Effective Attention Sheds Light On Interpretability

Kaiser Sun[*] **Ana Marasović**[†*]
[*]Paul G. Allen School of Computer Science & Engineering, University of Washington
[†]Allen Institute for AI
Seattle, WA, USA
huikas@cs.washington.edu, anam@allenai.org

Abstract

An attention matrix of a transformer self-attention sublayer can provably be decomposed into two components and only one of them (*effective attention*) contributes to the model output. This leads us to ask whether visualizing effective attention gives different conclusions than interpretation of standard attention. Using a subset of the GLUE tasks and BERT, we carry out an analysis to compare the two attention matrices, and show that their interpretations differ. Effective attention is less associated with the features related to the language modeling pretraining such as the separator token, and it has more potential to illustrate linguistic features captured by the model for solving the end-task. Given the found differences, we recommend using effective attention for studying a transformer's behavior since it is more pertinent to the model output by design.

1 Introduction

Attention mechanism (Bahdanau et al., 2015) is an essential component of many NLP models, including those that are built on the ubiquitous transformer architecture (Vaswani et al., 2017). As a result, visualizing attention weights is a widely used technique to interpret models' behavior (Belinkov and Glass, 2019). Despite that, the validity of this analysis method is a subject undergoing intense discussion and study in NLP (Jain and Wallace, 2019; Wiegreffe and Pinter, 2019; Serrano and Smith, 2019; Moradi et al., 2019; Mohankumar et al., 2020; Tutek and Snajder, 2020, *i.a.*).

Related to this discussion, Brunner et al. (2020) show that, under mild conditions, the attention matrix of a transformer self-attention sublayer can be written as a sum of two components. One of them is irrelevant for the model output because its product with the value matrix is zero. They term the other component as *effective attention* (formally defined in §2). We study whether effective attention gives interpretations that differ from conclusions we get by analyzing standard attention. If this is the case, interpretation of effective attention is better suited for studying transformers' internals because it is more pertinent to the model output by design.

Brunner et al. (2020) briefly discuss this by comparing standard and effective attention matrices from a single BERT head (Devlin et al., 2019) for one example. They observe that: (i) standard attention is largely concentrated on the delimiter tokens ([SEP], [CLS]) or on near-diagonal elements; (ii) effective attention is more dispersed; (iii) effective attention disregards the delimiters. They stress that we should not extrapolate too much from these observations since they are based on a single example, and that further research is needed on this topic.

In this work, we aim to reliably answer whether effective attention disregards the [SEP] and [CLS] tokens, and if so, are effective attention weights dispersed to linguistic features? To address these questions, we embrace the methodology for a quantitative analysis of the attention patterns produced by individual transformer heads proposed by Kovaleva et al. (2019). We carry out their experiments on a subset of the GLUE tasks with BERT's standard and effective attention. We show that effective attention "ignores" [SEP] and punctuation symbols (§3.1, §3.2), but not [CLS] (§3.2), and that it highlights end-task features instead (§3.1, §3.2, §3.3).[1]

2 Background: Effective Attention

Each transformer layer consists of multi-head self-attention and feedforward sublayers (Vaswani et al., 2017, see Appendix A). Brunner et al. (2020) show that the **standard attention** matrix A can be decomposed into two components, if a mild condition

[1]Our code is available at https://github.com/KaiserWhoLearns/Effective-Attention-Interpretability

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 4126–4135

is satisfied. Specifically, if the left nullspace of the value matrix V:

$$\text{LN}(V) := \{x^\top \in \mathbb{R}^{1\times d_s} | x^\top V = 0\},$$

is not trivial (contains vectors other than $\vec{0}$). This is satisfied when the maximum input sequence length is larger than the value matrix dimension (see Appendix A). The two components are: the component in the left nullspace of V ($A^{\|}$) and the component orthogonal to the nullspace ($A^{\perp}$). Notably, $A^{\|}$ does not contribute to the output of the self-attention sublayer:

$$AV = (A^{\|} + A^{\perp})V = \vec{0} + A^{\perp}V = A^{\perp}V. \quad (1)$$

The **effective attention** matrix is defined as $A^{\perp}$. If visualizations of standard and effective attention differ, interpretation of effective attention is an accurate interpretation because effective attention is what contributes to the model output (per Eq. 1).

