deep Bidirectional Transformers for Language Understanding

Claim / Target Task:

Unlike previous research, which mainly focuses on explaining Transformer models by their attention weights, authors argue that hidden states contain equally valuable information.

Proposed Solution:

1. Embed input tokens for each probing task sample with fine-tuned BERT model. Every layer is taken into account.

1. Use only the output embedding from n-th layer at step n.

1. Tokens are first pooled for a fixed-length representation.

1. Feed tokens into a two-layer Multi-layer Perceptron (MLP) classifier, that predicts label-wise probability scores

Proposed Solution (Fig.):

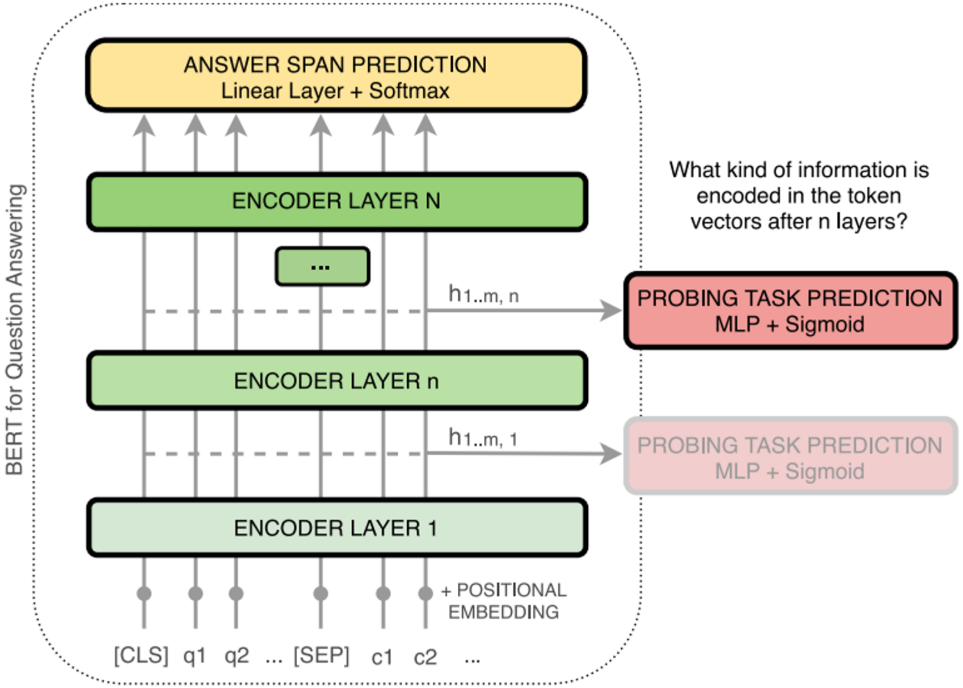


Figure 1: Schematic overview of the BERT architecture and our probing setup. Question and context tokens are processed by N encoder blocks with a Positional Embedding added beforehand. The output of the last layer is fed into a span prediction head consisting of a Linear Layer and a Softmax. We use the hidden states of each layer as input to a set of probing tasks to examine the encoded information.

Datasets:

1. **SQuAD 1.1:** Contains 100,000 natural question-answer pairs on 500 Wikipedia articles. Don't use version 2.0 as it contains some unanswerable questions as well.

1. **HotpotQA:** This Multihop QA task contains 112,000 natural question-answer pairs. The questions are especially designed to combine information from multiple parts of a context.

1. **bAbI**: Set of artificial toy tasks developed to further understand the abilities of neural models. The 20 tasks require reasoning over multiple sentences (Multihop QA) and are modeled to include Positional Reasoning.

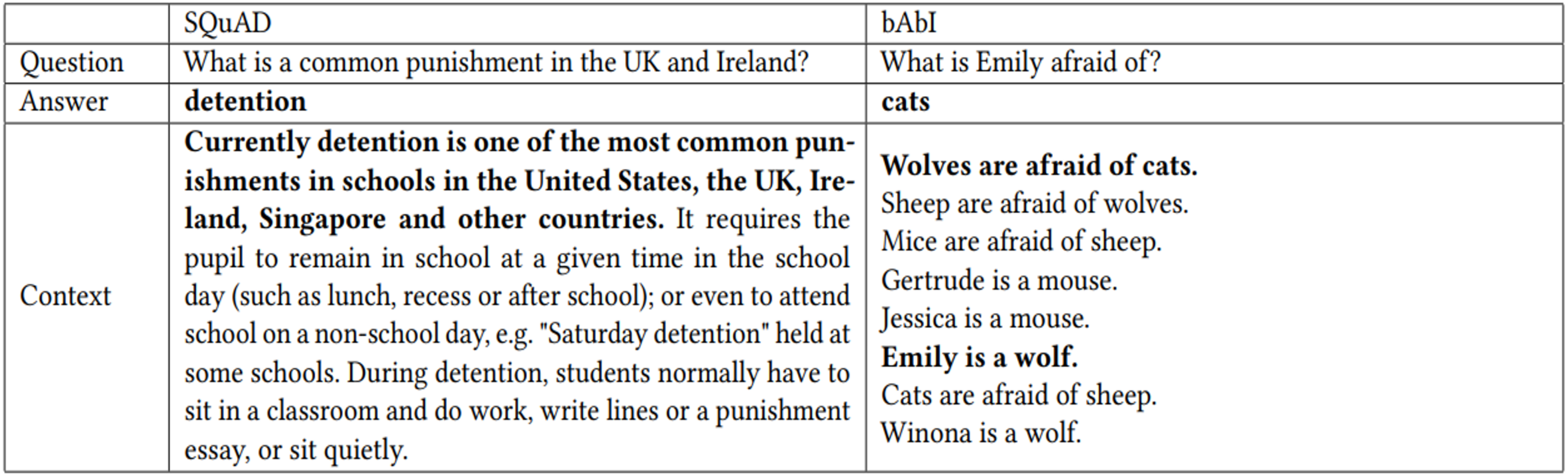
Models:

1. **BERT:** 12 transformer blocks for base and 24 for large.

1. **GPT-2 (small):** 12 transformer blocks. Large was not released.

- Both are fine tuned on each of the datasets before applying probing to QA.

