

Does BERT Solve Commonsense Task via Commonsense Knowledge?

Leyang Cui[†], Sijie Cheng[‡], Yu Wu[◇], Yue Zhang[†]

[†]Westlake University

[‡]Fudan University

[◇]Microsoft Research Asia

[†]cuileyang@westlake.edu.cn

Abstract

The success of pre-trained contextualized language models such as BERT motivates a line of work that investigates linguistic knowledge inside such models in order to explain the huge improvement in downstream tasks. While previous work shows syntactic, semantic and word sense knowledge in BERT, little work has been done on investigating how BERT solves CommonsenseQA tasks. In particular, it is an interesting research question whether BERT relies on shallow syntactic patterns or deeper commonsense knowledge for disambiguation. We propose two attention-based methods to analyze commonsense knowledge inside BERT, and the contribution of such knowledge for the model prediction. We find that attention heads successfully capture the structured commonsense knowledge encoded in CONCEPTNET, which helps BERT solve commonsense tasks directly. Fine-tuning further makes BERT learn to use the commonsense knowledge on higher layers.

1 Introduction

Pre-trained language models (Peters et al., 2018; Radford et al., 2019; Devlin et al., 2019; Liu et al., 2019b) achieve highly competitive results on a variety of downstream NLP tasks (Zhou and Zhao, 2019; Joshi et al., 2019; Liu and Lapata, 2019; Cui et al., 2020). Previous work shows that they effectively capture syntactic information (Goldberg, 2019), semantic information (Liu et al., 2019a) and factual knowledge (Petroni et al., 2019), without fine-tuning on task-specific datasets, which provides strong support for the success in downstream tasks. Recently, there has been some debate about whether commonsense knowledge can be learned by leveraging a language model trained on large corpora. While Davison et al. (2019), Bosselut et al. (2019) and Rajani et al. (2019) argue that pre-trained language models can identify

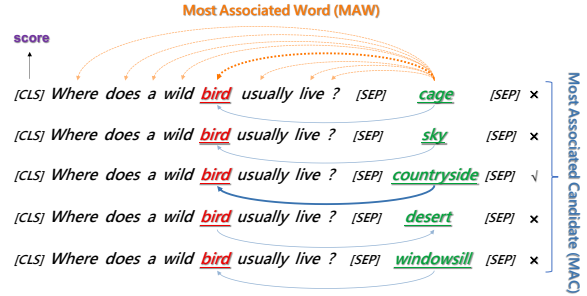


Figure 1: An illustration of two methods used to study structured commonsense knowledge in pre-trained transformer. Commonsense link is the link from the Target Concept (Answer Concept) to the Source Concept (Question Concept).

commonsense facts directly, Lin et al. (2019) and Klein and Nabi (2019) believe that structured commonsense knowledge is not captured well.

Pre-trained language models have achieved empirical success when fine-tuned on specific commonsense tasks such as COSMOS QA (Huang et al., 2019), SWAG (Zellers et al., 2018) and CommonsenseQA (Talmor et al., 2019). One possible reason of the high performance is that there exist superficial and statistical cues in the dataset, which enables models to answer questions without understanding the task (Niven and Kao, 2019; Yu et al., 2020). It remains an interesting research question whether pre-trained language models solve these tasks by making use of shallow clues, or real commonsense information.

We try to answer the research question by using the CommonsenseQA dataset, which asks a model to solve a multiple-choice problem. As shown in Figure 1, given a question and five candidate answers, a model should select one candidate answer as the output. The current state-of-the-art pre-trained language models solve the problem by representing the question jointly with each can-

didate answer (which is called a *sentence* thereafter), and use pre-trained language models as the main encoder. Scoring of each sentence is based on a sentence-level hidden vector, and the candidate answer that corresponds to the highest-scored sentence is taken as the output.

We investigate the presence of commonsense knowledge in the BERT representation of a sentence by examining the **commonsense link** from the **answer concept** to its related **question concept**, which is manually labelled in CommonsenseQA. Figure 2 shows one example, where the question concept is “bird”, and the correct answer is the answer concept connected through an ATLOCATION link in the CONCEPTNET knowledge graph. Such related concepts are *not* explicitly used in a BERT model for CommonsenseQA, and therefore its existence in the BERT representation reflects the use of commonsense knowledge. We call such knowledge *structured commonsense*, which can be a part of the commonsense knowledge that BERT makes use of, and a source of knowledge that we can explicitly measure. Two methods are used for measuring structured commonsense knowledge, including directly measuring the attention weights (Clark et al., 2019) and measuring attribution scores by considering gradients (Mudrakarta et al., 2018).

We consider two main types of experiments. In the first set, we consider the strengths of commonsense link to understand the existence of commonsense knowledge in the representation (Section 5). In the second set of experiments, we compare the strengths of commonsense link across the five sentence pairs given a question input, and thereby investigating the correlation between commonsense links with model predictions (Section 6). For each experiment above, we compare models without fine-tuning BERT and those with fine-tuning. While the former can serve as a probing task for understanding commonsense learned by pre-training, the latter can serve as a means for understanding whether fine-tuning allows a model to make better outputs by exploiting the use of commonsense knowledge.

Results suggest that BERT does have commonsense knowledge from pre-training, just as syntactic and word sense information. In addition, through fine-tuning, BERT relies more on commonsense information in making a prediction, which is demonstrated by stronger commonsense

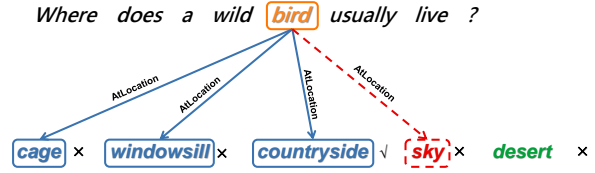


Figure 2: From CONCEPTNET to CommonsenseQA.

links in the representation, and a salient correlation between model predictions and commonsense link strengths, despite the fact that neither the answer concept nor the related question concept in a commonsense link is directly connected to the output layer. Interestingly, results also indicate that the stronger the structured commonsense knowledge is, the more accurate the model is. To our knowledge, we are the first to investigate how BERT solves the CommonsenseQA task, providing several evidences that commonsense knowledge is indeed made use of.

2 Related Work

Analysis of Pretraining. Peters et al. (2018) first point out that lower layers and higher layers in ELMo contain more syntactic and semantic information, respectively, based on the performance of downstream tasks for each layer. Tenney et al. (2019), Liu et al. (2019a) and Jawahar et al. (2019) use probing models on hidden states to analyze linguistic information within pre-trained language models. Goldberg (2019) assess BERT’s syntactic abilities by masking the verb, and compare the prediction probability of the original verb with incorrect verbs. Our work is similar to Clark et al. (2019) and Htut et al. (2019), with focuses on attention heads. The difference lies in that our primary goal is to investigate what information is learned on commonsense tasks. In contrast, their work focuses on analyzing syntactic and semantic information that exists in pre-trained language models.

Commonsense Reasoning. Commonsense reasoning is a challenging task in natural language processing. Traditional methods rely heavily on hand-crafted features (Rahman and Ng, 2012; Bailey et al., 2015) and external knowledge bases (Schüller, 2014). With recent advances in deep learning, pre-trained language models have been used as a powerful method for commonsense tasks. To solve Pronoun Disambiguation and

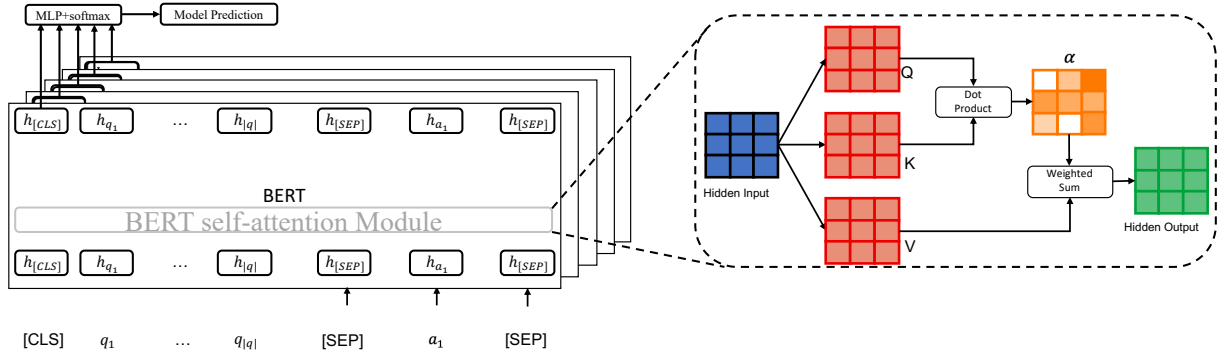


Figure 3: Architecture of BERT for CommonsenseQA. $q_1, \dots, q_{|q|}$ - question, a_1 - answer.