We explain how to compute $A^{\perp}$ since that was not described in Brunner et al. (2020). We first compute the singular value decomposition (SVD) of the value matrix $V = U\Sigma W^T$. The rows of U that correspond to singular values equal to zero span $\text{LN}(V)$:

$$\text{LN}(V) = \text{span}\{u_1, \ldots, u_k\},$$

where k is the number of singular values that equal zero. We project each row a_i of the attention matrix $A \in \mathbb{R}^{d_s\times d_s}$ to $\text{LN}(V)$ to construct a projection of the *matrix* A to $\text{LN}(V)$:

$$\text{P}_{\text{LN}(V)}(a_i) = \sum_{j=1}^{k} \langle a_i, u_j\rangle u_j, \forall i \in \{1, \ldots, d_s\},$$

$$\text{P}_{\text{LN}(V)}(A) = [\text{P}_{\text{LN}(V)}(a_1), \ldots, \text{P}_{\text{LN}(V)}(a_{d_s})]^\top,$$

where $\langle\cdot,\cdot\rangle$ denotes the dot product. Finally, effective attention equals to:

$$A^{\perp} := A - \text{P}_{\text{LN}(V)}(A).$$

Effective attention is not guaranteed to be a probability distribution as some of its weights might be negative and larger than 1.

We observe that effective attention is slower to compute due to the SVD decomposition of V for each out of 144 BERT-base heads, and additional matrix multiplications (Table 3; §B). If speed is bottleneck, we recommend doing quantitative analyses with effective attention on a subset of the dev set. For qualitative analyses, common practice is already to select a subset for a manual analysis.

Dataset	Task	|Train|	|Test|
RTE	NLI	2.5K	3K
MRPC	paraphrase identification	3.7K	1.7K
QNLI	QA as NLI	105K	5.4K
SST-2	binary sentiment classification	67K	1.8K
STS-B	sentence similarity	7K	1.4K

Table 1: Specifications of the datasets.

3 What Does Effective Attention Reveal?

We compare visualizations of standard and effective attention following the methodology for analysis of the attention patterns (Kovaleva et al., 2019). We carry out our analyses using five English-language datasets in the GLUE benchmark (Wang et al., 2019): RTE (Dagan et al., 2005; Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009), MRPC (Dolan and Brockett, 2005), QNLI (Rajpurkar et al., 2016; Wang et al., 2019), SST-2 (Socher et al., 2013), and STS-B (Cer et al., 2017).[2] See Table 1 for their specifications. For each dataset, we train BERT-base with standard attention, a batch size of 8, maximum sequence length of 128, and 3 training epochs.[3] For analyzing effective attention, we replace standard with effective attention at the test time.

3.1 Classification of Attention Patterns

In this section, we start studying whether effective attention disregards the delimiter tokens.

The visualizations of attention matrices exhibit patterns (Clark et al., 2019; Vig and Belinkov, 2019). Kovaleva et al. (2019) identified five frequently occurring pattern categories:

- vertical (associated with the *delimiters tokens*)
- diagonal (either syntactic features between neighbouring words in the English language or the previous/following token attention coming from the language modeling pretraining)
- vertical + diagonal
- block (intra-sentence attention for the tasks with two distinct sequences; potentially encodes semantic and syntactic information)
- heterogeneous (as "block", more likely to capture interpretable linguistic features).

They annotated 400 BERT's attention matrices using these categories, and used them to train a

[2] We omit larger datasets (QQP, MNLI), due to the limit of our computation budget (a single Nvdia GTX1070 with 8GB memory), and CoLA/WNLI following Kovaleva et al. (2019).

[3] All other hyperparameters are set to default values in the transformers library (Wolf et al., 2020).

Task	Attention	B	D	V+D	H	V
RTE	Standard	4.50	7.40	15.20	45.10	27.90
	Effective	32.60	12.80	2.80	40.30	11.50
MRPC	Standard	3.40	10.20	14.90	39.80	31.80
	Effective	25.50	17.40	3.60	40.40	13.00
QNLI	Standard	4.70	7.40	15.20	45.10	27.90
	Effective	29.30	15.80	3.40	46.40	5.10
SST-2	Standard	38.50	6.10	0.00	37.80	17.60
	Effective	33.80	11.50	0.80	39.40	14.60
STS-B	Standard	4.00	8.20	1.80	50.40	35.50
	Effective	36.00	10.30	0.60	39.40	13.60

Table 2: Estimated percentage of the attention patterns (§3.1): block (B), diagonal (D), vertical + diagonal (V + D), heterogeneous (H), vertical (V). Effective attention exhibits different patterns than standard attention, i.e., less vertical patterns (associated with delimiter tokens) and more block patterns (associated with task features).

