Corpus-level and Concept-based Explanations for Interpretable Document Classification

TIAN SHI, XUCHAO ZHANG, PING WANG, and CHANDAN K. REDDY, Virginia Tech

Using attention weights to identify information that is important for models' decision making is a popular approach to interpret attention-based neural networks. This is commonly realized in practice through the generation of a heat-map for every single document based on attention weights. However, this interpretation method is fragile and it is easy to find contradictory examples. In this article, we propose a corpus-level explanation approach, which aims at capturing causal relationships between keywords and model predictions via learning the importance of keywords for predicted labels across a training corpus based on attention weights. Based on this idea, we further propose a concept-based explanation method that can automatically learn higher level concepts and their importance to model prediction tasks. Our concept-based explanation method is built upon a novel Abstraction-Aggregation Network (AAN), which can automatically cluster important keywords during an end-to-end training process. We apply these methods to the document classification task and show that they are powerful in extracting semantically meaningful keywords and concepts. Our consistency analysis results based on an attention-based Naïve Bayes classifier (NBC) also demonstrate that these keywords and concepts are important for model predictions.

CCS Concepts: • **Information systems → Sentiment analysis**; **Clustering and classification**; *Information extraction;*

Additional Key Words and Phrases: Attention mechanism, model interpretation, document classification, sentiment classification, concept-based explanation

ACM Reference format:
Tian Shi, Xuchao Zhang, Ping Wang, and Chandan K. Reddy. 2021. Corpus-level and Concept-based Explanations for Interpretable Document Classification. *ACM Trans. Knowl. Discov. Data.* 16, 3, Article 48 (October 2021), 17 pages.
https://doi.org/10.1145/3477539

1 INTRODUCTION

Attention Mechanisms [2] have boosted performance of deep learning models in a variety of **natural language processing (NLP)** tasks, such as sentiment analysis [35, 47], semantic parsing [46], machine translation [30], reading comprehension [9, 17], and others. Attention-based deep learning models have been widely investigated not only because they achieve state-of-the-art performance, but also because they can be interpreted by identifying important input information via

This work was supported in part by the US National Science Foundation grants IIS-1707498, IIS-1838730, and Amazon AWS credits.

Authors' addresses: T. Shi, X. Zhang, P. Wang, and C. K. Reddy, Virginia Tech, Blacksburg, VA 24061; emails: {tshi, xuczhang, ping, reddy}@vt.edu.

1556-4681/2021/10-ART48 $15.00
https://doi.org/10.1145/3477539

visualizing heat-maps of attention weights [11, 40, 42, 45], namely attention visualization. Therefore, attention mechanisms help end-users to understand models and diagnose trustworthiness of their decision making.

However, the attention visualization approach still suffers from several drawbacks: (1) The fragility of attention weights can easily make end-users find contradicting examples, especially for noisy data and cross-domain applications. For example, a model may attend on punctuation or stop-words. (2) Attention visualization cannot automatically extract high-level concepts that are important for model predictions. For example, when a model assigns news articles to *Sports*, relevant keywords may be *player*, *basketball*, *coach*, *nhl*, *golf*, and *nba*. Obviously, we can build three concepts/clusters for this example, i.e., roles (*player*, *coach*), games (*basketball*, *soccer*), and leagues (*nba*, *nhl*). (3) Attention visualization still relies on human experts to decide if keywords attended by models are important to model predictions.

There have been some studies that attempt to solve these problems. For example, Jain and Wallace [20], Serrano and Smith [38] focused on studying if attention can be used to interpret a model, however, there are still problems in their experimental designs [48]. Yeh et al. [53] tried to apply a generic concept-based explanation method to interpret BERT models in the text classification task, however, they did not obtain semantically meaningful concepts for model predictions. Antognini and Faltings [1] introduced a concept explanation method that first extracts a set of text snippets as concepts and infers which ones are described in the document, and then it explained the predictions of sentiment with a linear aggregation of concepts. In this article, we propose a general-purpose corpus-level explanation method and a concept-based explanation method based on a novel **Abstraction-Aggregation Network** (**AAN**) to tackle the aforementioned drawbacks of attention visualization. We summarize the primary contributions of this article as follows:

— To solve the first problem, we propose a *corpus-level explanation* method, which aims at discovering causal relationships between keywords and model predictions. The importance of keywords is learned across the training corpus based on attention weights. Thus, it can provide more robust explanations compared with attention visualization case studies. The discovered keywords are semantically meaningful for model predictions.
— To solve the second problem, we propose a *concept-based explanation method* (case-level and corpus-level) that can automatically learn semantically meaningful concepts and their importance to model predictions. The concept-based explanation method is based on an AAN that can automatically cluster keywords, which are important to model predictions, during the end-to-end training for the main task. Compared to the basic attention mechanisms, the models with AAN do not compromise on classification performance or introduce any significant number of new parameters.
— To solve the third problem, we build a ***Naïve Bayes Classifier*** (**NBC**), which is based on an *attention-based bag-of-words document representation* technique and the causal relationships discovered by the corpus-level explanation method. By matching predictions from the model and NBC, i.e., consistency analysis, we can verify if the discovered keywords are important to model predictions. This provides an automatic verification pipeline for the results from the corpus-level explanation and concept-based explanation methods.

The rest of this article is organized as follows: In Section 2, we introduce related work of feature-based and concept-based explanation. In Section 3, we first present details of our proposed AAN, and then discuss corpus-level and concept-based explanation methods. In Section 4, we evaluate different self-attention and AAN-based models on three different datasets. We also show how corpus-level and concept-based explanations can help in the interpretation of attention-based classification models and can potentially provide a better understanding of the training corpus. Our discussion concludes in Section 5.

2 RELATED WORK

Increasing the interpretability on machine learning models has become an important topic of research in recent years. Most prior work [14, 27, 29, 41] focus on interpreting models via feature-based explanations, which alters individual features such as pixels and word-vectors in the form of either deletion [36] or perturbation [43]. However, these methods usually suffer from the reliability issues when adversarial perturbations [12] or even simple shifts [23] of the input data. Moreover, the feature-based approaches explain the model behavior locally [36] for each data samples without a global explanation [13, 21] on how the models make their decisions. In addition, feature-based explanation is not necessarily the most effective way for human understanding.

To alleviate the issues of feature-based explanation models, some researchers have focused on explaining the model results in the form of high-level human concepts [4, 6, 8, 44, 50, 54, 56]. Unlike assigning the importance scores to individual features, the concept-based methods use the corpus-level concepts as the interpretable unit. For instance, concept “wheels” can be used for detecting the vehicle images and concept “Olympic Games” for identifying the sports documents. However, most of the existing concept-based approaches require human supervision in providing hand-labeled examples of concepts, which is labor intensive and some human bias can be introduced in the explanation process [19, 21, 39]. Recently, automated concept-based explanation methods [5, 53] are proposed to identify higher-level concepts that are meaningful to humans automatically. However, they have not shown semantically meaningful concepts on text data. In text classification area, most of the existing approaches focus on improving the classification performance, but ignore the interpretability of the model behaviors [52]. Liu and Avci [27] utilize the feature attribution method to help users interpret the model behavior. Bouchacourt and Denoyer [5] propose a self-interpretable model through unsupervised concept extraction. However, it requires another unsupervised model to extract concepts. Different from these studies, our corpus-level explanation method can be generally applied to self-attention mechanisms, and our concept-based explanation method, which is based on a hierarchical attention network, can automatic learn concepts during the end-to-end training.

3 PROPOSED WORK

In this section, we first introduce the classification framework and our AAN. Then, we discuss the *corpus-level explanation*, *concept-based explanation*, and attention-based NBC.