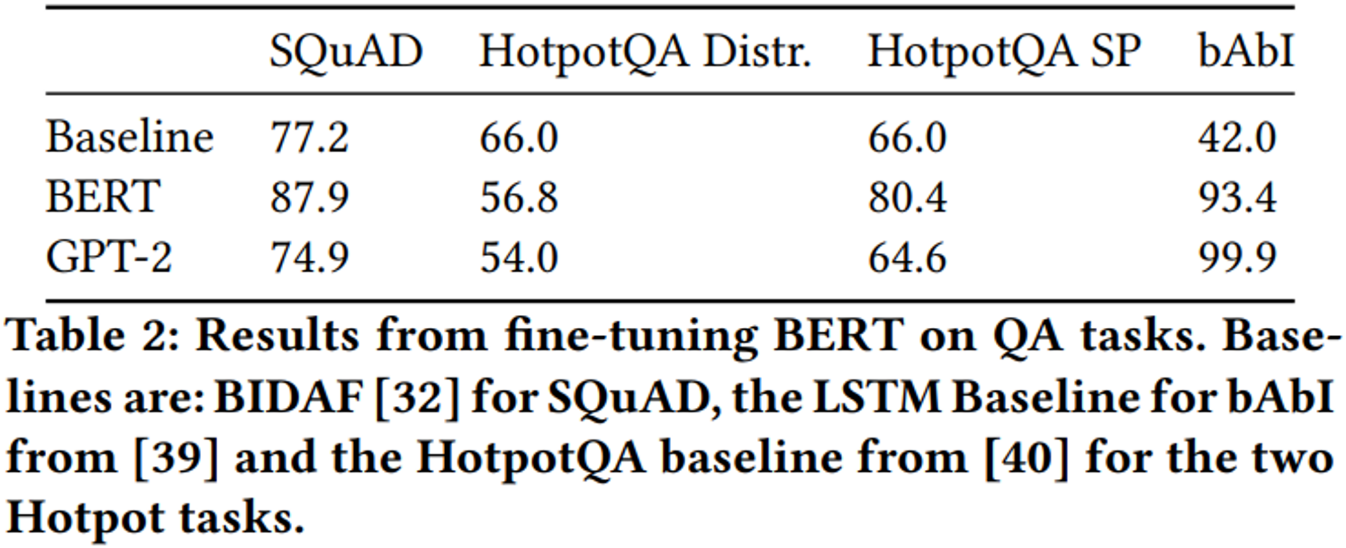
Results and Discussion:



	SQuAD	bAbI
Question	What is a common punishment in the UK and Ireland?	What is Emily afraid of?
Answer	**detention**	**cats**
Context	**Currently detention is one of the most common punishments in schools in the United States, the UK, Ireland, Singapore and other countries.** It requires the pupil to remain in school at a given time in the school day (such as lunch, recess or after school); or even to attend school on a non-school day, e.g. "Saturday detention" held at some schools. During detention, students normally have to sit in a classroom and do work, write lines or a punishment essay, or sit quietly.	**Wolves are afraid of cats.** Sheep are afraid of wolves. Mice are afraid of sheep. Gertrude is a mouse. Jessica is a mouse. **Emily is a wolf.** Cats are afraid of sheep. Winona is a wolf.

Table 1: Samples from SQuAD dataset (left) and from Basic Deduction task (#15) of the bAbI dataset (right). Supporting Facts are printed in bold. The SQuAD sample can be solved by word matching and entity resolution, while the bAbI sample requires a logical reasoning step and cannot be solved by simple word matching. Figures in the further analysis will use these examples where applicable.

Results and Discussion:



	SQuAD	HotpotQA Distr.	HotpotQA SP	bAbI
Baseline	77.2	66.0	66.0	42.0
BERT	87.9	56.8	80.4	93.4
GPT-2	74.9	54.0	64.6	99.9

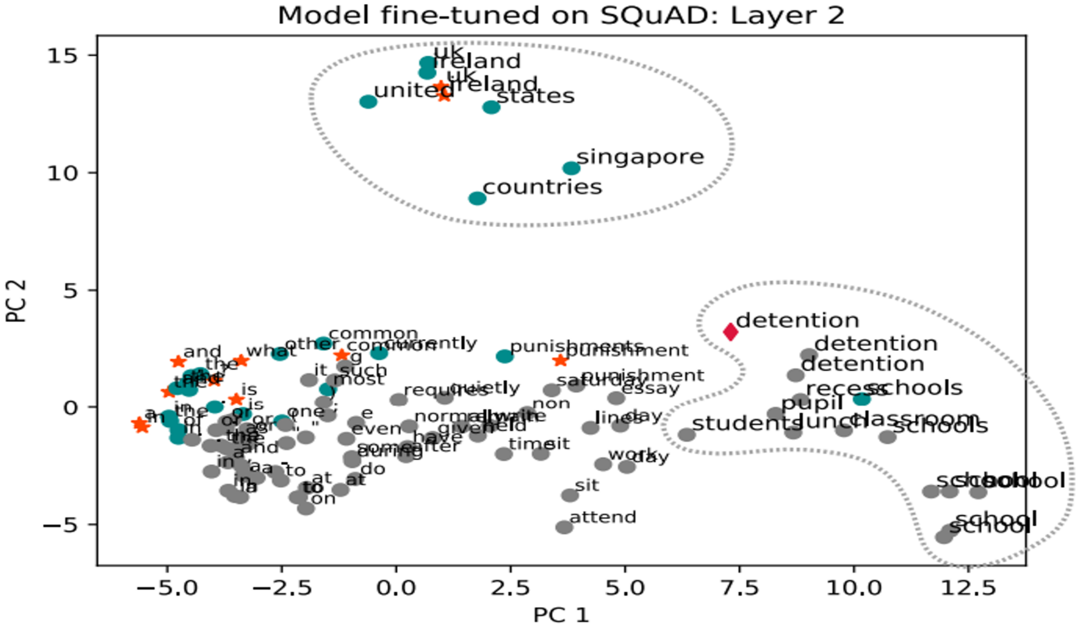
Table 2: Results from fine-tuning BERT on QA tasks. Baselines are: BIDAF [32] for SQuAD, the LSTM Baseline for bAbI from [39] and the HotpotQA baseline from [40] for the two Hotpot tasks.