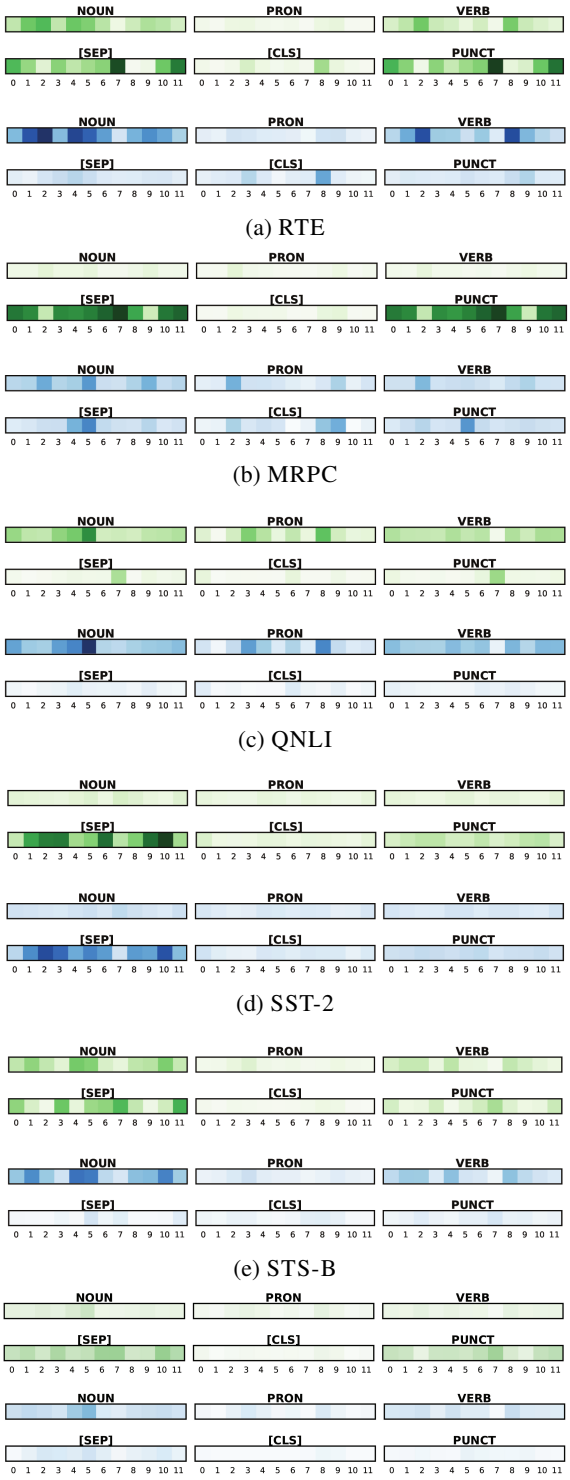
ConvNet for pattern classification of 1K random test set attention matrices. We replicate their results for standard attention (using their code), and classify effective attention matrices for a comparison.[4]

Results Table 2 (Fig. 4 in Appendix B) shows a drop in the percentage of the "vertical" and "vertical + diagonal" patterns when we replace the standard with effective attention. Since the vertical patterns are associated predominantly with attention to the delimiters tokens, this result supports the hypothesis that effective attention disregards the delimiter tokens. Moreover, although the amount of "heterogeneous" patterns did not change notably, the amount of "block" and "diagonal" patterns increased. This suggests that we are better positioned to find end-task linguistic features captured by the model by visualizing effective attention.

As an illustration, Figure 2 presents the attention matrices for one sentence from one attention head. In this example, effective attention highlights all mentions of the noun "antibiotics" that the adjective "new" modifies and that is also the object of the preposition "against", instead of giving prominence to the [SEP] token as standard attention.

3.2 Delimiter Tokens vs. Linguistic Features

We showed that the "vertical" pattern, *associated* with the delimiter tokens, is less dominant with effective attention (§3.1). To verify that both delimiter tokens are indeed less relevant with effective attention, following Kovaleva et al. (2019), we re-