3.1 The Proposed Model

3.1.1 Basic Framework. A typical document classification model is equipped with three components, i.e., an encoder, an attention or pooling layer, and a classifier. (1) **Encoder:** An encoder reads a document, denoted by $d = (w_1, w_2, \ldots, w_T)$, and transforms it to a sequence of hidden states $H = (h_1, h_2, \ldots, h_T)$. Here, w_t is the one-hot representation of token t in the document. h_t is also known as a word-in-context representation. Traditionally, the encoder consists of a word embedding layer followed by an LSTM [18] sequence encoder. Recently, pre-trained language models [9, 34, 51] have emerged as an important component for achieving superior performance on a variety of NLP tasks including text classification. Our model is adaptable to any of these encoders. (2) **Attention/Pooling:** The attention or pooling (average- or max-pooling) layer is used to construct a high-level document representation, denoted by v^{doc}. In attention networks, the attention weights show the contributions of words to the representations [26, 52]. Compared with pooling, attention operations can be well interpreted by visualizing attention weights [52]. (3) **Classifier:**

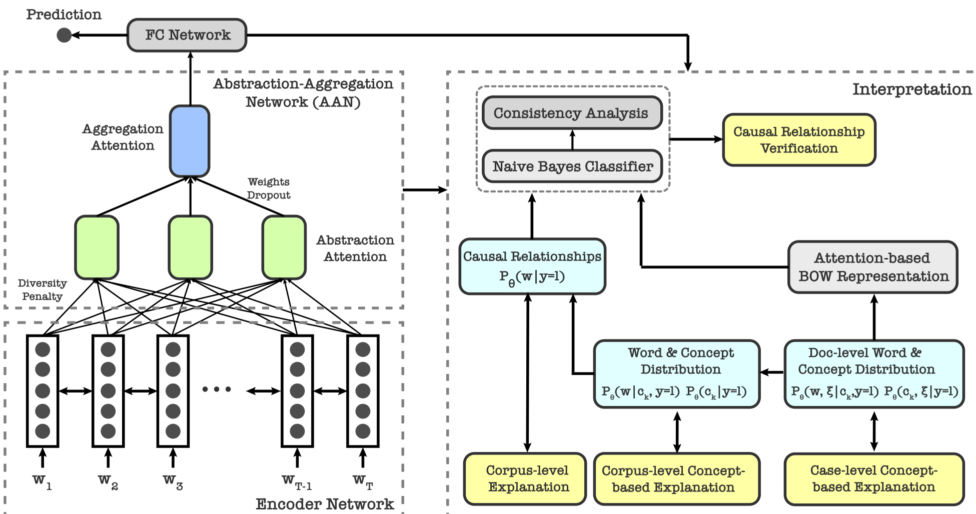


Fig. 1. The proposed AAN and different interpretation methods.

The document representation is passed into a classifier to get the probability distribution over different class labels. The classifier can be a multi-layer feed-forward network with activation layer followed by a softmax layer, i.e., $y = \text{softmax}(W_2 \cdot \text{ReLU}(W_1 \cdot v^{\text{doc}} + b_1) + b_2)$, where W_1, W_2, b_1 and b_2 are model parameters.

To infer parameters, we can minimize the averaged cross-entropy error between predicted and ground-truth labels. Here, loss function is defined as $\mathcal{L}_\theta = -\sum_{l=1}^{L} \hat{y} \log(y)$, where $\hat{y}$ represents the ground-truth label and L is the number of class labels. The model is trained in an end-to-end manner using back-propagation.

3.1.2 Abstraction-Aggregation Network. In order to use different explanation methods, especially concept-based explanation, to interpret deep neural networks, we propose a novel AAN for the Attention/Pooling layer, which first captures keywords for different concepts from a document, and then aggregates all concepts to construct the document representation (see Figure 1).

An AAN has two stacked attention layers, namely, *abstraction-attention (abs)* and *aggregation-attention (agg)* layers. In the *abs* layer, for each attention unit k, we calculate the alignment score $u_{k,t}^{\text{abs}}$ and attention weight $\alpha_{k,t}^{\text{abs}}$ as follows:

$$u_{k,t}^{\text{abs}} = (g_k^{\text{abs}})^\top h_t,$$
$$\alpha_{k,t}^{\text{abs}} = \frac{\exp(u_{k,t}^{\text{abs}})}{\sum_{\tau=1}^{T} \exp(u_{k,\tau}^{\text{abs}})}, \tag{1}$$