- Accuracy on the SQuAD task is close to human performance.
- Tasks derived from HotpotQA prove much more challenging.
- bAbI was easily solved by both BERT and GPT-2. But, GPT-2 performed better.
- Most of BERT's error in the bAbI multi-task setting comes from tasks that require positional or geometric reasoning, this is a skill where GPT-2 is better than BERT's reasoning capabilities

Results and Discussion:

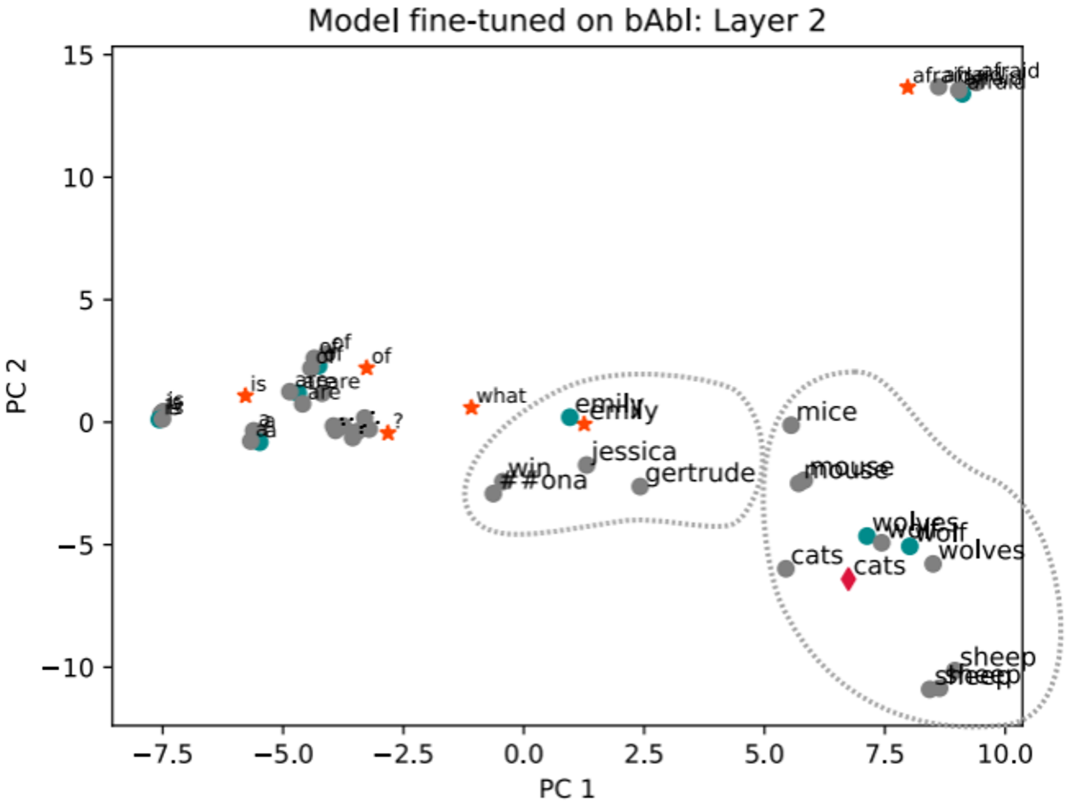
The PCA representations of tokens in different layers suggest that the model is going through multiple phases while answering a question.

1. **Semantic Clustering:** Early layers within the BERT-based models group tokens into topical clusters. Therefore, these initial layers reach low accuracy on semantic probing tasks.



(a) SQuAD Phase 1: Semantic Clustering. We observe a topical cluster with 'school'-related and another with 'country'-related tokens.

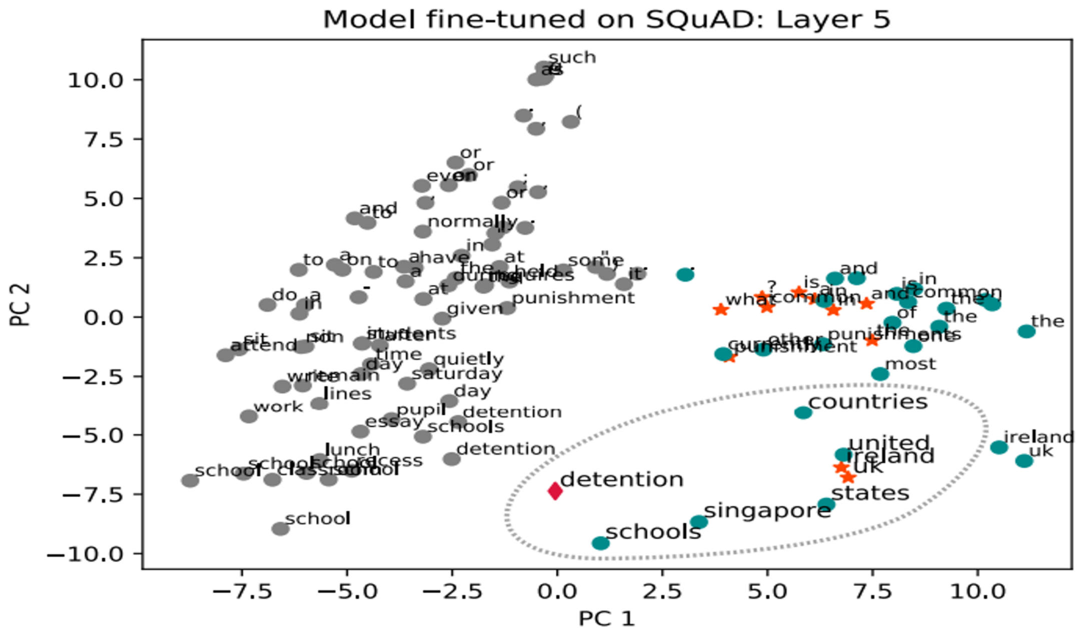
Results and Discussion:



(a) bAbI Phase 1: Semantic Clustering. Names and animals are clustered.

