Less is More: Attention Supervision with Counterfactuals for Text Classification

Seungtaek Choi
Yonsei University
hist0613@yonsei.ac.kr

Haeju Park
Yonsei University
phj0225@yonsei.ac.kr

Jinyoung Yeo
Yonsei University
jinyeo@yonsei.ac.kr

Seung-won Hwang
Yonsei University
seungwonh@yonsei.ac.kr

Abstract

We aim to leverage human and machine intelligence together for attention supervision. Specifically, we show that human annotation cost can be kept reasonably low, while its quality can be enhanced by machine self-supervision. Specifically, for this goal, we explore the advantage of counterfactual reasoning, over associative reasoning typically used in attention supervision. Our empirical results show that this machine-augmented human attention supervision is more effective than existing methods requiring a higher annotation cost, in text classification tasks, including sentiment analysis and news categorization.

1 Introduction

The practical importance of attention mechanism has been well-established, for both (a) improving NLP models (Vaswani et al., 2017), and also (b) enhancing human understanding of models (Serrano and Smith, 2019; Wiegreffe and Pinter, 2019).

This paper pursues the former direction, but unlike existing models, typically using attention in "unsupervised" nature. Adding human supervision to attention has been shown to improve model predictions and explanations (Jain and Wallace, 2019). For example, consider a review in (Tang et al., 2019) "*this place is small and crowded but the service is quick*". Models with unsupervised attention may attend highly on "*quick*", a generic strong signal for restaurant reviews, but one may supervise to focus on "*crowded*" to guide models to predict a negative sentiment correctly.

For this goal, *attention supervision* task (Yu et al., 2017; Liu et al., 2017) treats attention as output variables so that models can be trained to generate similar attention to human supervision. We categorize such human supervision into the following two levels:

- **Sample level rationale**: In the above example, whether to attend on *quick* or *crowded* depends on the ground-truth sentiment class. Human annotator is required to examine each training sample, and highlight important words specific to a sample and its class label.
- **Task level**: An alternative with lower annotation overhead would be annotating vocabulary, separately from training samples. That is, both *quick* and *crowded* are annotated to attend, since both have high importance for the target task of sentiment classification.

A naive belief would be assuming the former with a higher annotation cost is more effective at supervising the model's attention. Our key claim, in contrast, is that requiring **more** annotation, or, sample-specific supervision, can be less effective than requiring **less** from human then augmenting it by machine (**less-is-more**-hypothesis). Similar skepticism on asking **more**, or sample-level rationales from humans, was explored in (Bao et al., 2018), where machine attention from large additional annotations was more effective supervisions than rationales.

In this paper, we validate less-is-more without additional annotation overhead, by proposing a holistic approach of combining both human annotation and machine attention. Key distinctions from (Bao et al., 2018) are (a) humans annotate even less, and (b) without additional training resources. Specifically, we start by loosening the definition of human annotation (Camburu et al., 2018; Zhong et al., 2019) into the task-level annotation: it reduces annotation cost to the size of vocabulary, or often to zero, when public resources such as sentiment lexicon replace such annotation. We show the effectiveness of this zero-cost supervision, for both sentiment classification and news categorization scenarios, after our proposed adaptation.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 6695–6704,
November 16–20, 2020.

Our adaptation goal is an unsupervised adaptation of task-level human annotation to sample-level supervision signals for attention/classification models. Specifically, we propose Sample-level AttentioN Adaptation (SANA). Specifically, for self-supervising such adaptation, SANA conducts *what-if* tests per each sample, of whether the permutation on human annotation changes the machine prediction. That is, we collect the counterfactual (machine) supervisions for free, by observing whether highly attended word by human leads to the same machine prediction, compared to when such attention is counterfactually lowered. In such a case, SANA supervises to reduce the importance of the word. We validate such counterfactual signals are missing pieces for adapting word importance to sample-specific prediction.

We evaluate SANA on three popular datasets, SST2, IMDB, and 20NG. In all of the text classification datasets, SANA achieves significant improvements over baselines, using unsupervised attention or supervised with task- or sample-level human annotations, in the following four dimensions: Models supervised by SANA predict more accurately, explain causality of attention better, and are more robust over adversarial attacks, and more tolerant of the scarcity of training samples.

2 Preliminaries

2.1 Text Classification with Attention

Text classification assumes a dataset $D = \{\mathbf{x}_i, y_i\}_{i=1}^N$ which associates an input text $\mathbf{x}_i$ to its corresponding class label y_i. We will omit the index i when dealing with a single input sample. Let the input sequence of word features (*e.g.*, embeddings) be denoted as $\mathbf{x} = \{w_t\}_{t=1}^T$, where T is the length of the sequence. The sequence of hidden states produced by an encoding function f_ϕ with learnable parameters ϕ is then $\mathbf{h} = \{h_t\}_{t=1}^T$. Formally, $f_\phi : \mathbf{x} \rightarrow (\mathbf{h}, \hat{\alpha})$, where attention weights $\hat{\alpha} = \{\hat{\alpha}_t\}_{t=1}^T$ indicate a probability distribution over the hidden states (Zou et al., 2018; Yang et al., 2016). Finally, the hidden representations are fed into a function $g_\theta : (\mathbf{h}, \hat{\alpha}) \rightarrow \hat{y}$ with learnable parameters θ and a softmax layer that predicts the probabilities $\hat{y}$ over classes:

$$\hat{y} = \texttt{Softmax}(W^\top \tilde{\mathbf{h}} + b), \quad \theta = \{W, b\} \tag{1}$$

where $\tilde{\mathbf{h}} = \sum_{h_t \in \mathbf{h}} \hat{\alpha}_t h_t$ and $\texttt{Softmax}(z_i) = e^{z_i} / \sum_j e^{z_j}$. The parameters ϕ and θ are trained to minimize the cross-entropy loss $L_{task}(\hat{y}, y)$ between the predicted label $\hat{y}$ and the ground-truth label y.

2.2 Attention Supervision

Attention can be treated as output variables, so that humans can supervise. Given an input sample $\mathbf{x}$, let α and $\hat{\alpha}$ be the attention labels (provided by human annotators) and the trained attention weights. Then, the loss for attention supervision is defined as the cross-entropy loss $L_{att}(\hat{\alpha}, \alpha)$ between $\hat{\alpha}$ and α. Finally, the parameters of the text classification network with attention supervision are trained to minimize both loss terms together as follows:

$$L = L_{task}(\hat{y}, y) + \mu \cdot L_{att}(\hat{\alpha}, \alpha) \tag{2}$$

where μ is a preference weight.

Requiring humans to explicitly annotate soft labels α has been considered unrealistic (Barrett et al., 2018), and often delegated to implicit signals such as eye gaze. As an alternative to asking humans to annotate, important words for the given sample and class label have been typically annotated as rationale (Bao et al., 2018; Zhao et al., 2018). Formally, given an input sample $\mathbf{x}$ and its class label y, let $A \in \{0, 1\}^T$ be a binary vector of selecting words in $\mathbf{x}$, *i.e.*, $\forall w_t \in \mathbf{x} : A(w_t) \in \{0, 1\}$. Then, we convert the attention annotation A into a soft distribution of target attention labels α using softmax:

$$\alpha_t = \frac{\texttt{exp}(\lambda \cdot A(w_t))}{\sum_{t'=1}^T \texttt{exp}(\lambda \cdot A(w_{t'}))} \tag{3}$$

where λ is a positive hyper-parameter that controls the variance of scores: when λ increases, the distribution of α becomes more skewed, guiding to attend a few of more important words.

To illustrate a rationale, when given the aforementioned review sample in Sec. 1, possible annotations for the negative label are either "this place is small and crowded but the service is quick" or "this place is small and crowded but the service is quick", where the underlines indicate the hard selection by human. Then, we can translate them into the sample-level annotation $A = [1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0]$ or $A = [0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]$.

3 Less is More for Attention Supervision

Sample-level annotation is reportedly too expensive in many practical settings (Zhong et al., 2019), and is far difficult for humans to capture the dependency with corresponding class labels. In contrast, annotators may select important words for a target task, namely task-level attention annotation

(Def. 3.1), without looking up individual samples and their labels.

Definition 3.1 *(Task-level Attention Annotation) Assuming the existence of the vocabulary V, the vocab-level annotation* $A_{task} \in \{0,1\}^{|V|}$ *is a binary vector of the hard selection for words in V, i.e.,* $\forall w_t \in V : A_{task}(w_t) \in \{0,1\}$. *Based on* A_{task}*, when given an input sample* $\boldsymbol{x}$*, we can use a proxy of the sample-level annotation A, i.e.,* $\forall w_t \in \boldsymbol{x} : A(w_t) = A_{task}(w_t)$.

	Sample-level	Task-level	Reduction ratio
SST2	208K	16K	-92.3%
IMDB	5M	124K	-97.5%
20NG	232K	22K	-90.5%

Table 1: Comparison of annotation space

As shown in Tab. 1, the annotation space, which is referred to as a word set size for annotation, is 10~36 times smaller at task-level than at sample-level. Generally, the vocabulary size is far smaller than the total number of word occurrences in training samples. Our goal is thus to keep annotation cost cognitively reasonable (Zou et al., 2018; Zhao et al., 2018), leaving machine self-supervision to close the annotation quality gap (Sec. 3.1 and 3.2). Meanwhile, we present a setup of zero-cost supervision, which allows us attention supervision without any human efforts in all scenarios using public resources and tools (Sec. 3.3).

3.1 Counterfactuals as Causal Signals

Our key idea is to leverage causal signals (Johansson et al., 2016) from human annotation A (or attention labels α) of an input sample $\mathbf{x}$ to its corresponding model prediction $\hat{y}$. More specifically, we test whether two different attentions (one is original and the other is counterfactual) on the same input sample $\mathbf{x}$ lead to different prediction results $\hat{y}$. If high (original) and low (counterfactual) attention weights for an word w_t yield the same (or very similar) prediction, it provides evidence to edit the importance of word w_t in A into a lower value.