where g_k^{abs} are model parameters. Here, we do not apply linear transformation and tanh activation when calculating alignment scores for two reasons: (1) **Better intuition:** Calculating attention between g_k^{abs} and h_t in Equation (1) is the same as calculating a normalized similarity between them. Therefore, abstraction-attention can also be viewed as a clustering process, where g_k^{abs} determines the centroid of each cluster. In our model, concepts are related to the clusters discovered by AAN. (2) **Fewer parameters:** Without the linear transformation layer, the abstraction-attention layer only introduces $K \times |h_t|$ new parameters, where $|h_t|$ is the dimension of h_t and $K \ll |h_t|$. The k th representation is obtained by $v_k^{\text{abs}} = \sum_{t=1}^{T} \alpha_{k,t}^{\text{abs}} h_t$. We use K to denote the total number of attention units.

In the *agg* layer, there is only one attention unit. The alignment score u_k^{agg} and attention weight α_k^{agg} are obtained by

$$u_k^{\text{agg}} = (g^{\text{agg}})^\top \tanh(W_{\text{agg}} v_k^{\text{abs}} + b_{\text{agg}}),$$

and

$$\alpha_k^{\text{agg}} = \frac{\exp(u_k^{\text{agg}})}{\sum_{\kappa=k}^{K} \exp(u_\kappa^{\text{agg}})},$$

where $W_{\text{agg}}, b_{\text{agg}}$, and g^{agg} are model parameters. The final document representation is obtained by $v^{\text{doc}} = \sum_{k=1}^{K} \alpha_k^{\text{agg}} v_k^{\text{abs}}$. It should be noted that AAN is different from hierarchical attention [52], which aims at getting a better representation. However, AAN is used to automatically capture concepts/clusters. We have also applied two important techniques to obtain semantically meaningful concepts.

(1) Diversity penalty for abstraction-attention weights: To encourage the diversity of concepts, we introduce a new penalization term to abstraction-attention weights $A = [\overrightarrow{\alpha}_1^{\text{abs}}, \overrightarrow{\alpha}_2^{\text{abs}}, \ldots, \overrightarrow{\alpha}_K^{\text{abs}}] \in \mathbb{R}^{T \times K}$, where $\overrightarrow{\alpha}_k^{\text{abs}} = (\alpha_{k,1}^{\text{abs}}, \alpha_{k,2}^{\text{abs}}, \ldots, \alpha_{k,T}^{\text{abs}})^\top$. We define the penalty function as

$$\mathcal{L}_{\text{div}} = \frac{1}{K} \|A^\top A - I\|_F, \tag{2}$$

where $\|\cdot\|_F$ represents the Frobenius norm of a matrix. Hence, the overall loss function is expressed as $\mathcal{L} = \mathcal{L}_\theta + \mathcal{L}_{\text{div}}$.

(2) Dropout of aggregation-attention weights: In the aggregation-attention layer, it is possible that $\alpha_k^{\text{agg}} \approx 1$ for some k, and other attention weights tend to be 0. To alleviate this problem, we apply dropout with a small dropout rate to aggregation-attention weights $(\alpha_1^{\text{agg}}, \alpha_2^{\text{agg}}, \ldots, \alpha_K^{\text{agg}})$, namely attention weights dropout. It should be noted that a large dropout rate has a negative impact on the explanation, since it discourages the diversity of concepts. More specifically, the model will try to capture keywords in the dropped abstraction-attention units by the other units.

3.2 Explanation

In this section, we discuss corpus-level and concept-based explanations. Given a corpus C with $|C|$ documents, we use d or ξ to represent a document. Let us also use θ to denote all parameters of a model and $\mathcal{V}$ to represent the vocabulary, where $|\mathcal{V}|$ is the size of $\mathcal{V}$. Throughout this article, we will assume that both prior document probability $p(d)$ and prior label probability $p_\theta(y = l)$ are constants. For example, in a label-balanced dataset, $p_\theta(y = l) \approx 1/L$.

We will first apply the attention weights visualization technique to the proposed AAN model. Here, the document representation can be directly expressed by the hidden states, i.e.,

$$v^{\text{agg}} = \sum_{t=1}^{T} \left(\sum_{k=1}^{K} \alpha_k^{\text{agg}} \alpha_{k,t}^{\text{abs}} \right) h_t,$$

where

$$\alpha_t^d = \sum_{k=1}^{K} \alpha_k^{\text{agg}} \alpha_{k,t}^{\text{abs}}, \tag{3}$$

gives the contribution of word w_t to the document representation. Therefore, we can interpret any single example via visualizing the combined weights α_t^d.