Results and Discussion:

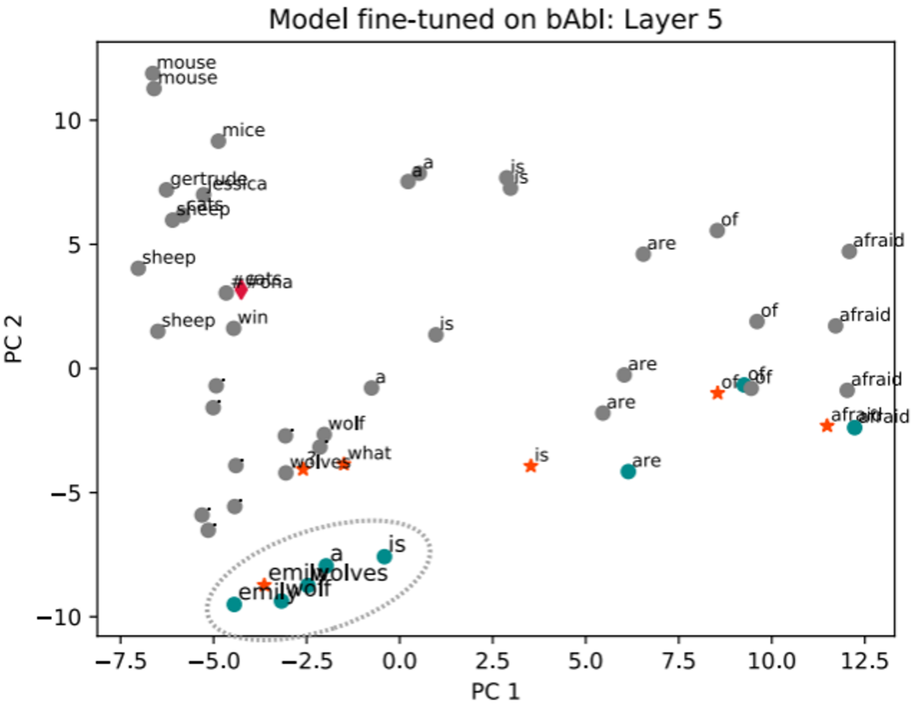
2. **Connecting Entities with Mentions and Attributes:** In the middle layers of the observed networks clusters of entities are less connected by their topical similarity rather, they are connected by their relation within a certain input context.



(b) SQuAD Phase 2: Entity Matching. The marked cluster contains matched tokens ’detention’, ’schools’ and the countries that are applying this practice.

This cluster helps to solve the question "What is a common punishment in the UK and Ireland?".

Results and Discussion:

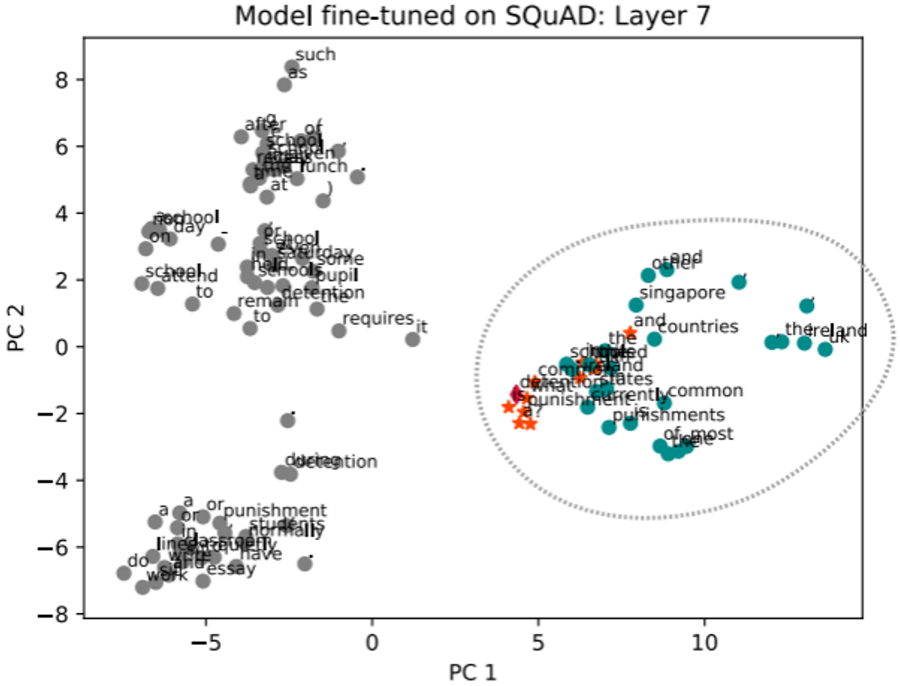


(b) bAbI Phase 2: Entity Matching. The determining relation between the entities 'Emily' and 'Wolf' is resolved in a cluster.

Challenge within this sample is to identify the two facts that *Emily is a wolf* and *Wolves are afraid of cats.*

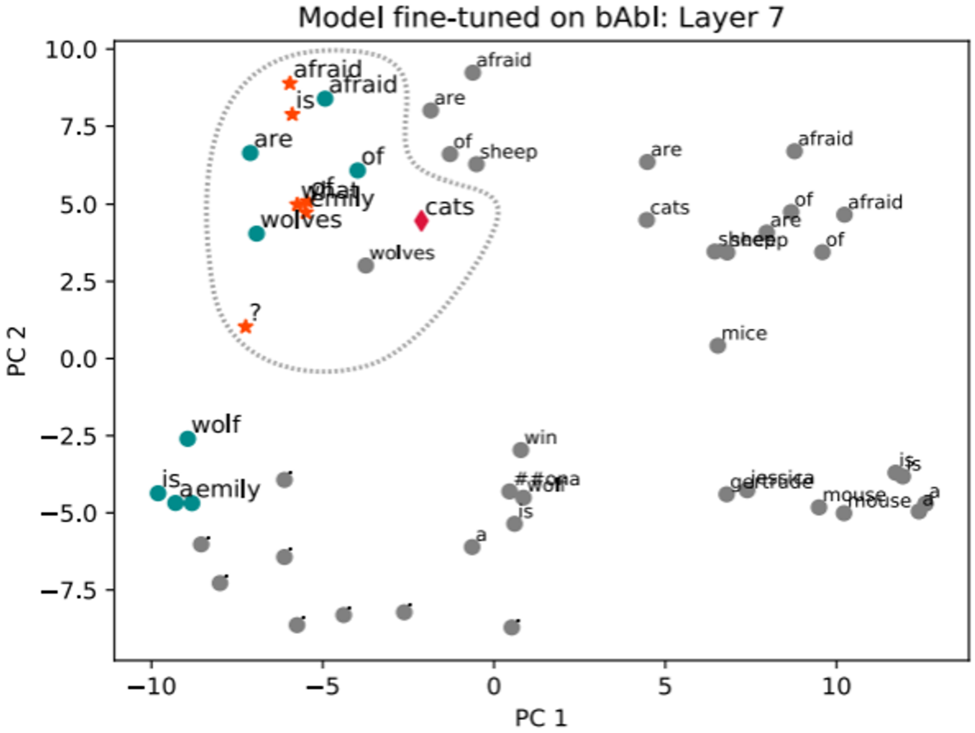
Results and Discussion:

3. Matching Questions with Supporting Facts: Identifying relevant parts of the context is crucial for QA and Information Retrieval in general. BERT models perform a comparable step by transforming the tokens so that question tokens are matched onto relevant context tokens.