Formally, let $\hat{\alpha}$ and $\bar{\alpha}$ be the original and counterfactual attention weights, respectively, and let $\hat{y}$ and $\bar{y}_t$ be the original prediction and its counterfactual prediction with attention change (*i.e.*, from $\hat{\alpha}_t$ to $\bar{\alpha}_t$) on $w_t \in \mathbf{x}$, respectively. Then, knowing the quantity $|\hat{y} - \bar{y}_t|$, measured as the individualized treatment effect (ITE), enables measuring how much the word w_t contributes to the original prediction via attention mechanism. For this measurement, we adopt the Total Variance Distance (Jain and Wallace, 2019) between the two predictions, which is defined as follows:

$$TVD(\hat{y}, \bar{y}_t) = \frac{1}{2} \sum_{c=1}^{C} |\hat{y}^c - \bar{y}_t^c| \tag{4}$$

where c is the class index. If TVD value is too low, we can give a penalty by decaying the human annotation $A(w_t)$ with a factor of γ, which we empirically set as 0.5, to update the attention labels.

Algorithm 1 SANA

Input: Training dataset D, Task-level annotation A
Output: Model parameters $\{\phi, \theta\}$
Initialize attention labels α from A ▷ Using Eq (3)
$\{\phi, \theta\} \leftarrow \text{argmin}_{\phi,\theta} L(D, \alpha; \phi, \theta)$ ▷ Using Eq (2)
for $z = 1$ *to* z_{max} **do**
  for *each* $(\boldsymbol{x}, y) \in D$ **do**
    $\mathbf{h}, \hat{\alpha} \leftarrow f_\phi(\mathbf{x})$
    $\hat{y} \leftarrow g_\theta(\mathbf{h}, \hat{\alpha})$
    for *each* $w_t \in \boldsymbol{x}$ **do**
      if $A(w_t) > 0$ **then**
        $\bar{\alpha} \leftarrow \texttt{Counterfactuals}(\hat{\alpha}, w_t)$
        $\bar{y}_t \leftarrow g_\theta(\mathbf{h}, \bar{\alpha})$
        if $TVD(\hat{y}, \bar{y}_t) < \epsilon$ **then**
          $A(w_t) \leftarrow \gamma \cdot A(w_t)$
        end
      end
    end
  end
  $\lambda \leftarrow \gamma^{-1}\lambda$ ▷ In Eq (3)
  Update attention labels α from A ▷ Using Eq (3)
  $\{\phi, \theta\} \leftarrow \text{argmin}_{\phi,\theta} L(D, \alpha; \phi, \theta)$ ▷ Using Eq (2)
end
return $\{\phi, \theta\}$

3.2 Sample-level Attention Adaptation

Based on TVD, we propose a simple yet effective approach, Sample-level AttentioN Adaptation (SANA), to derive the sample-level machine attention from the task-level human annotation. As described in Alg. 1, SANA starts with the classification model trained with the initial attention labels α. Based on ϕ and θ, we run the classification inference several times for an input sample: one for obtaining the original attention weights $\hat{\alpha}$ and the others for *counterfactual* attention weights $\bar{\alpha}$. More specifically, we first store the hidden representations $\mathbf{h}$ and the attention weights $\hat{\alpha}$ from f_ϕ, and the original prediction $\hat{y}$. Then, for each

word w_t, `Counterfactuals` returns the counterfactual attention weights $\bar{\alpha}$, by 1) copying $\hat{\alpha}$ but 2) assigning zero to the t-th dimension and 3) renormalizing as probability distribution, and we obtain its corresponding prediction result $\bar{y}_t$ by re-using $\mathbf{h}$.

Note that, since the hidden representation at time step t contextualizes a word w_t with surrounding words, we adopt perturbing only single words in SANA, not multiple words at the same time, also enjoying the computational advantage.

Finally, based on $\hat{y}$ and $\bar{y}_t$, as defined in Eq (4), we compute TVD and update the human annotation A by threshold ϵ and decay ratio γ. Once an iteration[1] is completed over the whole training corpus, we re-train the network with the updated attention annotation and labels. For the stable update, we observe that increasing the coefficient λ in Eq (3) is crucial, as TVD is not an optimal metric, preventing α from being flattened.

3.3 Zero-cost Supervision

From this point on, for task-level supervision, we assume zero-cost human annotation efforts, either by using public resources or self-supervision.

Supervision by public resources Task-level annotation are often publicly available as resources or tools. For example, sentiment lexicon (Esuli and Sebastiani, 2006) consists of sentiment words, which are important to the sentiment classification task, and named-entity recognizer (NER) (Peters et al., 2017) can collect entity words commonly attended in news categorization task. We empirically show that both lexicon and NER can be adequate substitutes for the manual task-level annotation.

Model distillation In an extreme scenario without any human annotator and public resources, inspired by self knowledge distillation (Furlanello et al., 2018), we report results for using the attention weights of the unsupervised model as a supervision. Note, however, this is highly unlikely in practice, but reported as a lower bound accuracy, when unsupervised attention noise is propagated through distillation supervision. Using SANA is even more critical in this noisy annotation scenario, to denoise attention supervision from counterfactual reasoning, which we empirically analyze this in the subsequent section.