Winograd Schema Challenge (Levesque et al., 2012), Trinh and Le (2018) use a pre-trained language model to score candidate sentences. Klein and Nabi (2020) use a sentence-level loss to enhance commonsense knowledge in BERT. Mao et al. (2019) demonstrate that pre-trained language models fine-tuned on SWAG (Zellers et al., 2018) are able to provide commonsense grounding for story generation. For commonsense question answering, pre-trained language models with fine-tuning give the state-of-the-art performance (Zellers et al., 2018; Huang et al., 2019; Talmor et al., 2019).

This is also a line of work leveraging CONCEPTNET to enhance model’s commonsense reasoning ability. Lin et al. (2019) inject path information from question concepts to answer concepts to a model. Ye et al. (2019) use CONCEPTNET to construct pre-training dataset for BERT. Lv et al. (2019) extract evidence from CONCEPTNET and Wikipedia to build a relational graph for CommonsenseQA. We use CONCEPTNET for measuring commonsense knowledge in BERT, but do not aim to improve a model.

3 Task and Model

We introduce CommonsenseQA (Section 3.1), before showing how to apply BERT to CommonsenseQA (Section 3.2).

3.1 CommonsenseQA

CommonsenseQA (Talmor et al., 2019) is a multiple-choice question answering dataset constructed based on the CONCEPTNET knowledge graph (Speer et al., 2017), which is composed of a large set of triples of the relation pair ⟨source concept, relation, target concept⟩, such as ⟨BIRD, AT-LOCATION, COUNTRYSIDES⟩. As shown in Fig-

	CommonsenseQA		CommonsenseQA*	
	Train	Dev	Train	Dev
# Instances	9,741	1,221	9,741	1,147
# Relation	22	20	22	20

Table 1: Data statistics of CommonsenseQA and CommonsenseQA*

ure 2, given a source concept BIRD and the relation type ATLOCATION, there are three target concepts CAGE, WINDOWSILL and COUNTRYSIDE.

In the development of the CommonsenseQA dataset, crowd-workers are requested to generate question and candidate answers based on the source concept and three target concepts, respectively. Following Talmor et al. (2019), we call the source concept in the question as *question concept*, and the target concept in the answer as *answer concept*. To make the task more difficult, two additional incorrect answers are added. We define *commonsense link* as the link from the answer concept to the question concept. In order to analyze implicit structured commonsense knowledge, which is based on the link from the answer concept to the question concept, we filter out questions which do not contain the question concept in its CONCEPTNET form (e.g. paraphrase). The detailed statistics of the resulting dataset CommonsenseQA* are summarized in Table 1.

3.2 BERT for CommonsenseQA

We adopt the method of Talmor et al. (2019) using BERT (Devlin et al., 2019) on CommonsenseQA. The structure is shown in Figure 3. In particular, given a question q and 5 candidate answers a_1, \dots, a_5 , we concatenate the question with each answer to obtain 5 concatenated sequences (i.e. sentences) s_1, \dots, s_5 , respectively. In each sentence, we use a special symbol $[CLS]$ in the be-

ginning, a symbol *[SEP]* between the question and the candidate answer, a symbol *[SEP]* in the end.

BERT consists of L stacked Transformer layer (Vaswani et al., 2017) to encode each sentence. The last layer hidden state of the *[CLS]* token is used for linear classification with softmax, and the candidate among s_1, \dots, s_5 with the highest score is chosen as the prediction. More details of our implementation are shown in Appendix A.

4 Analysis Methods

We analyze commonsense links using the attention weight and the corresponding attribution score.

4.1 Attention Weights

Given a sentence, attention weights in Transformer can be viewed as the relative importance weight between each token and the other tokens when producing the next layer representation (Kovaleva et al., 2019; Vashishth et al., 2020). In particular, given a sequence of input vectors $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_{|H|}]$, its self-attention representation uses each vector as a query to retrieve all context, yielding a matrix of attention weights $\alpha \in \mathbb{R}^{|H| \times |H|}$.

The attention weight α is computed from the scaled dot-product of the query vector of $\mathbf{Q} = \mathbf{W}^Q \mathbf{H}$ and the key vector of $\mathbf{K} = \mathbf{W}^K \mathbf{H}$, followed by softmax normalization

$$\alpha = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right), \quad (1)$$

where d_k is the dimension size of the key vector \mathbf{K} . $\alpha_{i,j}$ represents the attention from \mathbf{h}_i to \mathbf{h}_j . For multi-head attention, the queries, keys, and values are linearly projected T times, where T is the number of heads. The attention operation of each head is performed in parallel, with the results being concatenated. In BERT-base, there are 12 layers in total. We use $\alpha^{m,n}$ to denote the n -th attention head in the m -th layer. $\alpha^{m,n}$ are used as a measure of commonsense link strengths.

4.2 Attribution Scores

Kobayashi et al. (2020) point out that analyzing only attention weights can be insufficient to investigate the behavior of attention head, because attention weights disregard the values of the hidden vector \mathbf{H} . As a supplement of attention weights, gradient-based feature attribution methods have been studied to interpret the contribution of each

input feature to the model prediction in back-propagation (Baehrens et al., 2010; Mudrakarta et al., 2018; Hao et al., 2020). Analysis of both attention weights and the corresponding attribution scores allows us to more comprehensively understand the commonsense link in BERT.

We employ an attribution technique called *Integrated Gradients* (Sundararajan et al., 2017) to analyze commonsense links in BERT. Intuitively, *integrated gradients* simulate the process of pruning the specific attention head (from the original attention weight α to a zero vector α'), and calculate the integrated gradients in back-propagation. The attribution score directly reflects how much changing attention weights will change model’s outputs. A higher attribution score represents more importance of individual attention weight. Suppose that $F(x)$ represents the BERT model output for CommonsenseQA given an input x . The attribution of attention head t can be computed by:

$$\text{Atr}(\alpha^t) = (\alpha^t - \alpha'^t) \otimes \int_{x=0}^1 \frac{\partial F(\alpha' + x(\alpha - \alpha'))}{\partial \alpha^t} dx \quad (2)$$

where \otimes is the element-wise multiplication, $\alpha = [\alpha^1, \dots, \alpha^T]$. In this equation, $F(\alpha' + x(\alpha - \alpha'))$ is closer to $F(\alpha')$ when x is closer to 0, and closer to α when x is closer to 1. Therefore, $\int_{x=0}^1 \frac{\partial F(\alpha' + x(\alpha - \alpha'))}{\partial \alpha^t} dx$ gives the amortized gradient with all different x . $\text{Atr}(\alpha^t) \in \mathbb{R}^{n \times n}$ denotes the attribution score which corresponding to the attention weight α^t . $\text{Atr}(\alpha_{i,j}^t)$ is represented for the interaction from token \mathbf{h}_i to \mathbf{h}_j . We set the uninformative baseline α' as zero vector. Following Sundararajan et al. (2017), we approximate $\text{Atr}(\alpha^t)$ via a gradient summation function,

$$\text{Atr}(\alpha^t) ::= (\alpha^t - \alpha'^t) \odot \sum_{i=1}^s \frac{\partial F(\alpha' + i/s(\alpha - \alpha'))}{\partial \alpha^t} \times \frac{1}{s}, \quad (3)$$

where s is the number of approximation steps for computing integrated gradients.

4.3 Experimental Settings

Sections 5 and 6 report results in one random execution. We additionally tried five runs for each experiments, and found that the result variation is small. We list detailed results as well as their standard deviation in Appendix C.