3.2.1 Corpus-Level Explanation. Different from case-level explanation (attention weights visualization), which focuses on per-sample features, corpus-level explanation aims at finding causal relationships between keywords captured by the attention mechanism and model predictions, which can provide robust explanation for the model. To achieve this goal, we learn distributions of keywords for different predicted labels on the training corpus based on attention weights.

Formally, for a given word $w \in \mathcal{V}$ and a label l predicted by a model θ[1], the importance of the word to the label can be estimated by the probability $p_\theta(w|y = l)$ across the training corpus C_{train} since the model is trained on it. Therefore, $p_\theta(w|y = l)$ can be expanded as follows:

$$p_\theta(w|y = l) = \sum_{\xi \in C^l_{\text{train}}} p_\theta(w, \xi|y = l), \tag{4}$$

where $C^l_{\text{train}} \subset C_{\text{train}}$ consists of documents with model predicted label l. For each document $\xi \in C^l_{\text{train}}$, probability $p_\theta(w, \xi|y = l)$ represents the importance of word w to label l, which can be defined using attention weights, i.e.,

$$p_\theta(w, \xi|y = l) := \frac{\sum_{t=1}^{T} \alpha_t^\xi \cdot \delta(w_t, w)}{\sum_{\xi' \in C_{\text{train}}} f_{\xi'}(w) + \gamma}, \tag{5}$$

where $f_{\xi'}(w_t)$ is frequency of w_t in document ξ' and γ is a smoothing factor. $\delta(w_t, w) = \begin{cases} 1 & \text{if } w_t = w \\ 0 & \text{otherwise} \end{cases}$ is a delta function. The denominator is applied to reduce noises from stop-words and punctuation. For the sake of simplicity, we will use $p_\theta(w, l, C)$ to denote $p_\theta(w_t|y = l)$, where C corresponds to the corpus Equation (4), and can be different from C_{train} in our applications. The denominator in Equation (5) is always determined by the training corpus.

With respect to the applications: (1) Since Equation (4) captures the importance of words to model predicted labels, we can use it as a criterion for finding their causal relationships. In experiments, we can collect top-ranked keywords for each label l for further analysis. (2) We can also use corpus-level explanation to measure the difference between two corpora (i.e., C_{test1} and C_{test2}). Formally, we can compare $\frac{|C_{\text{train}}|}{|C_{\text{test1}}|} \cdot p_\theta(w, l, C_{\text{test1}})$ with $\frac{|C_{\text{train}}|}{|C_{\text{test2}}|} \cdot p_\theta(w, l, C_{\text{test2}})$ across different words and class labels. The difference can be evaluated by Kullback–Leibler divergence [25]. In addition, we can get mutual keywords shared across different domains based on these distributions.

It should be noted that the corpus-level explanation discussed in this section can be applied to interpret different attention-based networks.

3.2.2 Concept-Based Explanation. The corpus-level explanation still suffers from the drawback that it cannot automatically obtain higher level concepts/clusters for those important keywords. To alleviate this problem, we propose concept-based explanation for our AAN model. In AAN, each abstraction-attention unit can capture one concept/cluster. Here, we will take distribution of concepts into consideration. Formally, we express $p_\theta(w_t|y = l)$ as follows:

$$p_\theta(w|y = l) = \sum_{k=1}^{K} p_\theta(w|c_k, y = l) p_\theta(c_k|y = l),$$

[1]Here, the label is the model's prediction, not the ground-truth label, because our goal is to explain the model.

where $p_\theta(w|c_k, y = l)$ captures the distribution of w across C_{train} for the kth concept and label l, while $p_\theta(c_k|y = l)$ captures the distribution of the concept c_k across C_{train} for label l. They can be computed using the following equations.