(a) RTE

(b) MRPC

(c) QNLI

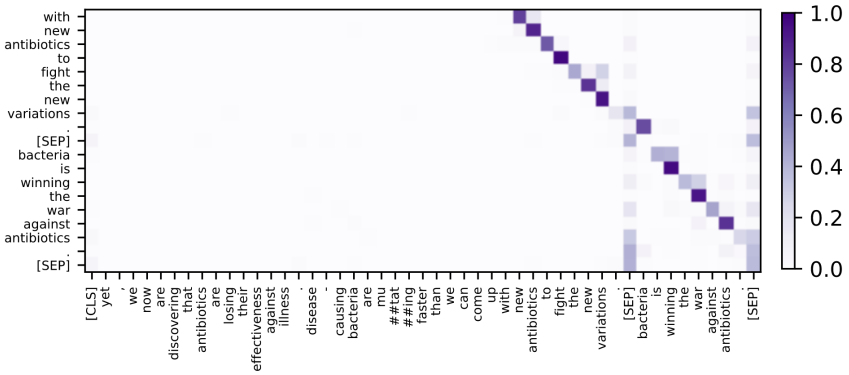
(d) SST-2

(e) STS-B

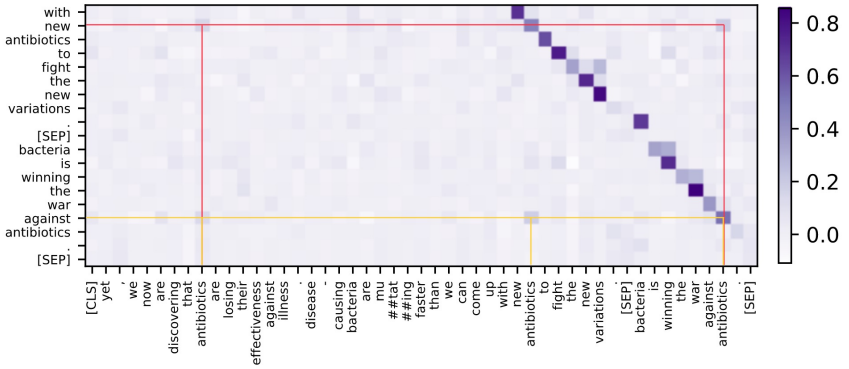
(f) AVERAGE OVER TASKS

Figure 1: Effective attention "pays less attention" to [SEP] and punctuation. Per-task and per-head (0–11) attention when processing [CLS] in the final layer, averaged over test set. The darker colors correspond to larger attention values. The green plots (two upper rows in subfigures) illustrate standard, and blue plots (two lower rows in subfigures) effective attention.

[4]We thank the authors for sharing their code and model weights for this experiment.



(a) Standard attention



(b) Effective attention

Figure 2: Visualizations of standard and effective attention from one head for one example from the RTE dataset (recognizing textual entailment). Only the last few rows are visible; see the full version in Fig. 7 (Appendix §B).

port the standard and effective attention weights of specific token types when processing the [CLS] token in the final layer. Namely, the attention weights of linguistic features (nouns, pronouns, verbs), the delimiter tokens ([SEP], [CLS]), and punctuation symbols that are conceptually similar to [SEP].[5]

Results Figure 1 shows that [SEP] is among the two most relevant features for all tasks except QNLI according to standard attention (upper two rows in each subfigure, colored green). For all but one task (SST-2), it loses its dominance with effective attention and its weights are apparently shifted to linguistic features. This is also the case for punctuation symbols. This result shows that the [SEP] token and punctuation symbols are not as important for understanding how the model solves the end-task as standard attention suggests.

We observe that [CLS] is attended similarly with effective and standard attention, contrary to what Brunner et al. suggested. To rule out this is because we plot the attention assigned to [CLS] when processing [CLS], we report the attention assigned to [CLS] when processing other input words (regardless of their type) in Fig. 5 in Appendix B. Again, we do not observe differences between standard and effective attention, unlike for [SEP] (Fig. 6 in §B). These results confirm the hypothesis of Brunner et al. that effective attention disregards [SEP], but not [CLS] as they also hypothesized. Notably, [SEP] is associated with the LM pretraining and [CLS] only with the task-specific finetuning.

3.3 Effects of Task-Specific Finetuning

To provide our final evidence that effective attention captures end-task features, we investigate how attention changes with finetuning layer-wise; again following Kovaleva et al. (2019). They calculate the cosine similarity between pretrained and finetuned flattened attention matrices. The layers that change the most, encode most task-specific features. To reiterate, effective attention is the part of standard attention that contributes to the model output (Eq. 1; §2), and we showed that it is less associated with the pretraining feature [SEP] and more with linguistic features (§3.1, §3.2). Thus, changes of standard attention from task-specific finetuning

[5] If there are multiple tokens of the same type in the input, we use the one with the maximum weight. If a word consists of the multiple subtokens, we use the weight of the first subtoken.