5 Does BERT Contain Structured Commonsense Knowledge?

We first conduct a set of experiments to investigate the commonsense link weight, which can reflect

whether commonsense knowledge is captured by the BERT representation of the sentence. Intuitively, if the link weight from the answer concept to the question concept is higher than those from the answer concept to other question words, then we know that the commonsense knowledge in CONCEPTNET is captured by the representation empirically. It is worth noting that rather than the question concept, the representation of the $[CLS]$ token is directly connected to the output layer for candidate scoring. Hence there is no direct supervision signal from the output layer to the link weight from answer concept tokens to the question concept tokens during both pre-training and fine-tuning.

5.1 Probing Task

We evaluate link weights by calculating the **most associated word** (MAW), namely the question concept word that receives the maximum link weight from the answer concept among all question words. We calculate MAW for each individual attention head in each layer.

Formally, assume that the hidden state of the whole question, question concept and answer concept are $[\mathbf{h}_1, \dots, \mathbf{h}_{|q|}]$, $[\mathbf{h}_{b_s}, \dots, \mathbf{h}_{e_s}]$ and $[\mathbf{h}_{b_t}, \dots, \mathbf{h}_{e_t}]$, respectively. If the answer concept is composed of multiple tokens, we consider the link weight from the answer concept to the i th token \mathbf{h}_i ($i \in [1, |q|]$), as the mean of the link weights over all answer tokens $\alpha_i = \frac{1}{e_t - b_t} \sum_{j=b_t}^{e_t} \alpha_{j,i}$. For the n -th attention head in the m -th layer, if the question concept receives maximum link weight from the answer concept ($\mu^{m,n} = \arg \max_i \alpha_i^{m,n}$, $\mu^{m,n} \in [b_s, e_s]$), we consider this attention head gives correct prediction for MAW, and tracks the commonsense dependency precisely in this sentence.

We take two different measures of MAW accuracies, measuring the average accuracy among all attention head, and the accuracy of the most-accurate head, respectively. Previous work probing syntactic information from attention head takes the second method (Clark et al., 2019; Htut et al., 2019). We additionally take the first method in order to comprehensively evaluate the degree of commonsense knowledge in addition to their existence. Formally, the MAW accuracy of a specific head is the percentage of MAW words that are consistent with the relevant question concept from CONCEPTNET.

Relation Type	Max	Avg	Layer-Head
Random	10.53	10.53	-
OVERALL(BERT)	46.82	12.38	9-0
OVERALL(BERT-FT)	49.22	17.35	8-7
ATLOCATION	55.85	18.42	8-7
CAUSES	55.93	18.91	8-7
CAPABLEOF	47.88	14.71	8-1
ANTONYM	52.53	10.97	4-3
HASPREREQUISITE	54.15	18.93	9-8
HASSUBEVENT	55.29	18.74	9-0
DESIRE	40.00	7.92	8-1
CAUSESDESIRE	48.89	14.28	4-0
PARTOF	59.09	18.56	9-0
HASPROPERTY	54.00	15.12	9-1
MOTIVATEDBYGOAL	75.56	24.31	9-7
HASA	68.89	22.10	8-1
RELATEDTO	62.22	18.44	9-0

Table 2: The average and maximum MAW accuracies of BERT-FT for different commonsense relations. We exclude the relation types with frequencies of occurrence less than 9. Layer-Head represents the best performing attention head for each relation.

The average MAW accuracy is measured by: $acc^{avg} = \frac{\sum_{m=0}^{11} \sum_{n=0}^{11} \sum_{d=1}^D \mathbb{1}(\mu^{m,n} \in [b_s, e_s])}{12 \times 12 \times D}$. The maximum MAW accuracy is measured by: $acc^{max} = \max_{m=0}^{11} \max_{n=0}^{11} \frac{\sum_{d=1}^D \mathbb{1}(\mu^{m,n} \in [b_s, e_s])}{D}$, where D represents the number of instances for evaluation.

In theory, if link weights for each attention head are randomly distributed, the average MAW accuracy and the maximum MAW accuracy should be both $acc^{baseline} = \frac{\sum_{d=1}^D \frac{e_s - b_s}{|q|}}{D}$, which reflects the fact that the representation does not contain explicitly correlation between the answer concept and its related question concept. In contrast, MAW accuracies significantly better than this baseline indicates that commonsense knowledge is contained in the representation.

5.2 Results

The results for the original normalization BERT (BERT) and a BERT model fine-tuned on CommonsenseQA (BERT-FT) are shown in table 2. First, looking at the original non-fine-tuned BERT, the maximum MAW accuracy of each layer significantly outperforms the random baseline, which shows that commonsense knowledge is indeed captured by BERT. In addition, the average MAW of BERT significantly outperforms the random baseline (p -value < 0.01), which indicates that the relevant question concept plays a highly important role in BERT encoding without fine-tuning. Second, BERT-FT outperforms BERT in terms of both

Head	Attention				Attribution	
	BERT-FT		BERT-Probing		BERT-FT	
	MAC	MAS	MAC	MAS	MAC	MAS
0	49.00	18.92	29.21	4.01	51.61	23.54
1	49.17	19.62	20.75	10.99	27.46	24.85
2	32.00	56.23	16.04	43.85	49.17	33.83
3	41.33	16.74	32.17	9.68	22.93	47.08
4	49.96	24.32	33.91	6.28	31.04	44.29
5	45.42	13.25	34.87	4.62	34.26	20.14
6	48.39	13.33	25.72	7.41	33.83	22.67
7	54.14	13.39	28.07	3.66	25.98	49.61
8	39.67	16.74	28.86	9.50	36.97	22.84
9	38.71	13.95	24.50	18.66	52.14	21.01
10	49.17	8.89	36.88	7.15	36.79	21.19
11	53.53	11.07	30.08	3.31	25.81	26.94
Avg	45.87	18.85	28.42	10.76	35.67	29.83

Table 3: Comparison between $\text{MAC}^{\text{overlap}}$ and $\text{MAS}^{\text{overlap}}$ in the top layer.

average MAW accuracy and maximum MAW accuracy, which shows that structured commonsense knowledge is enhanced by supervised training on commonsense tasks.

We further explore the best performing attentions head for each commonsense relation type in Table 2, finding that certain attention heads capture specific commonsense relations. There is no single attention head that does well for all relation types, which is similar to the previously finding for syntactic heads (Raganato and Tiedemann, 2018; Clark et al., 2019).

Finally, structured commonsense knowledge that we examine does not show a salient trend of distribution among different layers. The maximum MAW exists in the 9-th and 8-th layers before and after fine-tuning, respectively. However, we will see in the next section that the model relies more commonsense knowledge on higher layers for making prediction.

6 How Does BERT Use Commonsense Knowledge for Commonsense Task?

We further conduct a set of experiments to draw the correlation between commonsense links and model prediction. The goal is to investigate whether the link weights from different candidate answer concepts to the question concept influence the model decision among those candidates. In particular, we compare the link weights across the five candidates for the same question, and find out the candidate that is the most associated with the relevant question concept. This candidate is called the **most associated candidate** (MAC). Correlations are drawn between MACs and the model pre-

diction for each question. Intuitively, if the MACs are correlated with the model predictions, then we have evidence that the model makes use of commonsense knowledge in making prediction. Experiments are conducted to evaluate the contribution of MAC to the model decision, and the correlation between the reliance on MACs and the output accuracies. Both attention weights and the corresponding attribution scores are used to measure links, because now we are considering model prediction, for which gradients play a role. For all experiments, the trend of attribute scores is consistent with that measured using attention weights. We mainly report results measured as attention weights, and put the corresponding attribution scores in Appendix B.

6.1 Probing Tasks

Formally, given a question q and 5 candidate answers a_1, \dots, a_5 , we make comparisons across five candidate sentences s_1, \dots, s_5 . In each candidate sentence, we calculate the link weight from the answer concept to the question concept according to CONCEPTNET. We denote the hidden states of the question concept and the answer concept as $[\mathbf{h}_{b_s}, \dots, \mathbf{h}_{e_s}]$ and $[\mathbf{h}_{b_t}, \dots, \mathbf{h}_{e_t}]$, respectively. The link weight of the answer-question-concept pair (α_{a2q}) is the averaged link weights from each answer concept token to each question concept token

$$\alpha_{a2q} = \frac{\sum_{i=b_s}^{e_s} \sum_{j=b_t}^{e_t} \alpha_{j,i}}{(e_s - b_s)(e_t - b_t)}$$

Among the five candidates in each instance, we take the one with the highest α_{a2q} as the MAC, denoted as $p^{\text{MAC}} \in [1, 5]$.