(c) SQuAD Phase 3: Question-Fact Matching. The question tokens form a cluster with the Supporting Fact tokens.

Results and Discussion:

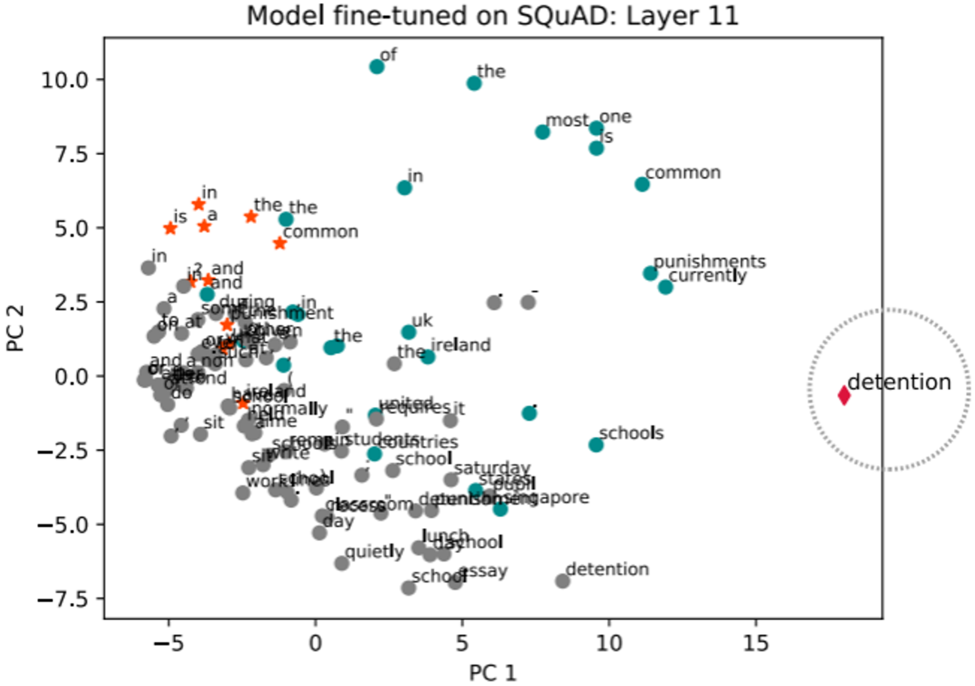


(c) bAbI Phase 3: Question-Fact Matching. In this case the question tokens match with a subset of Supporting Facts ('Wolves are afraid of cats'). The subset is decisive of the answer.

- Model transforms the token representation of question and Supporting Facts into the same area of the vector space.

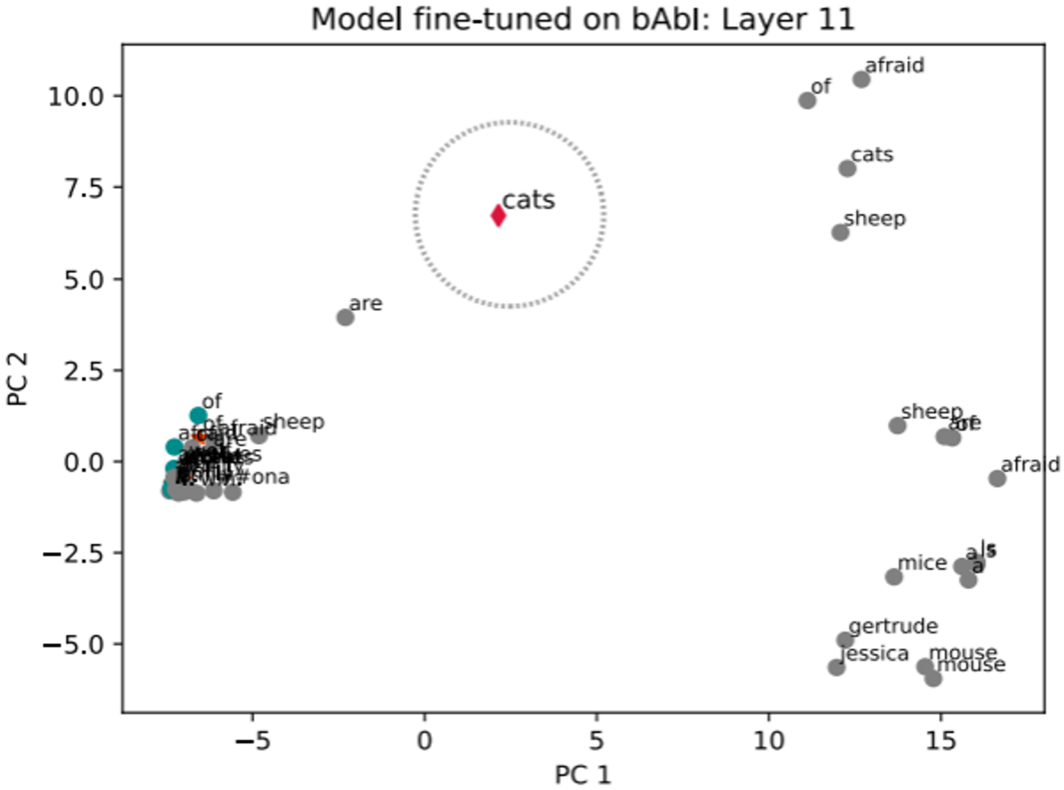
Results and Discussion:

4. Answer Extraction: In the last network layers the model dissolves most of the previous clusters. Model separates the correct answer tokens, and sometimes other possible candidates, from the rest of the tokens. The remaining tokens form one or multiple homogeneous clusters. The vector representation at this point is largely task-specific and learned during fine-tuning.


Model fine-tuned on SQuAD: Layer 11

(d) SQuAD Phase 4: Answer Extraction. The answer token ’detention’ is separated from other tokens.