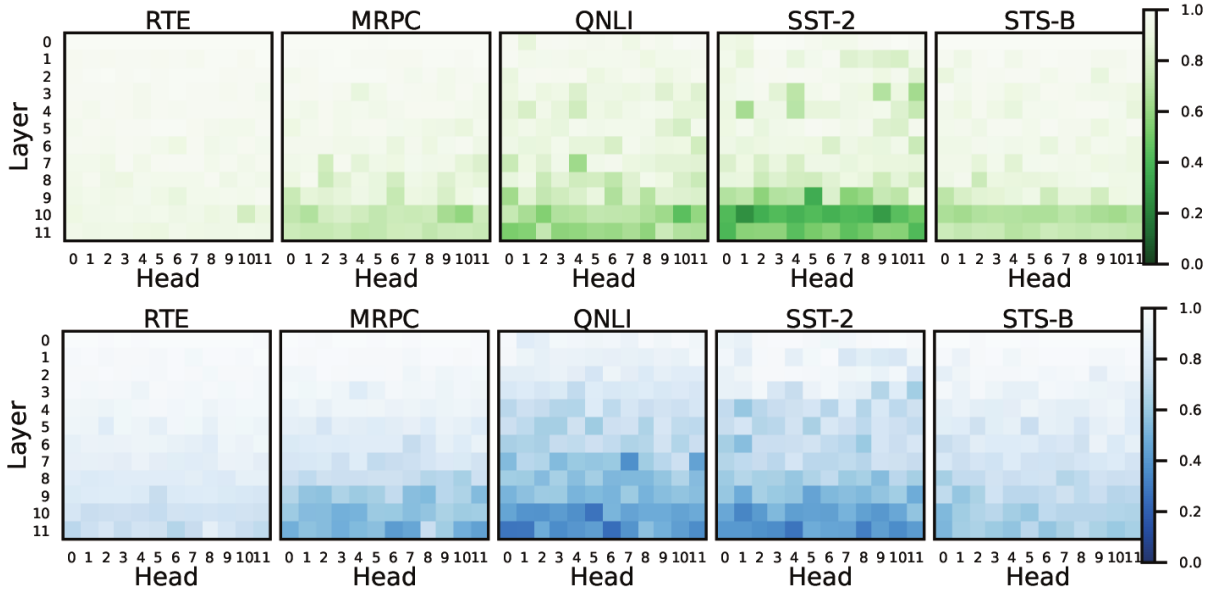


Figure 3: Per-task cosine similarity between the pretrained and finetuned attention weights for selected GLUE tasks, calculated across layers and heads. The darker colors corresponding to larger absolute attention weights. The top (green) figure is computed with the standard attention, and the bottom (blue) figure with the effective attention.

should be the product of changes of effective attention, and the outcome of this analysis should be the same, regardless of the attention "type".

Results As expected, we come to the same conclusion with effective attention as Kovaleva et al. did with the standard: the last two layers change the most with finetuning (Fig. 3). This soundness check suggests once again that effective attention is the component of standard attention that manifests end-task features.

4 Conclusions

We study whether effective attention, the part of the transformer attention matrix that does not get canceled out with the value matrix, gives different interpretations than standard attention. We present a comparison of the two attentions and show that they differ in weights assigned to delimiter tokens such as [SEP] and punctuation marks, but not [CLS] as it was previously thought. Instead, effective attention gives more weight to linguistic features. Given the differences, and that effective attention is more pertinent to the model output by design, we urge to use it for studying transformers' internals.

As an alternative to analyzing attention *weights*, Kobayashi et al. (2020) propose anayzing the norm of vectors produced by multiplying the outputs of the value matrix with the attention weights. Following the experimental setting of Clark et al. (2019), i.e., by analyzing 992 sequences extracted from Wikipedia, their norm-based analysis also shows that the contributions of [SEP] and punctuations are actually small. However, unlike us, they report the same observation for [CLS]. Future work might consider a more formal study between the norm-based analysis and effective attention, especially since the norm-based analysis could circumvent the problem of costly SVD.

Acknowledgments

The authors thank Noah A. Smith, members of Noah's ARK, as well as anonymous reviewers for their helpful feedback, and Olga Kovaleva for sharing the code and model weights for classification of attention patterns.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *the International Conference on Learning Representations*.

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72.

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo

Giampiccolo. 2009. The fifth pascal recognizing textual entailment challenge. In *TAC*.

Gino Brunner, Y. Liu, Damián Pascual, Oliver Richter, Massimiliano Ciaramita, and Roger Wattenhofer. 2020. On Identifiability in Transformers. In *the International Conference on Learning Representations*.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

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.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*, pages 177–190. Springer.

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.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9, Prague. Association for Computational Linguistics.

R Bar Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second pascal recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not Explanation. 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 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2020. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7057–7075, Online. Association for Computational Linguistics.

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.

Akash Kumar Mohankumar, Preksha Nema, Sharan Narasimhan, Mitesh M. Khapra, Balaji Vasan Srinivasan, and Balaraman Ravindran. 2020. Towards transparent and explainable attention models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4206–4216, Online. Association for Computational Linguistics.

Pooya Moradi, Nishant Kambhatla, and Anoop Sarkar. 2019. Interrogating the explanatory power of attention in neural machine translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 221–230, Hong Kong. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Sofia Serrano and Noah A. Smith. 2019. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951, Florence, Italy. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Martin Tutek and Jan Snajder. 2020. Staying true to your word: (how) can attention become explanation? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 131–142, Online. Association for Computational Linguistics.

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 *Advances in Neural Information Processing Systems*.

Jesse Vig and Yonatan Belinkov. 2019. Analyzing the structure of attention in a transformer language model. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 63–76, Florence, Italy. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *the International Conference on Learning Representations*.