We further define **most associated sentence** (MAS) by measuring the link weight from the answer concept to the $[CLS]$ token. The reason is that gradients are back-propagated from the $[CLS]$ token rather than the question concept or the answer concept. By comparing MAC and MAS, we can have useful information on whether MAC is an influencing factor for the model decision.

For analysis, we measure the correlation between MAC ($p^{\text{MAC}} \in [1, 5]$), the model prediction ($p^{\text{model}} \in [1, 5]$) and the gold-standard answer ($p^{\text{golden}} \in [1, 5]$) by using two metrics, including the overlapping rate between MACs and model predictions, and the accuracy of MACs.

The **overlapping rate of MACs** for each head

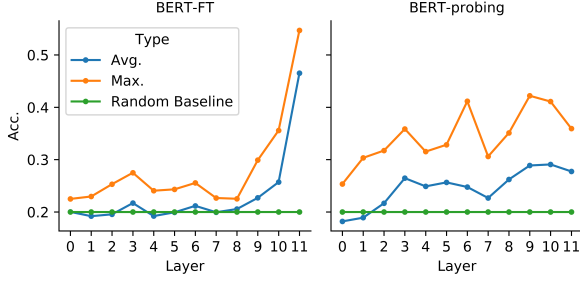


Figure 4: The performance of $\text{MAC}^{\text{overlap}}$ across different layers.

#H	#Ins	Model Acc.	#H	#Ins	Model Acc.
0	158	20.89	7	69	78.26
1	135	28.15	8	63	82.54
2	119	52.10	9	57	92.98
3	132	53.79	10	47	89.36
4	93	62.37	11	44	97.73
5	106	66.04	12	36	100.00
6	88	68.18	-	-	-

Table 4: The relationship between the MAC head count and the model prediction accuracy. #H denotes how many heads yield the correct MAC prediction.

is defined as:

$$\text{MAC}^{\text{overlap}} = \frac{\sum_d^D \mathbb{1}(p_d^{\text{MAC}} = p_d^{\text{model}})}{D}$$

The **accuracy of MACs** for each head is defined as the percentage of MACs that equals the gold-standard answer p^{golden} :

$$\text{MAC}^{\text{acc}} = \frac{\sum_d^D \mathbb{1}(p_d^{\text{MAC}} = p_d^{\text{golden}})}{D}$$

6.2 The Importance of Commonsense Link

We measure the MAC performance of BERT-FT, and a BERT model that is fine-tuned for the output layer only (BERT-probing). The latter is a linear probing model. Intuitively, if the linear classifier can predict the commonsense task, then the original non-fine-tuned BERT likely encodes the rich commonsense knowledge.

Table 3 shows the overlapping rates of MACs and MASS according to the 12 attention heads in the top Transformer layer. First, for both models, the overlapping rates of MACs are saliently larger than that with MASS. This suggests that the link weight from the answer concept to the question concept is more closely-related to the model prediction as compared to the link weight from the answer concept to the $[CLS]$ token, despite that model’s output scores are calculated on the

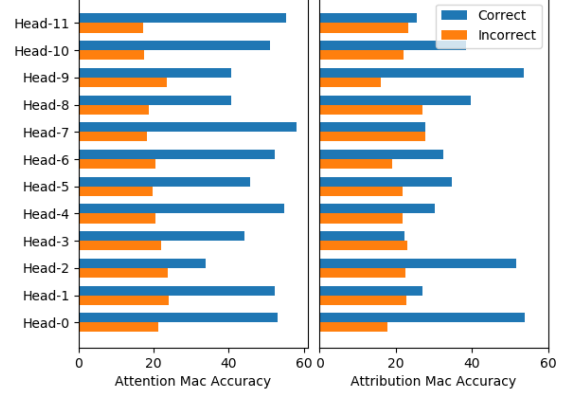


Figure 5: MAC^{acc} of each attention head in the top layer with correct and incorrect model predictions.

$[CLS]$ token. This result is consistent with the results in Section 5.2 in showing that commonsense knowledge does play a certain role in the model. Second, when fine-tuned with training data, the model gives an even stronger correlation between MAC and the model prediction. This suggests that the model can *learn* to make use of commonsense information for making prediction, which partly shows how a BERT model solves CommonsenseQA.

6.3 The Correlation between Commonsense Link and Model Prediction

To further investigate the contribution of commonsense knowledge for model prediction, Figure 4 shows the overlapping rate between MAC and model prediction at each Transformer layer. Both the maximum and the average overlapping rates across the 12 layers are shown. In addition, the random overlapping rate of 20% is drawn as a reference. It can be seen from the figure that the maximum overlapping rate of BERT-probing is significantly over the random baseline, which shows that the model prediction is associated with relevant structured commonsense information. In addition, after fine-tuning, the BERT-FT model shows a tendency of weakened maximum MAC overlapping rate on lower Transformer layers and much strengthened MAC overlapping rate on higher Transformer layers, and in particular the top layer. This suggests that fine-tuned model relies more on the commonsense structure in the top layer for making prediction.

Table 4 shows the correlation between MAC accuracies and model prediction accuracies. Each

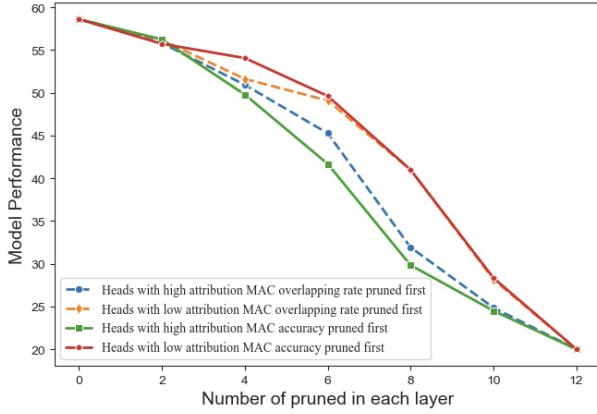


Figure 6: Model performance on the CommonsenseQA development set when different heads are pruned.

row shows a different number of heads in the top layer for which MAC corresponds to the correct answer candidate, together with the number of test instances for such cases, and the model prediction accuracy on the instances. There is an obvious trend where increased MAC accuracies correspond to increase model prediction accuracies, despite that commonsense links are not intentionally optimized in model training, which shows that making use of structured commonsense knowledge in the sentence representation leads to better model prediction.

Figure 5 shows the MAC accuracies of each attention head in the top layer for the test instances with correct and incorrect model predictions, respectively. The MAC accuracies of correctly predicted instances are larger than those of incorrectly predicted instances by a large margin. The finding is consistent with Table 4, which shows that structured commonsense knowledge is useful for making the correct decision.

We further evaluate the model performance after pruning specific heads. We sort all the attention heads in each layer according to their attribution MAC performance, and then prune these heads in order. Following Michel et al. (2019), we replace the pruned head with zero vectors. Figure 6 shows the model performance on the development set. As the number of pruned heads increases, the model performance decreases, which conforms to intuition. Furthermore, the model performance drops much more rapidly when the attention heads with higher MAC performance are pruned first, which demonstrates capturing commonsense link is effective to guide model prediction.

L	BERT-FT			BERT-probing		
	MAC ^{overlap}		Model	MAC ^{overlap}		Model
	Max	Avg		Max	Avg	
11	54.14	45.87	58.59	36.88	28.42	39.23
10	46.56	26.65	56.50	37.66	27.11	35.48
9	37.40	27.86	53.36	39.84	28.50	33.74
8	34.61	24.01	51.53	30.08	24.76	32.52
7	31.82	21.39	49.35	25.81	21.53	33.57
6	31.73	24.40	48.74	37.05	24.04	32.96
5	31.56	23.64	45.95	31.21	24.02	32.00
4	34.44	25.01	44.99	33.39	24.03	32.43
3	44.73	34.13	40.28	41.06	27.67	33.83
2	44.20	32.48	37.58	25.81	21.02	21.88
1	23.71	19.47	26.68	23.63	20.74	20.40
0	23.45	19.50	23.02	20.58	18.81	19.27

Table 5: Performance of MAC^{overlap} across different layers. L-*n* represents adding the output classifier on the hidden state of layer-*n*. Our BERT-FT model (layer-11) gives 58.15% accuracies, which is slightly higher than the reported results of 55.57% on Lin et al. (2019). It achieves 58.59% on our dataset CommonsenseQA*.