[1] $O(|D| \cdot T)$, where T is the maximum sequence length

4 Experiment Setup

4.1 Datasets

To validate the effectiveness of SANA, we use the following three text classification datasets, which are widely used (Wang et al., 2018; Jain and Wallace, 2019) and statistically diverse as well. We split the official training split into 90% and 10% as training and validation sets respectively. We expect SANA in two-sentence tasks, such as SNLI and MPQA, would be promising, which we leave as future work.

- **SST2** (Socher et al., 2013): *Stanford Sentiment Treebank* provides around 11K sentences tagged with sentiment on a scale from 1 (most negative) to 5 (most positive). We filter out neutral samples and dichotomize the remaining sentences into positive (4,5) and negative (1,2). We set the maximum sequence length as 30.

- **IMDB** (Maas et al., 2011): *IMDB Large Movie Review Corpus* is a binary sentiment classification dataset containing 50K polarized (positive or negative) movie reviews, split into half for training and testing. We set the maximum sequence length as 180.

- **20NG**: *20 Newsgroups*[2] contains around 19K documents evenly categorized into 20 different categories. Following (Jain and Wallace, 2019), we extract samples belonging to *baseball* and *hockey* classes, which we designate as 0 and 1, deriving a binary classification task (*Hockey vs Baseball*). We set the maximum sequence length as 300.

4.2 Implementation Details

For all datasets, we use skip-gram (Mikolov et al., 2013) (official GoogleNews-vectors-negative300) word embeddings with 300 dimensions. We use 1-layered GRU for each direction with hidden size of 150 for both SST2 and IMDB, and 300 for 20NG dataset, with g_θ of 300 dimension with 0.5 dropout rate. For attention mechanism, the size of trainable context vector is set to 100 for SST2 and 300 for IMDB and 20NG.

For attention supervision, we use the balancing coefficient $\mu = 1.0$ for SST2 and IMDB, and $\mu = 2.0$ for 20NG. Contrary to Zou et al. (2018), we

[2] http://qwone.com/~jason/20Newsgroups/

observe a larger μ is more effective for the smaller dataset. We set the contrasting coefficient $\lambda = 3$ except $\lambda = 5$ for 20NG dataset. In Alg. 1, we use decay ratio $\gamma = 2.0$ and TVD threshold $\epsilon = 0.3$. In our experiments, the decay ratio is not significantly correlated with the final accuracy, but correlated more with the convergence period. Setting $\gamma = 2.0$ leads to the reported performance within $z_{max} = 5$.

For BERT, we train BERT-base architecture with a batch size of 4 over 3 epochs. We used Adam with a learning rate of 6.25e-5 and PiecewiseLinear scheduler.

All parameters are optimized until convergence, using Adam optimizer of learning rate 0.001. The learning parameters were chosen by the best performance on the validation set. In Alg. 1, the models are additionally fine-tuned over 10 epochs for each iteration. Note that learning time longer than our setting does not contribute to improving the model accuracy.

5 Results and Discussion

We now proceed to empirically validate the effectiveness of SANA, compared to unsupervised attention, and attention supervision approaches using either task-level or sample-level annotations as baselines (shortly, **unsupervised**, **task-level**, and **sample**). For task-level annotations (*e.g.*, in SANA), we adopt pre-annotated task-level annotations without any additional human efforts: for the two sentiment tasks, we use SentiWordNet (Esuli and Sebastiani, 2006), and for 20NG task, we use entities recognized by AllenNLP NER (Peters et al., 2017). We thus present the empirical findings for the following four research questions:

RQ1: Does SANA improve model accuracy?
RQ2: Does SANA improve model robustness?
RQ3: Is SANA effective for data-scarce cases?
RQ4: Does SANA improve attention explainability?

5.1 RQ1: Classification Accuracy

The main objective of this work is to improve attention supervisions for the purpose of better text classification. Thus, we evaluate the three attention methods by their contribution to the classification performance. Tab. 2 shows the classification accuracy for three classification datasets. In the table, we can observe the proposed approach, SANA with task-level annotation, outperforms all baselines in all the datasets. Among the results,

	Accuracy		
	SST2	IMDB	20NG
BERT	91.67	94.10	93.25
unsupervised			
BiGRU	83.96	88.07	86.04
model distillation			
BiGRU	83.53	86.93	85.12
+ SANA	84.35	88.03	88.23
task-level annotation			
BiGRU	85.12	89.30	87.19
+ SANA	**85.72**	**90.10**	**89.13**

Table 2: **Classification Performance:** accuracy (%) on the three classification datasets.

SANA achieves the largest improvement over in 20NG dataset, which has the smallest training data. This suggests that SANA can also provide effective attention supervisions in data-scarce environments. To discuss this issue further, we will repeat this comparison over the varying size of training data for RQ3.