$$p_\theta(w|c_k, y = l) = \sum_{\xi \in C^l_{\text{train}}} p_\theta(w, \xi|c_k, y = l),$$
$$p_\theta(c_k|y = l) = \sum_{\xi \in C^l_{\text{train}}} p_\theta(c_k, \xi|y = l), \tag{6}$$

where we define

$$p_\theta(w, \xi|c_k, y = l) := \frac{\sum_{t=1}^{T} \alpha_{k,t}^{\text{abs},\xi} \cdot \delta(w_t, w)}{\sum_{\xi' \in C_{\text{train}}} f_{\xi'}(w) + \gamma}, \tag{7}$$

and

$$p_\theta(c_k, \xi|y = l) := \frac{\alpha_k^{\text{agg},\xi}}{|C_{\text{train}}|}, \tag{8}$$

where $\alpha_{k,t}^{\text{abs},\xi}$ represents $\alpha_{k,t}^{\text{abs}}$ for document ξ. Based on Equation (6), we are able to obtain scores (importance) and most relevant keywords for different concepts for a given label l.

3.2.3 Consistency Analysis. In corpus-level and concept-based explanations, we have obtained causal relationships between keywords and predictions, i.e., $p_\theta(w|y = l)$. However, we have not verified if these keywords are really important to predictions. To achieve this goal, we build an NBC [10] based on these causal relationships. Formally, for each testing document d, the probability of getting label l is approximated as follows:

$$p_\theta(y = l|d) = \frac{p_\theta(d|y = l)p_\theta(y = l)}{p(d)}$$
$$\propto p_\theta(d|y = l) = \prod_{t=1}^{T} p_\theta(w_t|y = l), \tag{9}$$

where $p_\theta(w_t|y = l)$ is obtained by Equation (4) or Equation (6) on the training corpus. We further approximate Equation (9) with

$$p_\theta(y = l|d) = \prod_{w \in d'} (p_\theta(w|y = l) + \lambda), \tag{10}$$

where $d' \subset d$ is an *attention-based bag-of-words representation* for document d. It consists of important keywords based on attention weights. λ is a smoothing factor. Here, we can conduct consistency analysis by comparing labels obtained by the model and NBC, which may also help estimate the uncertainty of a model [55].

4 EXPERIMENTS

4.1 Datasets

We conducted experiments on three publicly available datasets. Newsroom is used for news categorization, while IMDB and Beauty are used for sentiment analysis. The details of the three datasets are as follows: (1) **Newsroom** [15]: The original dataset, which consists of 1.3 million news articles, was proposed for text summarization. In our experiments, we first determined the category of each article based on the URL, and then, randomly sample 10,000 articles for each of the five categories, including business, entertainment, sports, health, and technology. (2) **IMDB** [31]: This dataset contains 50,000 movie reviews from the IMDB website with binary (positive or negative)

Table 1. Statistics of the Datasets Used

Dataset	#docs	Avg. Length	Scale
Newsroom	50,000	827	1–5
IMDB	50,000	292	1–2
Beauty	40,000	91	1–2

Table 2. Averaged Accuracy of Different Models on Newsroom, IMDB, and Beauty testing Sets

Model	Newsroom	IMDB	Beauty
CNN	90.18	88.56	88.42
LSTM-SAN	91.26	90.68	92.00
BERT-SAN	92.28	92.60	93.72
DistilBERT-SAN	**92.66**	92.52	92.82
RoBERTa-SAN	91.16	92.76	93.40
Longformer-SAN	92.04	**93.74**	**94.50**

labels. (3) **Beauty** [16]: This dataset contains product reviews in the beauty category from Amazon. We converted the original ratings (1–5) to binary (positive or negative) labels and sampled 20,000 reviews for each label. For all three datasets, we tokenized reviews using the BERT tokenizer [49] and randomly split them into train/development/test sets with a proportion of 8/1/1. Statistics of the datasets are summarized in Table 1.

4.2 Models and Implementation Details

We compare different classification models including several baselines, variants of our AAN model, and NBCs driven by a basic **self-attention network** (**SAN**) [38] and AAN.