6.4 The Contribution of Different Layers

We further investigate two detailed questions on the commonsense knowledge usage. First, which layer does BERT rely on the most for making its decision. Second, does the commonsense knowledge that BERT uses come from pre-training or fine-tuning. We compare 12 model variations by connecting the output layer on each of the Transformer layer, respectively. Table 5 shows the model accuracies and the MAC overlapping rates. First, BERT-probing gives the best performance when prediction is made on the top layer, and the accuracy generally decreases as the layer moves to the bottom. This indicates that relevant commonsense knowledge is more heavily distributed towards higher layers during pre-training. Our experimental settings here are the same as the probing task for syntactic information by Liu et al. (2019a), who find that syntactic information is distributed more heavily towards lower layers.

With fine-tuning, we observe stronger improvements of both model accuracies and MAC overlaps on higher layers when comparing BERT-FT and BERT-probing. This demonstrates that commonsense knowledge on higher layers is more useful to the CommonsenseQA task. Interestingly, comparing layer 11 and layer 10, the model accuracy after fine-tuning is similar, but the MAC overlap of layer 11 is significantly larger. This shows that the structured commonsense knowledge that we probe attributes only partly to the overall useful knowl-

edge for CommonsenseQA.

7 Conclusion

We conducted qualitative and quantitative analysis to investigate how BERT solves the CommonsenseQA task, aiming to gain evidence on the source of information involved in the disambiguation process. Empirical results demonstrated that BERT encodes structured commonsense knowledge, and is able to leverage such knowledge to a certain degree on downstream commonsense tasks. Our analysis has further revealed that with fine-tuning, BERT is able to leverage rich commonsense knowledge on higher layers. These suggest that BERT does not solely on superficial patterns for CommonsenseQA. To our knowledge, we are the first to show evidences on the mechanism for BERT when conducting CommonsenseQA, which can inspire further work exploiting the underlying mechanisms.

References

- David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. 2010. How to explain individual classification decisions. *J. Mach. Learn. Res.*, 11:18031831.
- Daniel Bailey, Amelia Harrison, Yuliya Lierler, Vladimir Lifschitz, and Julian Michael. 2015. The winograd schema challenge and reasoning about correlation. In *AAAI Spring Symposia*.
- Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli elikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. [What does BERT look at? an analysis of BERT’s attention](#). In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.
- Leyang Cui, Yu Wu, Shujie Liu, Yue Zhang, and Ming Zhou. 2020. Mutual: A dataset for multi-turn dialogue reasoning. In *Proceedings of the 58th Conference of the Association for Computational Linguistics*. Association for Computational Linguistics.
- Joe Davison, Joshua Feldman, and Alexander Rush. 2019. [Commonsense knowledge mining from pre-trained models](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1173–1178, Hong Kong, China. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Yoav Goldberg. 2019. [Assessing bert’s syntactic abilities](#).
- Yaru Hao, Li Dong, Furu Wei, and Ke Xu. 2020. [Self-attention attribution: Interpreting information interactions inside transformer](#).
- Phu Mon Htut, Jason Phang, Shikha Bordia, and Samuel R. Bowman. 2019. [Do attention heads in bert track syntactic dependencies?](#)
- Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. [Cosmos QA: Machine reading comprehension with contextual commonsense reasoning](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2391–2401, Hong Kong, China. Association for Computational Linguistics.
- Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. [What does BERT learn about the structure of language?](#) In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.
- Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel Weld. 2019. [BERT for coreference resolution: Baselines and analysis](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5803–5808, Hong Kong, China. Association for Computational Linguistics.
- Tassilo Klein and Moin Nabi. 2019. [Attention is \(not\) all you need for commonsense reasoning](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4831–4836, Florence, Italy. Association for Computational Linguistics.
- Tassilo Klein and Moin Nabi. 2020. [Contrastive self-supervised learning for commonsense reasoning](#).
- Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2020. Attention module is not only a weight: Analyzing transformers with vector norms. *ArXiv*, abs/2004.10102.

- Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. [Revealing the dark secrets of BERT](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4365–4374, Hong Kong, China. Association for Computational Linguistics.
- Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *13th International Conference on the Principles of Knowledge Representation and Reasoning, KR 2012*, pages 552–561.
- Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. [KagNet: Knowledge-aware graph networks for commonsense reasoning](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2829–2839, Hong Kong, China. Association for Computational Linguistics.
- Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019a. [Linguistic knowledge and transferability of contextual 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)*, pages 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.
- Yang Liu and Mirella Lapata. 2019. [Text summarization with pretrained encoders](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. 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. 2019b. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. 2019. [Graph-based reasoning over heterogeneous external knowledge for commonsense question answering](#).
- Huanru Henry Mao, Bodhisattwa Prasad Majumder, Julian McAuley, and Garrison Cottrell. 2019. [Improving neural story generation by targeted commonsense grounding](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5988–5993, Hong Kong, China. Association for Computational Linguistics.
- Paul Michel, Omer Levy, and Graham Neubig. 2019. [Are sixteen heads really better than one?](#) In H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 14014–14024. Curran Associates, Inc.
- Pramod Kaushik Mudrakarta, Ankur Taly, Mukund Sundararajan, and Kedar Dhamdhere. 2018. [Did the model understand the question?](#) In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1896–1906, Melbourne, Australia. Association for Computational Linguistics.
- Timothy Niven and Hung-Yu Kao. 2019. [Probing neural network comprehension of natural language arguments](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4658–4664, Florence, Italy. Association for Computational Linguistics.
- Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. [Deep contextualized word representations](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.
- Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. [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)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.
- Alessandro Raganato and Jörg Tiedemann. 2018. [An analysis of encoder representations in transformer-based machine translation](#). In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 287–297, Brussels, Belgium. Association for Computational Linguistics.
- Altaf Rahman and Vincent Ng. 2012. [Resolving complex cases of definite pronouns: The Winograd schema challenge](#). In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 777–789, Jeju Island, Korea. Association for Computational Linguistics.
- Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. [Explain yourself! leveraging language models for commonsense](#)

- [reasoning](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy. Association for Computational Linguistics.
- Peter Schüller. 2014. Tackling winograd schemas by formalizing relevance theory in knowledge graphs. In *KR*.
- Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI17, page 44444451. AAAI Press.
- Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML17, page 33193328. JMLR.org.
- Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. [CommonsenseQA: A question answering challenge targeting commonsense knowledge](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.
- 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 *International Conference on Learning Representations*.
- Trieu H. Trinh and Quoc V. Le. 2018. A simple method for commonsense reasoning. *ArXiv*, abs/1806.02847.
- Shikhar Vashishth, Shyam Upadhyay, Gaurav Singh Tomar, and Manaal Faruqui. 2020. [Attention inter-pretability across {nlp} tasks](#).
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.
- Zhi-Xiu Ye, Qian Chen, Wen Wang, and Zhen-Hua Ling. 2019. [Align, mask and select: A simple method for incorporating commonsense knowledge into language representation models](#).
- Weihaoyu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2020. Reclor: A reading comprehension dataset requiring logical reasoning. In *International Conference on Learning Representations*.
- Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. [SWAG: A large-scale adversarial dataset for grounded commonsense inference](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium. Association for Computational Linguistics.
- Junru Zhou and Hai Zhao. 2019. [Head-driven phrase structure grammar parsing on Penn treebank](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2396–2408, Florence, Italy. Association for Computational Linguistics.

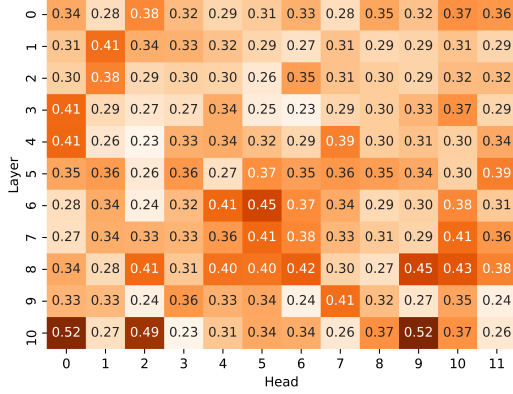


Figure 7: $MAC^{overlap}$ in a given head by attribution score.