Our study also confirms two additional observations to our advantage– counterfactual 1) is effective even in model distillation setting and 2) meaningfully contributes to performance gains. More specifically, 1) SANA achieves 84.35% in SST2 dataset which is higher than the distillation only model, but lower than task-level supervised model. 2) this model gets 88.23% in 20NG dataset, which outperforms even task-level supervised model with 1.04 point gains. This also suggests the limitation of model distillation as supervision signals and supervision by public resources can provide better initial point for SANA than model distillation.

Our key contribution is to show zero-cost attention supervision can improve a simple model closer to a highly sophisticated model, such as BERT (Devlin et al., 2019) requiring more layers and data. This motivates us to supervise attention for BERT, though understanding of BERT internals, such as (Rogers et al., 2020), is mostly observational at this stage– Intervening with attention would be an interesting future work.

Our experimental results show that SANA works well in diverse scenarios, but we observe that the effectiveness is reduced when the length of target text increases (Figure 2) or token identifiability decreases (e.g., complex architecture): SANA more

effectively works when the token identifiability is improved (by adding residual connection between two recurrent layers), achieving 0.83 point gain from 89.14%, which is larger gap than 0.47 point gain without residual connection.

5.2 RQ2: Robustness in Adversarial Attacks

Having tested for the overall performance with the original datasets, we evaluate the robustness of SANA with the the adversarial datasets. Recently, adversarial examples (Zhang et al., 2019) have been employed as an evaluation tool for model robustness: while the adversarial example conveys very similar semantics of its original sample, but with small and intentional feature perturbations to cause classification models to make false predictions. For robustness analysis, we thus test whether the attention models can keep the original predictions from adversarial examples.

This experiment consists of the following steps: First, based on the original training data, we set a basic BiGRU model (without attention mechanism) as *threat* model, which an adversarial attack method aims to deceive. Second, based on the original test data, we generate paraphrase texts by using the state-of-the-art attack method (Alzantot et al., 2018) with word-level perturbations. Third, we randomly select almost 500 paraphrase texts, which succeed in changing the prediction of *threat* model, *i.e.*, adversarial examples. Finally, we report the accuracy of the three attention models over both adversarial examples and their corresponding original samples, respectively.

Tab. 3 presents the results of adversarial attacks.[3] In the table, we can find that SANA is more robust, showing the smallest gap of the classification accuracy between the original and adversarial samples. It demonstrates that, when the network is attending to the words having causal signals to the model prediction, the network becomes more robust against adversarial attacks, which is consistent with the experimental results in Lai et al. (2019). In addition to that, we observe similar results against the white-box adversarial examples (Tsai et al., 2019), where SANA improves 3.20 and 1.80 point gains from both unsupervised and supervised attentions.

[3]The reason why "Original" is different from natural accuracy in Tab. 2 is that we conduct the experiments over the original samples only paired with the adversarial examples, incurring the biases in the test set.

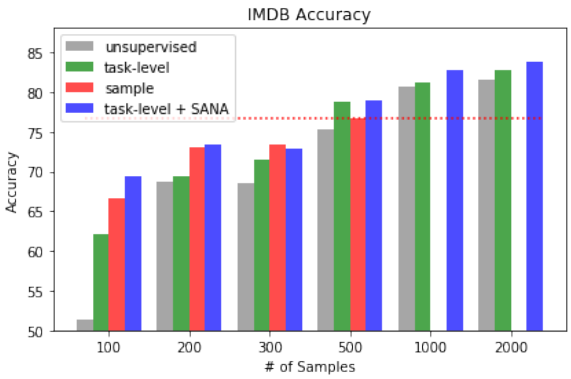


Figure 1: **Sample Effectiveness:** accuracy (%) on varying the amount of training samples in IMDB dataset.

5.3 RQ3: Sample Effectiveness

This section compares models over the varying amount of training samples in IMDB dataset, as a stress test for data-scarce scenarios.

For this experiment, we collect the sample-specific annotations from human workers. First, we randomly select 500 training samples from IMDB dataset, and ask the worker to underline the apparent rationales for the sentiment class, guided by the definition of rationale in Zhang et al. (2016). The data collection is conducted using an open annotation tool (Yang et al., 2018). Then, we build an additional method, named **sample**, which is trained with the collected sample-specific annotations.

The results are presented in Fig. 1. We notice that SANA and **sample** show much stronger performance when the training data is scarce, where similar results are reported in (Bao et al., 2018). As we expected, the attention supervision using the sample-specific annotations gets a higher accuracy than that using the task-level annotations, but cannot be scaled-up above 500 training samples, which is represented by the red reference line. In contrast, SANA improves accuracy with $\geq$ 1000 samples and its scalability. This result demonstrates that our counterfactual inferences successfully augment one annotation into multiple (counterfactual) attention supervisions, better regularizing from limited samples.

5.4 RQ4: Attention as Human Explanation

This section studies whether attention, after supervision, is more effective for human consumption as model explanation. Existing metrics for explainability measure whether attention correlates with (a) class prediction or (b) feature importance, discussed in the next sections respectively.