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. 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 11–20, Hong Kong, China. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

A Background: On The Rank Of The Value Matrix

The output Z of an individual self-attention head is given by:

$$Q = Z_{l-1}W^Q \in \mathbb{R}^{d_s \times d_q}$$
$$K = Z_{l-1}W^K \in \mathbb{R}^{d_s \times d_k}$$
$$V = Z_{l-1}W^V \in \mathbb{R}^{d_s \times d_v}$$
$$A = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \in \mathbb{R}^{d_s \times d_s}$$
$$Z = AV \in \mathbb{R}^{d_s \times d_v},$$

where d_s is the maximum length of the input sequence (in number of subtokens), Z_{l-1} is the output of the previous transformer layer, W^Q, W^K, W^V are the query, key, and value *weight* matrices, respectively. For BERT-base, $d_q = d_k = d_v = 64$, $n_{\text{heads}} = 12$, $d_s = 512$, and $d_v \cdot n_{\text{heads}} = 768$.

Brunner et al. (2020) show that the upper bound of the rank of the value matrix V is given by:

$$\begin{aligned}\text{rank}(V) &= \text{rank}(Z_{l-1}W^V) \\ &\leq \min\{d_s, d_v, d_s, d_v \cdot n_{\text{heads}}\} \\ &\leq \min\{d_s, d_v\}.\end{aligned}$$

As a result, the left nullspace of V, defined as:

$$\text{LN}(V) := \{x^\top \in \mathbb{R}^{1 \times d_s} | x^\top V = 0\},$$

is non-trivial ($\text{LN}(V) \neq \{\vec{0}\}$) when the maximum input length, d_s, is larger than the dimension of the value matrix d_v, i.e., $d_s > d_v$. In this case, we can construct infinitely many matrices $A + \tilde{A}$,

$$\tilde{A} = [x_1, \ldots, x_{d_s}]^\top, x_i \in \text{LN}(V),$$

which contribute exactly the same to the output as the attention matrix A:

$$(A + \tilde{A})V = AV + \tilde{A}V = AV + \vec{0} = AV.$$

This also holds when the weights of $A + \tilde{A}$ are constrained to the probability simplex, and such constrained matrices $A + \tilde{A}$ exist.

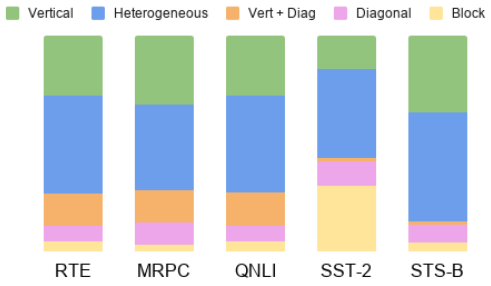
B Additional Results

We provide the following additional results that complement the discussions in Section 3:

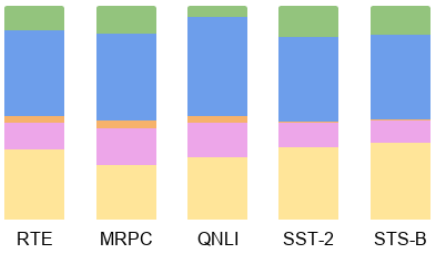
- A comparison of the evaluation time with standard vs. effective attention.
- In Figure 4, visualization of results presented in Table 2.
- Attention to the [CLS] token in Figure 5.
- Attention to the [SEP] token in Figure 6.
- Complete Figure 2.

	RTE	MRPC	QNLI	SST-2	SST-B
standard	0:29	0:45	10:59	1:41	2:54
effective	0:58	1:27	21:05	3:20	5:53

Table 3: A comparison of the evaluation clock time (minutes:seconds) of BERT models (trained with the standard attention) evaluated with standard attention and effective attention separately.



(a) Standard attention.



(b) Effective attention.

Figure 4: Estimated percentage of the attention patterns (§3.1) for each task.