#H	#Ins	Model Acc.	#H	#Ins	Model Acc.
0	89	10.11	5	171	72.51
1	114	22.81	6	119	81.51
2	148	51.35	7	85	82.35
3	156	56.41	8	43	74.72
4	207	66.67	9	13	84.62

Table 6: The relationship between head attribution MAC prediction and model prediction accuracy. #H denotes how many heads yield correct MAC prediction. We exclude the #H, if the frequency of occurrence is less than 3.

A Implementation Details

We adopt the huggingface BERT-base implementation for multiple-choice on CommonsenseQA. We conduct fine-tuning experiments using GeForce GTX 2080Ti. For BERT-FT and BERT-probing, we optimize the parameters with grid search: training epochs 3, learning rate $\{5e-4, 1e-5, 3e-5, 5e-5, 5e-6\}$, training batch size $\{8, 16, 32\}$, gradient accumulation steps $\{2, 4, 8\}$. To demonstrate the robustness of our analysis method, We repeat the experiment 5 times with the same hyperparameter, and report the experiment results based on one random model.

We calculate the attribution score to interpret BERT using captum, which is an extensible library for model interpret ability built on Pytorch.

B Detailed performance of MAW

We report the average and maximum MAW accuracy across different layers in Table 7. The average MAW of 6 layers significantly outperforms the random baseline, which indicates that the relevant question concept plays a highly important role in BERT encoding without fine-tuning. BERT-FT

	BERT-FT			BERT			
L	Max	Avg	t	Max	Avg	t	Rand
11	34.11	19.78	✓	32.44	14.47	✓	10.53
10	39.09	26.10	✓	40.84	22.22	✓	10.53
9	46.31	25.59	✓	46.82	18.68	✓	10.53
8	49.22	13.71	✓	44.48	10.15	-	10.53
7	32.76	8.88	-	28.00	5.61	-	10.53
6	40.68	12.16	✓	41.99	9.01	-	10.53
5	33.30	14.41	✓	13.22	4.34	-	10.53
4	38.89	19.09	✓	24.10	10.46	-	10.53
3	37.30	14.59	✓	24.74	7.43	-	10.53
2	35.08	17.71	✓	31.96	12.14	✓	10.53
1	29.01	15.08	✓	27.64	11.09	✓	10.53
0	45.55	23.05	✓	46.16	22.95	✓	10.53

Table 7: The average and maximum MAW accuracy across different layers. ✓ indicates p -value < 0.01 .

outperforms BERT in terms of both average MAW accuracy and maximum MAW accuracy, which shows that structured commonsense knowledge is enhanced by supervised training on commonsense tasks.

C Performance of MAC

Figure 7 shows the attribution $MAC^{overlap}$ of BERT-FT. Table 6 shows the correlation between attribution MAC accuracies and model prediction accuracies. The trend is consistent with attention weights, as reported in Figure 4 and Table 4, respectively.

Table 8 shows the MAC and MAS performance for each attention head across five turns. Noted that the standard derivations are only 1.17% and 1.76% for MAC and MAS, respectively, which demonstrates the robustness of our methods.

	MAC						MAS					
H	M1	M2	M3	M4	M5	mean±std	M1	M2	M3	M4	M5	mean±std
0-0	20.05	19.88	18.13	17.96	19.62	19.13±1.00	19.70	16.56	19.09	20.23	20.23	19.16±1.53
0-1	18.83	19.44	20.40	20.40	20.31	19.88±0.71	18.05	19.44	19.01	17.79	17.79	18.41±0.76
0-2	20.66	22.76	22.32	21.88	22.67	22.06±0.85	19.88	17.70	19.01	20.05	20.05	19.34±1.01
0-3	20.66	20.75	20.14	19.09	19.01	19.93±0.84	19.27	19.53	18.74	19.01	19.01	19.11±0.30
0-4	18.22	18.40	19.01	18.57	18.22	18.48±0.33	19.53	18.92	19.53	18.48	18.48	18.99±0.52
0-5	20.23	19.70	19.79	20.31	20.31	20.07±0.30	17.26	17.96	17.35	16.04	16.04	16.93±0.86
0-6	20.84	20.92	20.49	22.06	20.49	20.96±0.65	19.27	20.49	19.44	17.96	17.96	19.02±1.08
0-7	20.23	19.62	20.75	20.40	21.10	20.42±0.56	18.13	17.44	18.66	18.66	18.66	18.31±0.54
0-8	20.49	18.48	20.05	19.18	20.75	19.79±0.94	18.13	17.18	16.65	15.78	15.78	16.70±1.00
0-9	21.88	23.71	21.88	24.06	23.54	23.02±1.05	17.79	17.52	17.87	16.65	16.65	17.30±0.60
0-10	18.48	19.18	19.35	19.44	19.97	19.29±0.54	20.23	19.18	16.48	19.44	19.44	18.95±1.44
0-11	20.66	20.31	21.53	20.84	21.01	20.87±0.45	19.09	18.66	18.92	18.22	18.22	18.62±0.40
1-0	21.45	20.66	19.79	19.88	20.75	20.51±0.68	19.18	19.27	19.35	17.09	17.09	18.40±1.20
1-1	19.01	19.97	20.49	21.62	19.88	20.19±0.96	20.05	19.35	18.48	20.23	20.23	19.67±0.75
1-2	19.97	19.27	20.23	19.62	21.62	20.14±0.90	22.14	22.41	22.58	20.66	20.66	21.69±0.95
1-3	16.83	16.22	18.13	18.31	17.35	17.37±0.88	21.88	21.27	22.14	20.23	20.23	21.15±0.90
1-4	22.41	22.14	20.92	21.19	22.14	21.76±0.66	20.40	19.70	18.83	18.13	18.13	19.04±1.00
1-5	19.88	19.01	17.26	17.79	17.61	18.31±1.10	21.27	21.10	21.45	21.45	21.45	21.34±0.16
1-6	21.45	22.67	20.40	18.83	22.67	21.20±1.63	16.65	17.35	15.61	18.31	18.31	17.24±1.15
1-7	18.13	19.88	20.40	19.97	18.92	19.46±0.92	22.23	20.31	21.71	21.97	21.97	21.64±0.76
1-8	15.00	15.43	14.21	14.65	14.47	14.75±0.48	25.54	23.45	24.50	25.02	25.02	24.71±0.79
1-9	15.69	14.56	15.26	15.26	15.78	15.31±0.48	26.85	25.20	27.03	25.81	25.81	26.14±0.78
1-10	20.14	19.70	18.92	21.10	19.79	19.93±0.79	20.84	19.53	19.97	20.66	20.66	20.33±0.56
1-11	22.23	22.76	23.28	24.50	23.10	23.17±0.84	15.00	12.12	12.82	13.25	13.25	13.29±1.06
2-0	21.71	21.01	21.71	21.10	21.01	21.31±0.37	19.62	20.58	19.53	19.09	19.09	19.58±0.61
2-1	19.09	17.44	18.05	18.57	19.09	18.45±0.71	16.65	14.39	15.26	15.69	15.69	15.54±0.82
2-2	19.88	20.23	19.27	19.62	18.31	19.46±0.73	15.17	16.65	18.66	16.39	16.39	16.65±1.26
2-3	16.13	14.04	15.69	13.69	15.69	15.05±1.10	26.07	27.03	28.07	27.11	27.11	27.08±0.71
2-4	21.01	19.53	20.23	22.14	21.53	20.89±1.04	17.09	18.66	19.27	16.30	16.30	17.52±1.37
2-5	22.41	22.23	21.71	22.49	21.88	22.14±0.34	19.01	19.35	20.05	19.18	19.18	19.35±0.41
2-6	22.49	22.93	23.28	23.37	23.10	23.03±0.35	19.01	19.70	20.14	18.74	18.74	19.27±0.63
2-7	18.40	16.65	16.48	17.87	17.52	17.38±0.81	20.05	19.97	20.40	18.57	18.57	19.51±0.87
2-8	16.04	15.78	15.61	16.13	16.39	15.99±0.31	20.84	21.27	22.06	19.27	19.27	20.54±1.24
2-9	20.92	20.66	23.37	22.23	20.40	21.52±1.25	20.66	21.36	20.66	19.79	19.79	20.45±0.67
2-10	25.89	26.16	25.37	26.94	25.54	25.98±0.62	15.95	16.39	17.26	17.09	17.09	16.76±0.56
2-11	19.01	18.22	18.92	17.44	17.70	18.26±0.70	20.31	19.97	20.92	20.92	20.92	20.61±0.45
3-0	26.77	27.20	25.37	26.07	27.38	26.56±0.83	21.71	24.59	19.97	23.28	23.28	22.56±1.77
3-1	22.14	19.79	22.49	23.71	24.24	22.48±1.73	15.61	17.79	14.91	14.56	14.56	15.48±1.36
3-2	23.63	24.59	24.32	23.54	24.85	24.18±0.58	16.83	18.66	16.48	19.79	19.79	18.31±1.59
3-3	19.79	18.40	19.88	20.75	19.44	19.65±0.85	21.01	24.93	21.01	24.32	24.32	23.12±1.94
3-4	21.88	20.49	22.23	22.14	25.98	22.55±2.05	22.49	22.14	19.97	26.24	26.24	23.42±2.75
3-5	20.23	18.05	20.58	20.05	21.10	20.00±1.16	19.27	17.87	17.79	18.83	18.83	18.52±0.65
3-6	25.54	25.98	26.16	27.20	26.85	26.35±0.67	15.52	15.43	14.39	17.35	17.35	16.01±1.30
3-7	21.27	18.57	17.87	18.13	21.45	19.46±1.75	24.24	26.07	24.50	20.58	20.58	23.19±2.49
3-8	19.70	21.45	18.48	18.66	21.27	19.91±1.40	14.91	14.21	15.78	11.68	11.68	13.65±1.88
3-9	21.01	20.40	21.45	21.45	21.36	21.13±0.45	17.18	15.00	16.48	16.83	16.83	16.46±0.86
3-10	23.37	21.71	24.41	24.85	24.67	23.80±1.30	19.62	20.84	20.75	19.79	19.79	20.16±0.59
3-11	18.48	17.18	18.40	18.05	19.35	18.29±0.79	18.13	15.95	18.66	14.82	14.82	16.48±1.82
4-0	22.32	20.75	22.67	22.41	22.67	22.16±0.80	11.68	11.16	11.51	10.37	10.37	11.02±0.62
4-1	20.23	19.44	21.45	21.19	15.95	19.65±2.21	21.71	24.06	23.02	26.42	26.42	24.32±2.08
4-2	15.69	16.13	15.17	15.26	15.69	15.59±0.39	23.71	23.80	22.49	30.34	30.34	26.14±3.87
4-3	24.76	25.37	22.84	23.10	24.50	24.12±1.09	18.22	20.31	18.66	20.84	20.84	19.77±1.25
4-4	20.75	18.40	19.01	18.40	20.31	19.37±1.10	14.21	13.78	17.18	13.60	13.60	14.47±1.53
4-5	22.41	20.92	23.19	23.45	21.27	22.25±1.13	23.10	20.05	22.32	20.40	20.40	21.26±1.37
4-6	12.82	12.64	15.61	16.22	14.82	14.42±1.62	20.14	18.13	18.13	19.70	19.70	19.16±0.96
4-7	22.23	21.27	20.92	20.49	22.41	21.46±0.83	18.13	18.48	20.23	19.62	19.62	19.22±0.87
4-8	19.35	18.22	21.27	22.06	18.48	19.88±1.71	24.67	25.63	24.76	25.28	25.28	25.13±0.40
4-9	19.09	20.05	19.27	19.88	19.18	19.49±0.44	27.29	24.85	26.16	25.37	25.37	25.81±0.95
4-10	20.49	19.79	20.49	22.14	18.40	20.26±1.36	18.22	17.52	19.62	18.31	18.31	18.40±0.76
4-11	17.18	17.35	17.00	17.18	17.61	17.26±0.23	18.83	16.22	16.48	18.83	18.83	17.84±1.36
5-0	20.66	20.66	20.23	20.31	19.35	20.24±0.54	17.61	18.22	18.40	18.22	18.22	18.13±0.30
5-1	18.40	19.18	20.58	20.23	20.66	19.81±0.99	17.52	16.13	18.66	16.91	16.91	17.23±0.94
5-2	16.48	16.65	16.13	18.05	19.70	17.40±1.48	9.42	9.50	11.16	8.20	8.20	9.29±1.22
5-3	21.80	22.14	21.80	24.67	22.32	22.55±1.21	16.30	14.12	14.39	14.82	14.82	14.89±0.84
5-4	17.44	16.91	17.00	18.92	18.22	17.70±0.86	28.25	27.46	26.85	25.54	25.54	26.73±1.19
5-5	23.37	24.32	23.28	24.59	25.63	24.24±0.97	16.91	15.87	14.65	16.65	16.65	16.15±0.93
5-6	19.62	18.83	19.79	19.09	20.92	19.65±0.81	17.00	17.18	17.35	19.27	19.27	18.01±1.15
5-7	20.75	19.97	20.05	18.83	20.58	20.03±0.75	16.13	17.09	16.30	16.56	16.56	16.53±0.36