	SST2			IMDB			20NG		
	Original	Adversarial	|Δ|	Original	Adversarial	|Δ|	Original	Adversarial	|Δ|
unsupervised	47.2	47.8	0.6	68.8	64.1	4.7	47.7	48.3	0.6
task-level	50.3	48.3	2.1	69.2	65.0	**4.1**	48.7	48.2	0.5
task-level + SANA (Ours)	49.9	49.7	**0.2**	69.4	65.2	**4.1**	48.1	48.3	**0.2**

Table 3: **Adversarial Attack:** accuracy (%) for original and adversarial examples on the three classification dataset. Against the adversarial attacks, the proposed method SANA shows consistent performance with the smallest accuracy gap (|Δ|) over all the datasets. For this evaluation, we use 485, 532, and 478 pairs of original samples and adversarial examples, in SST2, IMDB, and 20NG respectively.

5.4.1 Attention as Causal Explanation

One measure for the explainability of attention is whether each attention weight captures the causality of word and class prediction, by permuting words and observing prediction changes. If the learning is successful, such causal signals should be consistently observed in the test predictions. To validate this, we employ the attention-permutation experiments designed in (Jain and Wallace, 2019), *i.e.*, *what-if* simulation. Specifically, when given an input sample in the test phase, we look into whether the randomly mutated attention (*i.e.*, cause) from the original attention yields any changes in the corresponding prediction result (*i.e.*, effect). Here, TVD for the permutation can be regarded as a desirable evaluation measure: as TVD is lower, the (original) learned attention has a weak mapping with the model prediction, and vice versa.

The results are presented in Fig. 2, where x-axis refers to TVD values, *i.e.*, the difference of model predictions, and y-axis refers to the frequency of *what-if* simulations on their returning TVD value. To carefully analyze this, we divide the simulation results by four different intervals of input sequence length, which can be an influencing factor: as the perturbations on longer texts are unlikely to make prediction changes (Sen et al., 2020).

In this figure, we can observe that SANA has the lowest frequency on $TVD = 0$ in all cases, showing the distribution skewed to larger TVD (*i.e.*, right on x-axis) compared to baselines. Such distribution suggests that attention in SANA strongly affects model prediction by the causal signals. In unsupervised and vocab (*i.e.*, task-level), the distributions are skewed to lower TVD (*i.e.*, left on x-axis), having larger frequency on zero TVD than SANA. These patterns indicate the baselines have weak attentions loosely aligned to model predictions, motivating SANA even working well in long texts.

5.4.2 Attention as Importance Indicator

As an alternative metric of attention explainablity, (Jain and Wallace, 2019) considers the relationship between attention weights and gradient-based feature importance score of each word.

However, prior research suggests using word as a unit of importance feature is rather artificial, as word is contextualized by, and interacts with other words: (Wiegreffe and Pinter, 2019) observes such limitation, and Shapley (Chen et al., 2018) measures interaction between features for capturing dependency of arbitrary subsets.

For this purpose, we report the KL divergence between C-Shapley[4] and attention weights, $D_{\mathbb{KL}}(\texttt{Shapley}(x) \parallel \texttt{attention}(x))$. We present the results in Tab. 4, showing SANA approach is the most well correlated method with Shapley scores, well capturing word dependency.

	unsupervised	task-level	SANA
IMDB	52.62	12.69	8.86

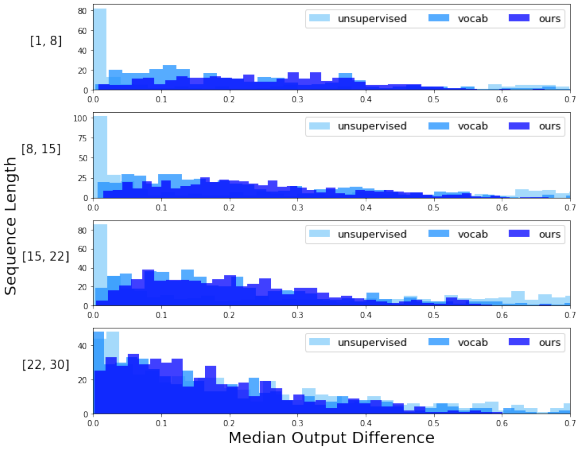
Table 4: KL-divergence from C-Shapley

Intuitively, C-Shapley observes the interaction in n-gram, and our work, attending upon hidden representations of RNN, which are *soft* n-grams, captures similar interactions. This result manifests that, standing on self-supervision signals, our counterfactual process can improve the explanation on the contextualization ability of RNN architectures.