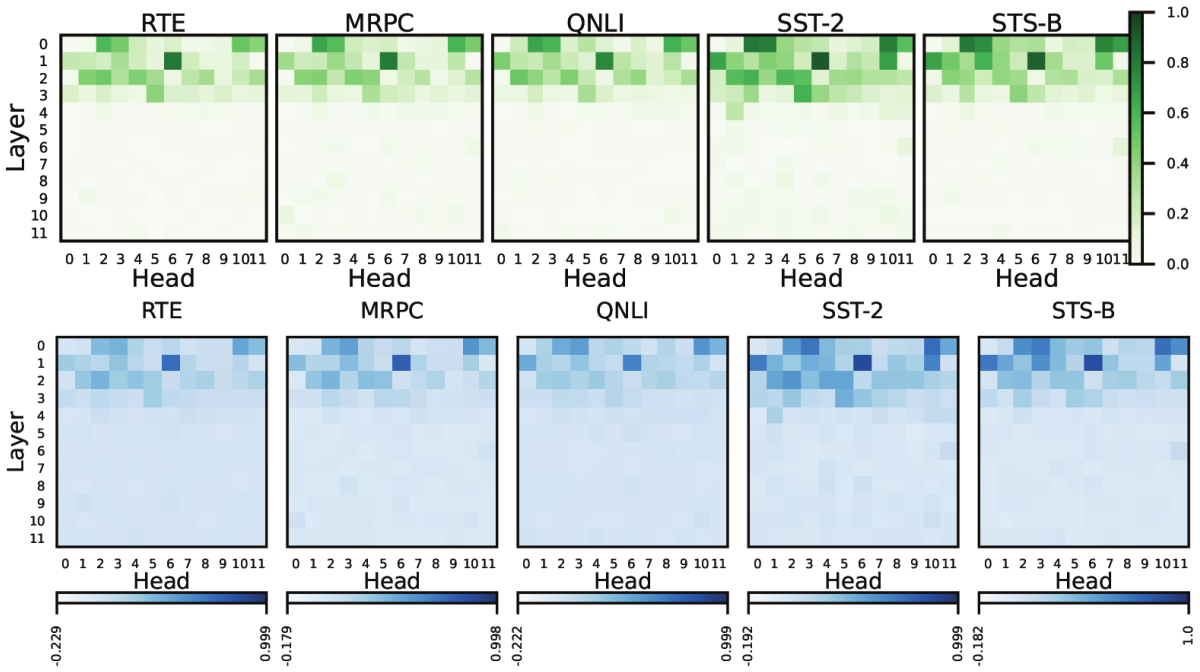


Figure 5: Per-task attention across layers and heads to the **[CLS]** token when processing other input tokens, averaged over sequence length and dataset items for the selected GLUE task. The darker colors corresponding to larger absolute attention weights. The top (green) figure is computed with the standard attention, and the bottom (blue) figure with the effective attention. Since the effective attention does not have a fixed range as the standard attention (from 0 to 1), we use the minimum and maximum effective attention weight for each task calculated across all weights (not only those associated with the [CLS] token).