	MAC						MAS					
H	M1	M2	M3	M4	M5	mean±std	M1	M2	M3	M4	M5	mean±std
5-8	21.27	19.62	21.10	20.75	21.10	20.77±0.67	21.19	21.10	22.32	20.31	20.31	21.05±0.82
5-9	17.00	18.05	18.13	18.13	17.52	17.77±0.50	20.23	19.70	18.83	20.84	20.84	20.09±0.85
5-10	19.35	19.18	19.88	19.35	18.92	19.34±0.35	16.56	18.57	18.92	18.13	18.13	18.06±0.90
5-11	22.67	21.36	20.66	21.01	21.71	21.48±0.77	16.04	17.35	16.91	18.57	18.57	17.49±1.09
6-0	24.59	24.67	25.11	24.50	25.28	24.83±0.35	7.15	10.64	9.76	8.63	8.63	8.96±1.32
6-1	21.53	20.84	21.10	22.23	18.83	20.91±1.27	20.58	19.44	22.32	23.63	23.63	21.92±1.87
6-2	21.62	22.41	19.35	19.70	22.32	21.08±1.45	20.31	21.01	19.53	18.40	18.40	19.53±1.16
6-3	22.67	22.67	20.31	20.58	21.62	21.57±1.12	15.95	16.56	15.00	19.01	19.01	17.11±1.82
6-4	22.84	21.10	20.49	21.10	19.79	21.06±1.13	14.47	16.56	12.99	14.21	14.21	14.49±1.30
6-5	18.66	16.74	17.09	17.52	18.13	17.63±0.78	15.61	15.17	14.65	14.91	14.91	15.05±0.36
6-6	19.97	20.40	20.40	20.75	19.27	20.16±0.57	17.00	16.56	14.91	19.53	19.53	17.51±2.00
6-7	18.83	20.40	19.97	21.36	19.01	19.91±1.04	14.91	16.39	14.65	13.08	13.08	14.42±1.39
6-8	22.93	23.71	24.59	24.76	23.63	23.92±0.75	24.15	22.67	24.50	29.21	29.21	25.95±3.05
6-9	18.74	19.18	22.23	21.53	18.13	19.97±1.81	16.91	18.74	19.01	19.35	19.35	18.67±1.02
6-10	19.01	20.23	17.96	17.79	19.70	18.94±1.06	11.16	14.30	13.86	15.26	15.26	13.97±1.68
6-11	22.49	21.97	22.84	21.53	19.62	21.69±1.26	18.13	17.61	17.26	19.18	19.18	18.27±0.88
7-0	23.19	22.84	22.32	21.53	21.27	22.23±0.82	26.33	26.50	22.93	25.54	25.54	25.37±1.43
7-1	20.75	20.23	20.58	20.23	20.40	20.44±0.23	20.49	22.14	19.44	21.45	21.45	20.99±1.05
7-2	18.92	17.18	15.95	16.91	17.35	17.26±1.07	22.67	21.97	17.35	24.76	24.76	22.30±3.04
7-3	20.40	22.41	19.01	19.09	19.27	20.03±1.44	13.08	14.73	11.33	16.56	16.56	14.46±2.27
7-4	22.14	20.58	24.15	21.62	20.14	21.73±1.57	17.44	22.58	19.44	22.84	22.84	21.03±2.47
7-5	20.92	22.06	21.01	21.10	18.83	20.78±1.18	11.51	13.69	11.42	11.86	11.86	12.07±0.93
7-6	20.75	21.27	19.09	19.44	19.62	20.03±0.93	15.17	16.13	12.47	15.69	15.69	15.03±1.47
7-7	20.23	20.40	18.48	18.66	20.49	19.65±0.99	15.26	17.79	13.43	21.36	21.36	17.84±3.57
7-8	17.52	17.09	17.44	19.62	15.61	17.45±1.43	26.33	23.89	25.54	29.64	29.64	27.01±2.56
7-9	17.52	18.74	17.44	19.01	16.04	17.75±1.19	20.31	23.19	18.13	21.19	21.19	20.80±1.83
7-10	21.62	20.14	21.71	20.66	19.70	20.77±0.89	15.34	15.69	12.99	18.74	18.74	16.30±2.46
7-11	24.59	22.67	21.80	23.80	22.93	23.16±1.07	16.91	18.05	14.39	18.31	18.31	17.19±1.67
8-0	19.88	21.19	21.36	19.70	18.57	20.14±1.15	19.88	23.45	19.44	26.07	26.07	22.98±3.22
8-1	20.75	23.45	22.84	23.89	19.79	22.14±1.78	36.88	34.00	38.88	39.06	39.06	37.58±2.20
8-2	11.94	14.91	15.69	16.30	15.26	14.82±1.69	14.82	17.09	12.99	15.17	15.17	15.05±1.46
8-3	17.52	21.88	20.14	19.35	20.75	19.93±1.63	11.42	13.08	9.50	9.94	9.94	10.78±1.48
8-4	20.31	22.41	22.76	23.63	20.40	21.90±1.48	13.86	12.03	14.82	16.56	16.56	14.77±1.92
8-5	18.92	21.01	19.97	18.83	19.44	19.63±0.89	16.56	19.53	16.48	18.48	18.48	17.91±1.34
8-6	20.31	19.35	18.48	18.92	15.26	18.47±1.92	14.12	15.87	13.78	21.71	21.71	17.44±3.98
8-7	20.05	21.62	22.76	22.23	20.84	21.50±1.08	21.97	24.67	22.14	22.14	22.14	22.62±1.15
8-8	19.62	21.88	20.66	20.05	20.92	20.63±0.87	17.52	19.09	15.52	14.30	14.30	16.15±2.11
8-9	19.44	21.97	22.32	21.19	19.01	20.78±1.49	22.14	25.98	25.89	27.55	27.55	25.82±2.21
8-10	22.93	21.01	23.45	21.53	20.05	21.80±1.39	13.43	12.82	13.86	12.47	12.47	13.01±0.62
8-11	22.14	23.71	21.62	22.14	19.27	21.78±1.61	15.26	18.74	13.60	15.95	15.95	15.90±1.86
9-0	26.07	26.24	25.89	24.93	27.03	26.03±0.75	13.16	16.39	14.21	15.00	15.00	14.75±1.18
9-1	24.41	23.10	24.76	24.06	24.41	24.15±0.63	15.78	18.83	17.70	17.26	17.26	17.37±1.09
9-2	22.76	22.41	19.53	19.53	21.88	21.22±1.57	13.95	14.12	14.12	14.12	14.12	14.09±0.08
9-3	19.27	18.57	23.02	22.14	20.58	20.71±1.87	10.37	12.55	11.86	13.69	13.69	12.43±1.39
9-4	21.71	21.53	21.71	20.75	21.88	21.52±0.45	15.08	16.56	14.12	15.34	15.34	15.29±0.87
9-5	29.73	26.42	27.99	25.20	28.25	27.52±1.75	10.37	11.33	10.72	8.98	8.98	10.08±1.06
9-6	11.25	15.69	16.13	16.04	14.47	14.72±2.05	13.43	16.74	14.82	17.52	17.52	16.01±1.82
9-7	25.37	25.02	26.68	24.93	26.42	25.68±0.81	9.59	13.34	10.29	13.69	13.69	12.12±2.01
9-8	26.24	26.42	27.46	26.94	29.12	27.24±1.16	5.93	9.24	7.06	8.37	8.37	7.79±1.30
9-9	18.66	18.74	19.88	18.40	20.84	19.30±1.03	12.21	10.64	9.85	12.03	12.03	11.35±1.05
9-10	24.15	23.63	27.46	25.11	26.77	25.42±1.65	21.27	23.98	20.49	23.10	23.10	22.39±1.45
9-11	13.78	14.39	13.95	13.60	12.90	13.72±0.54	24.85	24.76	28.51	23.19	23.19	24.90±2.17
10-0	21.27	17.44	21.80	20.23	21.45	20.44±1.78	20.14	23.10	16.04	12.73	12.73	16.95±4.60
10-1	31.39	30.51	33.39	31.21	32.52	31.80±1.14	23.28	23.63	20.66	20.31	20.31	21.64±1.67
10-2	22.67	22.76	23.45	22.76	24.41	23.21±0.74	6.89	8.98	6.45	8.81	8.81	7.99±1.21
10-3	12.64	13.25	15.43	15.69	14.47	14.30±1.33	30.69	25.89	32.17	34.61	34.61	31.60±3.60
10-4	28.07	27.55	25.20	24.93	25.72	26.29±1.43	4.62	3.57	2.62	3.66	3.66	3.63±0.71
10-5	25.37	24.32	20.75	20.75	20.49	22.34±2.32	21.10	21.01	17.26	20.92	20.92	20.24±1.67
10-6	35.92	33.48	35.31	32.61	31.82	33.83±1.75	16.56	18.74	14.47	17.52	17.52	16.97±1.59
10-7	17.61	18.40	21.01	21.71	17.09	19.16±2.07	20.40	25.20	28.16	27.64	27.64	25.81±3.23
10-8	23.80	23.37	23.63	24.06	24.93	23.96±0.60	13.25	20.84	14.21	13.08	13.08	14.89±3.36
10-9	30.95	32.26	32.35	31.30	31.47	31.67±0.61	32.43	30.95	28.51	29.64	29.64	30.24±1.50
10-10	19.79	20.05	20.75	20.84	20.84	20.45±0.50	10.29	13.34	6.80	12.12	12.12	10.93±2.55
10-11	37.58	36.01	34.26	33.39	35.66	35.38±1.62	11.07	10.90	11.86	11.60	11.60	11.40±0.40
11-0	54.49	49.00	50.65	47.95	46.38	49.69±3.10	11.07	18.92	6.45	21.53	21.53	15.90±6.80
11-1	49.17	49.17	46.12	42.98	42.98	46.09±3.10	16.74	19.62	8.54	25.28	25.28	19.09±6.96
11-2	30.86	32.00	33.04	32.35	36.70	32.99±2.22	45.68	56.23	47.08	49.35	49.35	49.54±4.06
11-3	42.46	41.33	43.16	40.10	39.41	41.29±1.57	11.16	16.74	4.80	17.18	17.18	13.41±5.45