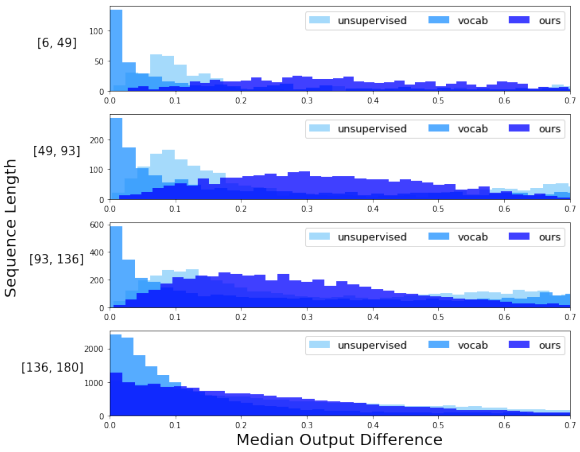
6 Related Work

Instead of treating attention as a by-product of model training, the following work explored how machine/human can **consume** attention for model improvement or explanation, respectively. Machine/human may also **provide** supervision. We thus categorize existing work by machine/human

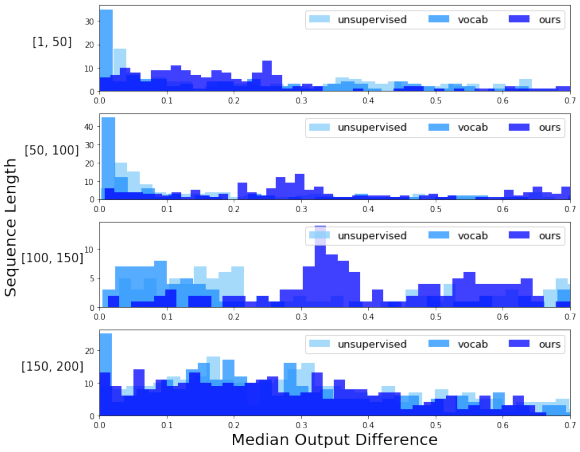
[4] https://github.com/Jianbo-Lab/LCShapley



(a) SST2



(b) IMDB



(c) 20NG

Figure 2: **Attention Analysis:** *x*-axis refers to *TVD* values returned by *what-if* simulations and *y*-axis refers to the simulation frequency according to the returning *TVD* value. The compared datasets are (a) SST2 for sentence-level binary classification, (b) IMDB and (c) 20NG for document-level binary classification.

consumption and supervision. Our work falls into human providing supervision (with machine augmenting supervision) for machine consumption.

6.1 Attention to/from Human

As for human consuming attention as explanation, there has been criticism that unsupervised attention weights are too poorly correlated with the contribution of each word for machine decision (or, unfaithful) (Jain and Wallace, 2019; Serrano and Smith, 2019; Pruthi et al., 2019). Meanwhile, (Wiegreffe and Pinter, 2019) develops diagnostics to decide when attention is good enough as explanation.

As for improving human consumption, one direction focuses on better aligning models to human, another on improving annotation quality.

First, *identifiability* (Brunner et al., 2020) explains human-machine discrepancy, where token-level information is lost in model hidden states. For better alignment, (Tutek and Šnajder, 2020) utilizes masked language model (MLM) loss and (Mohankumar et al., 2020) invents orthogonal LSTM representations.

Second, toward the direction of improving annotation, (Barrett et al., 2018; Zhong et al., 2019; Bao et al., 2018) adopts sample-specific human annotations. In addition to rationales, (Zhao et al., 2018) uses event trigger words and (Kim and Kim, 2018) leverages user authenticated domains to narrow down the scope of attentions. (Strubell et al., 2018) injects word dependency relations to recognize the semantic roles in text. Such annotation overhead can be replaced by existing pre-annotated resources: (Zou et al., 2018) considers sentiment lexicon dictionary for a related task.

We pursue the second direction, but without incurring additional human annotation, by exploring the counterfactual augmentation, originated from self-supervision signals, contributing towards both accuracy and robustness of the model.

6.2 Attention to/from Machine

Machine consuming attention for higher accuracy is the most classical target scenario. (Yang et al., 2016) proposes hierarchical attention for document classification, (Chen et al., 2016) personalizes classification to user and product attributes. (Margatina et al., 2019) incorporates knowledge information to the self-attention module, *i.e.*, lexicon features.

Alternatively, machine may mine or augment attention supervision: (Tang et al., 2019) automatically mines attention supervision by masking-out

highly attentive words in a progressive manner. (Choi et al., 2019) augments counterfactual observations to debias human attention supervision via instance similarity. Our work is of combining the strength of the two works: we automatically improve attention supervision via self-supervision signals, but we build it with free task-level resources.

7 Conclusion & Future Work

We studied the problem of attention supervision, and showed that requiring sample-level human supervision is often less effective than task-level alternative with lower (and often zero-) overhead. Specifically, we proposed a counterfactual signal for self-supervision, to augment task-level human annotation, into sample-level machine attention supervision, to increase both the accuracy and robustness of the model. We hope future research to explore scenarios where human intuition is not working as well as text classification, such as graph attention (Veličković et al., 2017).

Acknowledgments

This work is supported by AI Graduate School Program (2020-0-01361) and IITP grant (No.2017-0-01779, XAI) supervised by IITP. Hwang is a corresponding author.

References

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. In *EMNLP*.

Yujia Bao, Shiyu Chang, Mo Yu, and Regina Barzilay. 2018. Deriving machine attention from human rationales. *arXiv preprint*.