Results and Discussion:



(d) bAbI Phase 4: Answer Extraction. The answer token 'cats' is separated from other tokens.

Results and Discussion:

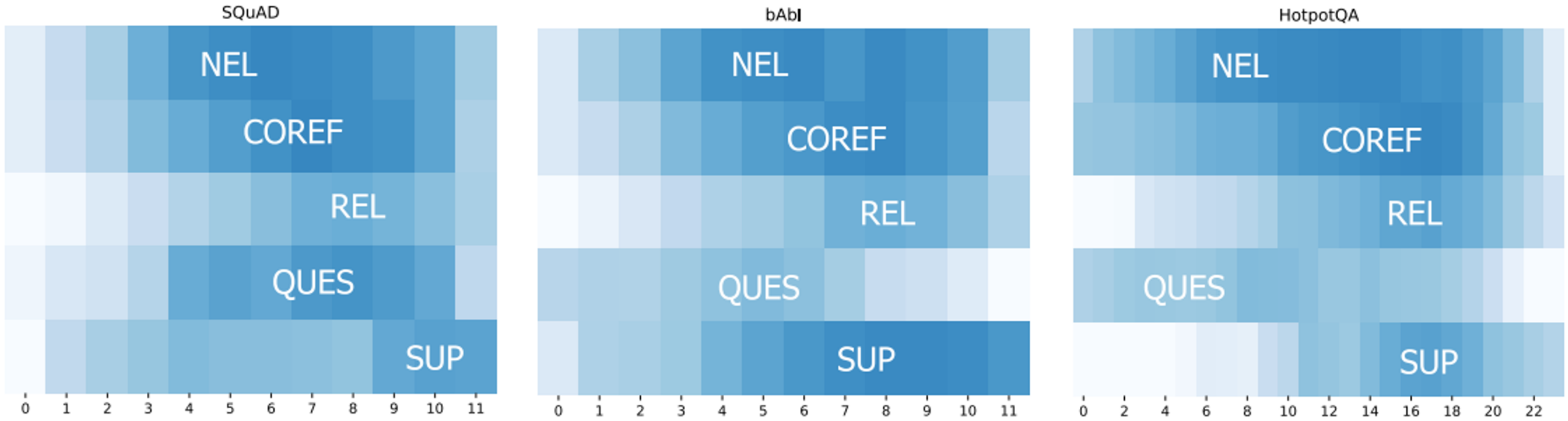
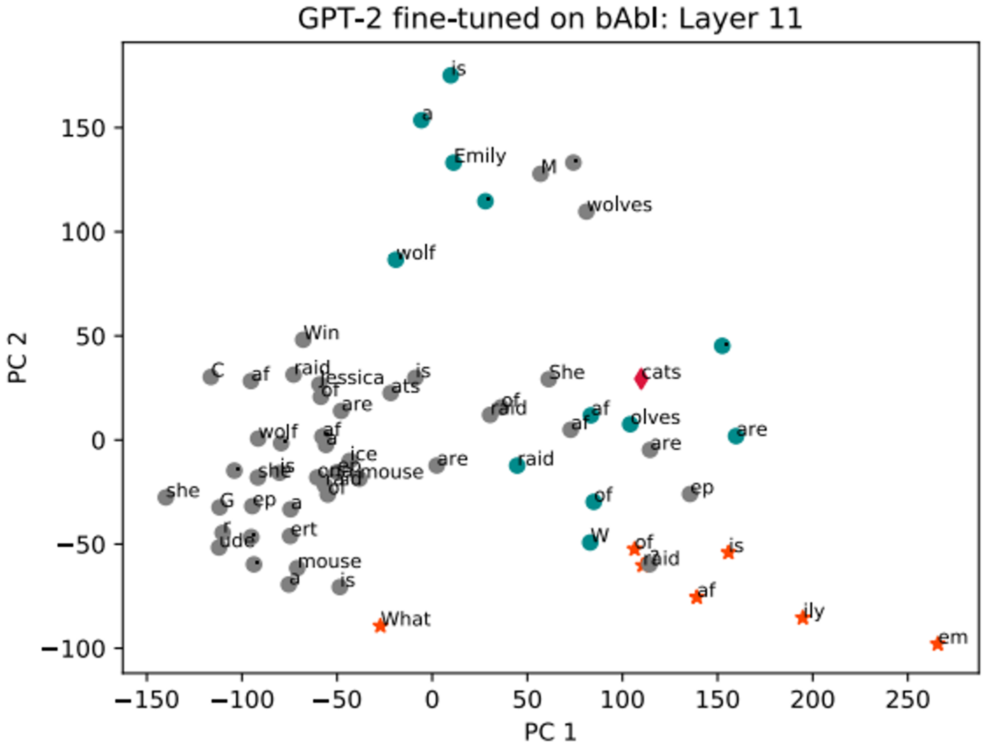


Figure 6: Phases of BERT's language abilities. Higher saturation denotes higher accuracy on probing tasks. Values are normalized over tasks on the Y-axis. X-axis depicts layers of BERT. NEL: Named Entity Labeling, COREF: Coreference Resolution, REL: Relation Classification, QUES: Question Type Classification, SUP: Supporting Fact Extraction. All three tasks exhibit similar patterns, except from QUES, which is solved earlier by the HotpotQA model based on BERT-large. NEL is solved first, while performance on COREF and REL peaks in later layers. Distinction of important facts (SUP) happens within the last layers.

Comparison to GPT-2:



GPT-2 fine-tuned on bAbI: Layer 11

Figure 7: bAbI Example of the Answer Extraction phase in GPT-2. Both the question and Supporting Fact are extracted, but the correct answer is not fully separated as in BERT's last layers. Also a potential candidate Supporting Fact in "Sheep are afraid of Wolves" is separated as well.

Observation of Failure States:

Rough difficulty of a specific task can be discerned by a glance at the hidden state representations. While for correct predictions the transformations run through the phases discussed in previous sections, for wrong predictions there is a possibility that:

1. If a candidate answer was found that the network has a reasonable amount of confidence in, the phases will look very similar to a correct prediction, but now centering on the wrong answer.

- Inspecting early layers in this case can give insights towards the reason why the wrong candidate was chosen, e.g. wrong Supporting Fact selected, mis-resolution of co-references etc.