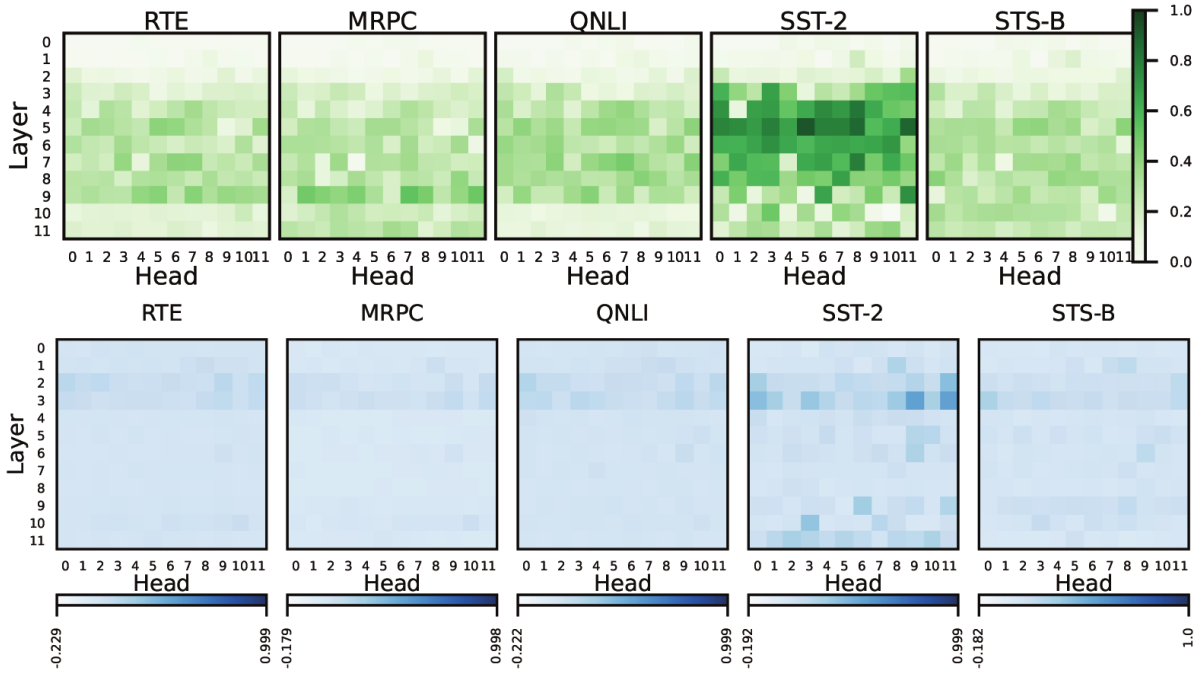
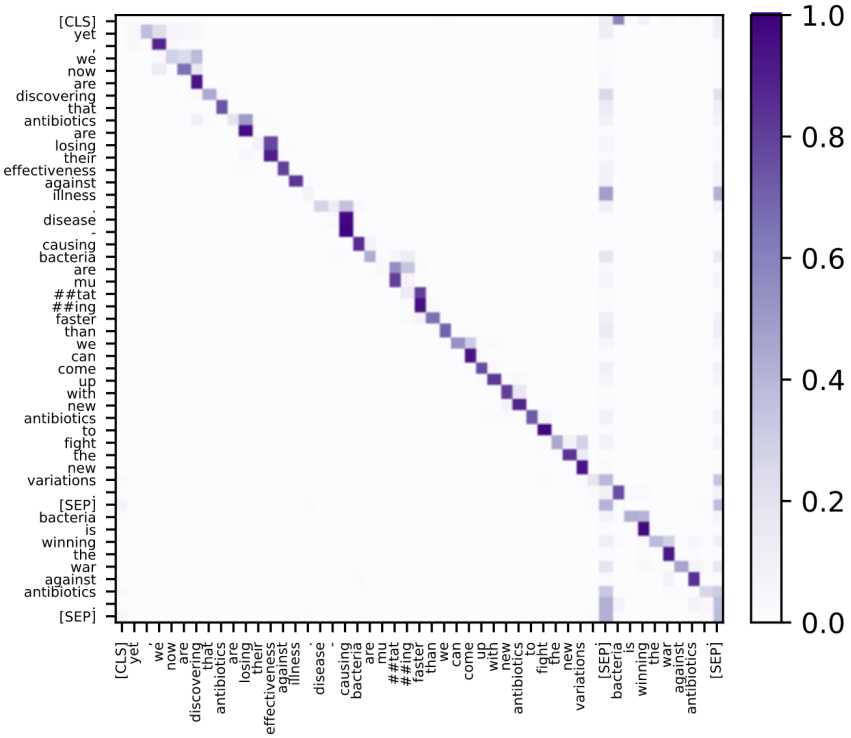
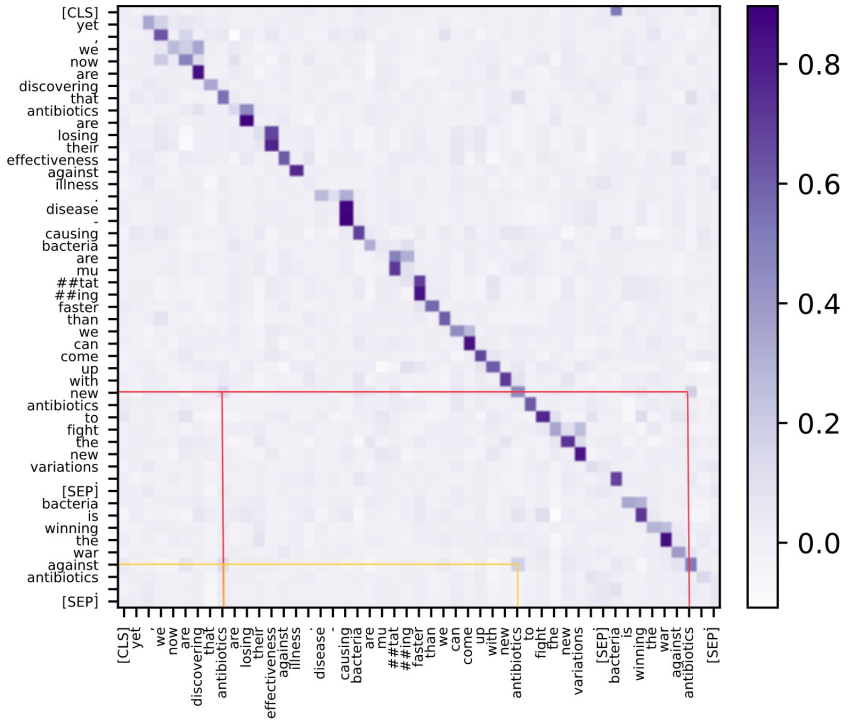


Figure 6: Per-task attention across layers and heads to the **[SEP]** token when processing other input tokens, averaged over sequence length and dataset items for the selected GLUE task. The darker colors corresponding to larger absolute attention weights. The top (green) figure is computed with the standard attention, and the bottom (blue) figure with the effective attention. Since the effective attention does not have a fixed range as the standard attention (from 0 to 1), we use the minimum and maximum effective attention weight for each task calculated across all weights (not only those associated with the [SEP] token).



(a) Standard attention



(b) Effective attention

Figure 7: Complete Figure 2.