Maria Barrett, Joachim Bingel, Nora Hollenstein, Marek Rei, and Anders Søgaard. 2018. Sequence classification with human attention. In *CoNLL*.

Gino Brunner, Yang Liu, Damian Pascual Ortiz, Oliver Richter, Massimiliano Ciaramita, and Roger Wattenhofer. 2020. On identifiability in transformers.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: natural language inference with natural language explanations. In *NeurIPS*.

Huimin Chen, Maosong Sun, Cunchao Tu, Yankai Lin, and Zhiyuan Liu. 2016. Neural sentiment classification with user and product attention. In *EMNLP*.

Jianbo Chen, Le Song, Martin J Wainwright, and Michael I Jordan. 2018. L-shapley and c-shapley: Efficient model interpretation for structured data.

Seungtaek Choi, Haeju Park, and Seung-won Hwang. 2019. Counterfactual attention supervision. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 1006–1011. IEEE.

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.

Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *LREC*.

Tommaso Furlanello, Zachary C Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. 2018. Born again neural networks. *ICML*.

Sarthak Jain and Byron C Wallace. 2019. Attention is not explanation. *arXiv preprint*.

Fredrik Johansson, Uri Shalit, and David Sontag. 2016. Learning representations for counterfactual inference. In *ICML*.

Joo-Kyung Kim and Young-Bum Kim. 2018. Supervised domain enablement attention for personalized domain classification. In *EMNLP*.

Qiuxia Lai, Wenguan Wang, Salman Khan, Jianbing Shen, Hanqiu Sun, and Ling Shao. 2019. Human vs machine attention in neural networks: A comparative study. *arXiv preprint*.

Chenxi Liu, Junhua Mao, Fei Sha, and Alan L Yuille. 2017. Attention correctness in neural image captioning. In *AAAI*.

Andrew L Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *ACL*.

Katerina Margatina, Christos Baziotis, and Alexandros Potamianos. 2019. Attention-based conditioning methods for external knowledge integration. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3944–3951.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*.

Akash Kumar Mohankumar, Preksha Nema, Sharan Narasimhan, Mitesh M Khapra, Balaji Vasan Srinivasan, and Balaraman Ravindran. 2020. Towards transparent and explainable attention models. *arXiv preprint arXiv:2004.14243*.

Matthew Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-supervised sequence tagging with bidirectional language models. In *ACL*.

Danish Pruthi, Mansi Gupta, Bhuwan Dhingra, Graham Neubig, and Zachary C Lipton. 2019. Learning to deceive with attention-based explanations. *arXiv preprint arXiv:1909.07913*.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in bertology: What we know about how bert works. *arXiv preprint arXiv:2002.12327*.

Cansu Sen, Thomas Hartvigsen, Biao Yin, Xiangnan Kong, and Elke Rundensteiner. 2020. Human attention maps for text classification: Do humans and neural networks focus on the same words? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Sofia Serrano and Noah A Smith. 2019. Is attention interpretable? *arXiv preprint*.

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 *EMNLP*.

Emma Strubell, Patrick Verga, Daniel Andor, David Weiss, and Andrew McCallum. 2018. Linguistically-informed self-attention for semantic role labeling. In *EMNLP*.

Jialong Tang, Ziyao Lu, Jinsong Su, Yubin Ge, Linfeng Song, Le Sun, and Jiebo Luo. 2019. Progressive self-supervised attention learning for aspect-level sentiment analysis. In *ACL*.

Yi-Ting Tsai, Min-Chu Yang, and Han-Yu Chen. 2019. Adversarial attack on sentiment classification. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 233–240.

Martin Tutek and Jan Šnajder. 2020. Staying true to your word:(how) can attention become explanation? *arXiv preprint arXiv:2005.09379*.

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 *NeurIPS*.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355.

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. *arXiv preprint*.

Jie Yang, Yue Zhang, Linwei Li, and Xingxuan Li. 2018. Yedda: A lightweight collaborative text span annotation tool. In *ACL*.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alexander J Smola, and Eduard H Hovy. 2016. Hierarchical attention networks for document classification. In *NAACL*.

Licheng Yu, Mohit Bansal, and Tamara Berg. 2017. Hierarchically-attentive rnn for album summarization and storytelling. In *EMNLP*.

Wei Emma Zhang, Quan Z Sheng, and Ahoud Abdulrahmn F Alhazmi. 2019. Generating textual adversarial examples for deep learning models: A survey. *arXiv preprint*.

Ye Zhang, Iain Marshall, and Byron C Wallace. 2016. Rationale-augmented convolutional neural networks for text classification. In *EMNLP*.

Yue Zhao, Xiaolong Jin, Yuanzhuo Wang, and Xueqi Cheng. 2018. Document embedding enhanced event detection with hierarchical and supervised attention. In *ACL*.

Ruiqi Zhong, Steven Shao, and Kathleen McKeown. 2019. Fine-grained sentiment analysis with faithful attention. *arXiv preprint*.

Yicheng Zou, Tao Gui, Qi Zhang, and Xuanjing Huang. 2018. A lexicon-based supervised attention model for neural sentiment analysis. In *COLING*.