Observation of Failure States:

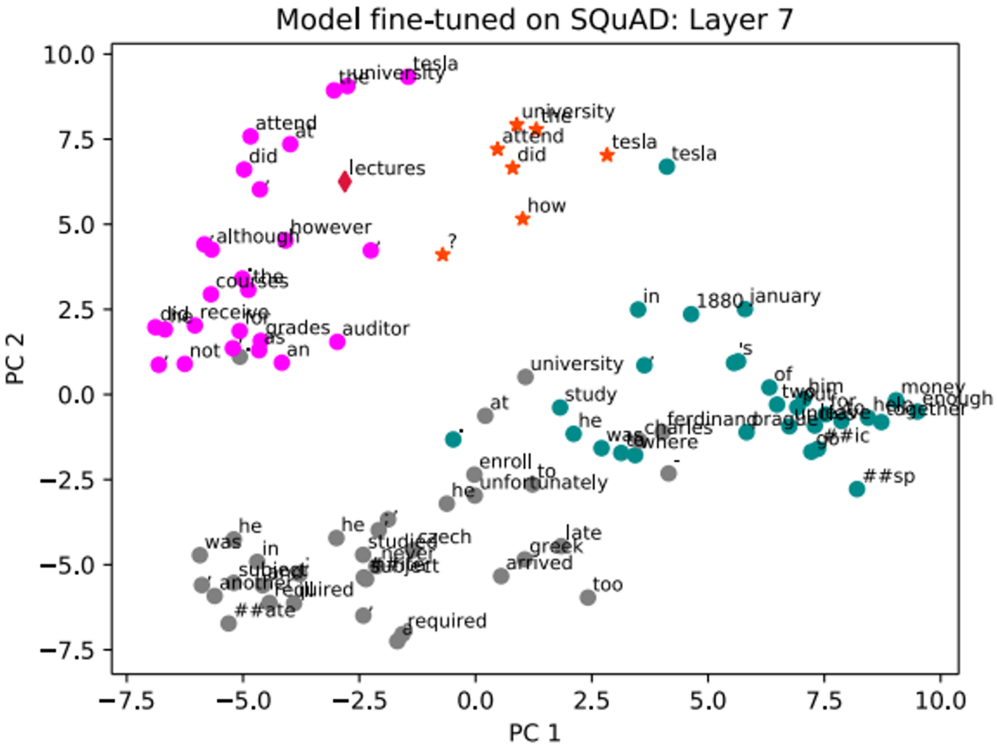


Figure 8: BERT SQuAD example of a falsely selected answer based on the matching of the wrong Supporting Fact. The predicted answer 'lectures' is matched onto the question as a part of this incorrect fact (magenta), while the actual Supporting Fact (cyan) is not particularly separated.

Maintained Positional Embedding:

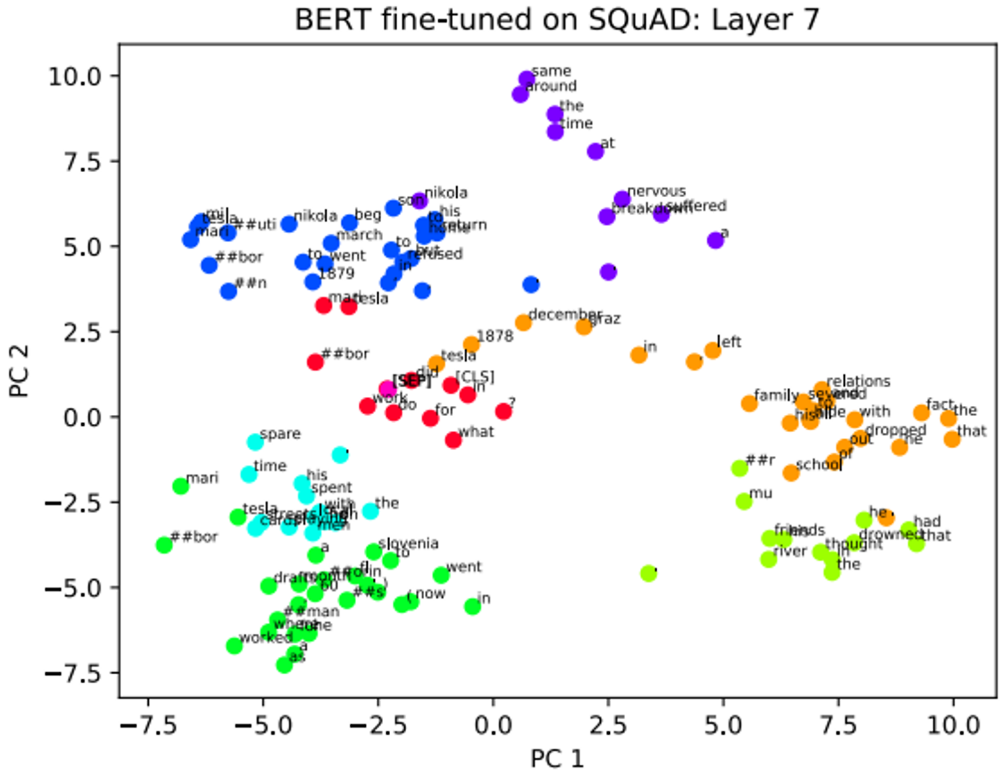


Figure 9: BERT SQuAD example Layer 7. Tokens are color-coded by sentence. This visualization shows that tokens are clustered by their original sentence membership suggesting far reaching importance of the positional embedding.

Abilities to resolve Question Type:

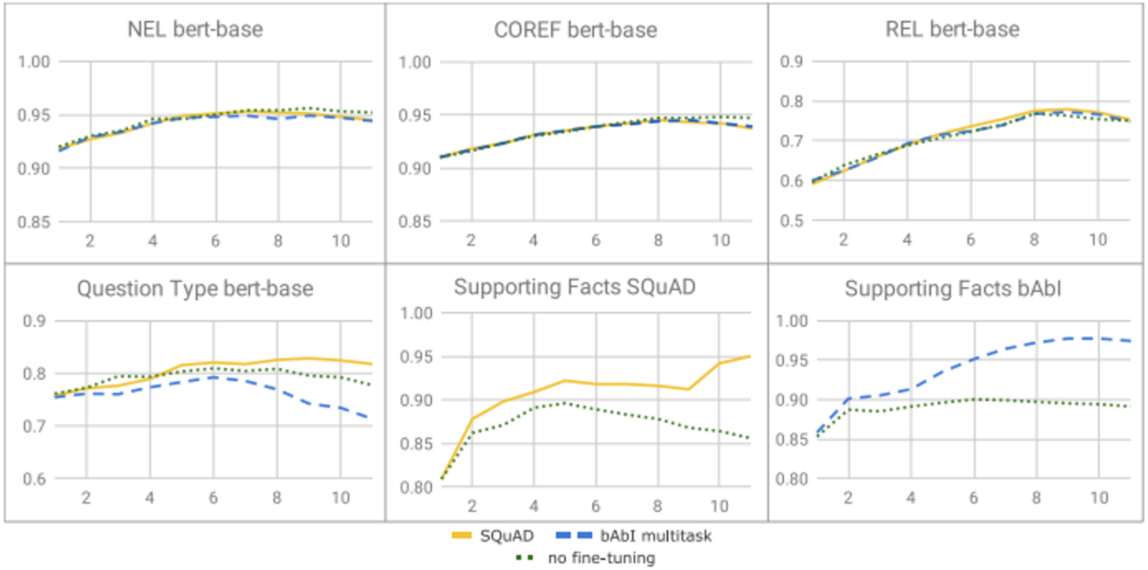


Figure 2: Probing Task results of BERT-base models in macro averaged F1 (Y-axis) over all layers (X-axis). Fine-tuning barely affects accuracy on NEL, COREF and REL indicating that those tasks are already sufficiently covered by pre-training. Performances on the Question Type task shows its relevancy for solving SQuAD, whereas it is not required for the bAbI tasks and the information is lost.

Model fine-tuned on the bAbI tasks, loose part of its ability to distinguish question types during fine-tuning. This is likely caused by the static structure of bAbI samples, in which the answer candidates can be recognized by sentence structure and occurring word patterns rather than by the question type.

Abilities to resolve Question Type:

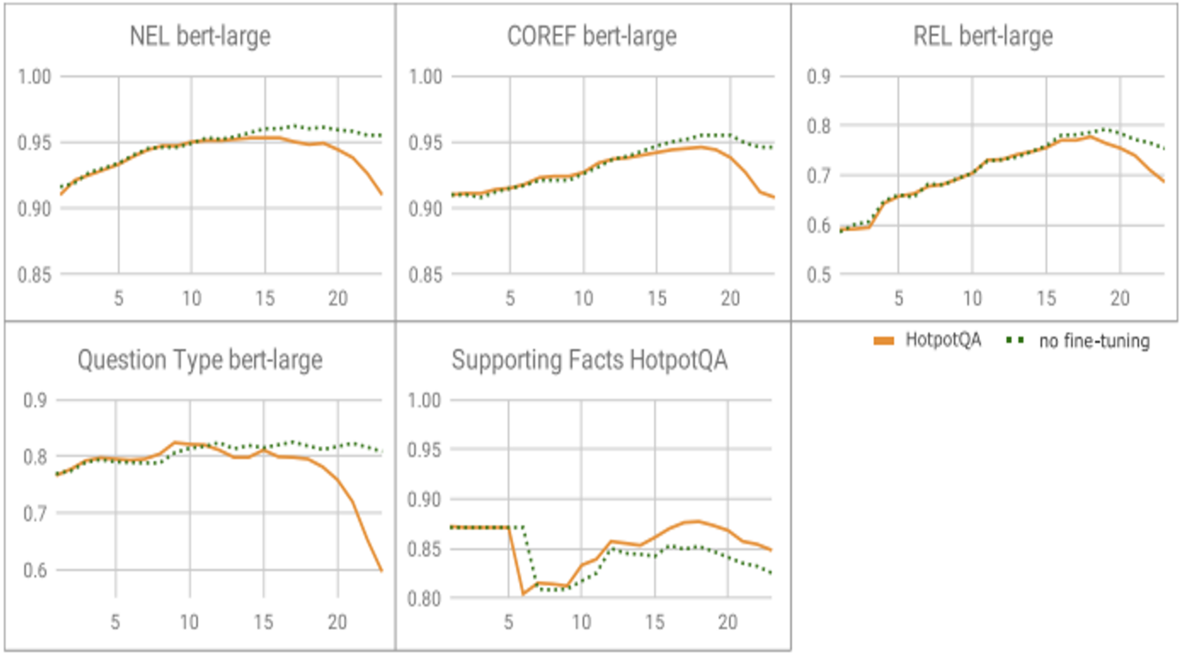


Figure 3: Probing Task results of BERT-large models in macro averaged F1 (Y-axis) over all layers (X-axis). Performance of HotpotQA model is mostly equal to the model without fine-tuning, but information is dropped in last layers in order to fit the Answer Selection task.

Surprisingly, model fine-tuned on HotpotQA does not outperform the model without fine-tuning. Both models can solve the task in earlier layers, which suggests that the ability to recognize question types is pre-trained in BERT-large.

Conclusion and Future Work:

Work reveals important findings about the inner functioning of Transformer networks.

Interpretability: The qualitative analysis of token vectors reveals that there is indeed interpretable information stored within the hidden states of Transformer models.

Transferability: We further show that lower layers might be more applicable to certain problems than later ones.

Modularity: Our findings support the hypothesis that not only do different phases exist in Transformer networks, but that specific layers seem to solve different problems. This hints at a modularity that can potentially be exploited in the training process

References:

- Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Sam Bowman, Dipanjan Das, and Ellie Pavlick. 2019. What do you learn from context? Probing for sentence structure in contextualized word representations. In Proceedings of ICLR 2019.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. In Proceedings of EMNLP 2016.
- Yifan Qiao, Chenyan Xiong, Zheng-Hao Liu, and Zhiyuan Liu. 2019. Understanding the Behaviors of BERT in Ranking. CoRR (2019).
- Yoav Goldberg. 2019. Assessing BERT's Syntactic Abilities. CoRR (2019).
- Quan-shi Zhang and Song-chun Zhu. 2018. Visual interpretability for deep learning: a survey. Frontiers of IT & EE (2018).
- Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew Peters, and Noah A. Smith. 2019. Linguistic Knowledge and Transferability of Contextual Representations. In Proceedings of NAACL 2019.
- Stuart P. Lloyd. 1982. Least squares quantization in PCM. IEEE Trans. Information Theory (1982).
- Dieuwke Hupkes, Sara Veldhoen, and Willem H. Zuidema. 2017. Visualisation and 'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. In Proceedings of IJCAI 2018.