	MAC						MAS					
H	M1	M2	M3	M4	M5	mean±std	M1	M2	M3	M4	M5	mean±std
11-4	46.64	49.96	46.82	43.42	40.37	45.44±3.66	14.21	24.32	15.08	26.68	26.68	21.39±6.24
11-5	43.85	45.42	40.28	37.75	33.57	40.17±4.76	8.20	13.25	5.49	14.56	14.56	11.21±4.13
11-6	48.65	48.39	48.56	46.38	46.47	47.69±1.16	8.72	13.34	3.57	26.42	26.42	15.69±10.38
11-7	54.58	54.14	54.93	52.66	52.14	53.69±1.22	9.59	14.30	7.67	17.79	17.79	13.43±4.65
11-8	38.01	39.67	34.26	34.79	33.30	36.01±2.70	14.65	16.74	8.54	30.08	30.08	20.02±9.67
11-9	35.14	38.71	31.91	31.56	31.04	33.67±3.24	10.37	13.95	10.37	16.65	16.65	13.60±3.14
11-10	56.93	49.17	54.23	49.08	51.96	52.28±3.37	9.50	8.89	6.63	15.43	15.43	11.18±4.03
11-11	50.13	53.53	51.35	48.30	48.21	50.31±2.23	6.10	11.07	2.27	12.64	12.64	8.95±4.60

Table 8: MAC and MAS overlapping rate for each attention head across five models, as well as their average value with a standard deviation. M - Model.