- **CNN [22]:** This model extracts key features from a review by applying convolution and max-over-time pooling operations [7] over the shared word embedding layer.
- **LSTM-SAN**, **BERT-SAN**, **DistilBERT-SAN**, **RoBERTa-SAN**, **and Longformer-SAN:** All these models are based on the SAN framework. In LSTM-SAN, the encoder consists of a word embedding layer and a Bi-LSTM encoding layer, where embeddings are pre-loaded with 300-dimensional GloVe vectors [33] and fixed during training. BERT [49], DistilBERT [37], RoBERTa [28], and Longformer [3] leverage different pre-trained language models, which have 110 Million, 66 Million, 125 Million, and 125 Million parameters, respectively.
- **AAN + C(c) + Drop(r):** These are variants of AAN. C(c) and Drop(r) represent the number of concepts and dropout rate, respectively.

We implemented all deep learning models using PyTorch [32] and the best set of parameters are selected based on the development set. For CNN-based models, the filter sizes are chosen to be 3, 4, and 5, and the number of filters is set to 100 for each size. For LSTM-based models, the dimension of hidden states is set to 300 and the number of layers is 2. All parameters are trained with the ADAM optimizer [24] with a learning rate of 0.0002. Dropout with a rate of 0.1 is also applied in the classification layer. For all explanation tasks, we set the number of concepts to 10 and dropout-rate to 0.02. Our codes and datasets are available at https://github.com/tshi04/ACCE.

4.3 Performance Results

We use accuracy as the evaluation metric to measure the performance of different models. All quantitative results have been summarized in Tables 2 and 3, where we use bold font to highlight

Table 3. Averaged Accuracy of BERT and Longformer-Based AAN Models on Newsroom, IMDB, and Beauty Testing Sets

	Newsroom		IMDB		Beauty	
	BERT	Longformer	BERT	Longformer	BERT	Longformer
SAN Framework	92.28	92.04	92.60	93.74	93.72	94.50
AAN + C(10) + Drop(0.01)	92.54	91.72	92.22	92.96	93.38	93.42
AAN + C(10) + Drop(0.02)	92.14	91.64	92.14	92.86	93.58	93.75
AAN + C(10) + Drop(0.05)	92.14	91.60	91.82	92.66	93.05	93.80
AAN + C(10) + Drop(0.10)	92.30	91.48	91.50	92.12	93.25	93.60
AAN + C(20) + Drop(0.01)	92.02	91.98	91.64	92.78	93.70	93.48
AAN + C(20) + Drop(0.02)	92.44	91.84	91.80	93.04	93.55	93.88
AAN + C(20) + Drop(0.05)	92.54	91.86	91.92	93.14	93.68	93.42
AAN + C(20) + Drop(0.10)	92.52	91.98	92.10	92.96	93.72	93.88

Table 4. Case-Level Concept-Based Explanation

ID	**Score**	**Keywords**
8	0.180	com(0.27), boston(0.26), boston(0.16), boston(0.1), and m(0.02)
6	0.162	marketing(0.28), ad(0.06), ##fs(0.05), investors(0.03), and said(0.03)
1	0.148	campaign(0.14), firm(0.14), money(0.06), brand(0.04), and economist(0.03)
2	0.122	economist(0.2), said(0.16), professional(0.16), investors(0.08), and agency(0.06)
9	0.116	boston(0.89), boston(0.11)
7	0.108	bloomberg(0.16), cn(0.11), global(0.09), money(0.06), and cable(0.05)
5	0.103	-(0.96), -(0.03), and s(0.01)
4	0.047	investment(0.76), money(0.14), investment(0.06), investment(0.02), and investors(0.01)
3	0.016	,(0.64), -(0.36)
10	0.000	.(0.93), -(0.07)

Here, each ID is associated with a concept, i.e., abstraction-attention unit. Scores and weights (following each keyword) are calculated with Equations (7) and (8). "-" represents special characters.

the highest accuracy on testing sets in Table 2. Comparing LSTM-SAN with BERT, DistilBERT, RoBERTa, and Longformer, we first find that different pre-trained language model-based encoders are better than the conventional LSTM encoder with pre-trained word embeddings. In Table 3, we replace self-attention on top of pre-trained language models with the AAN. We observe that different AAN models do not significantly lower the classification accuracy, which indicates we can use AAN for the concept-based explanation task without losing the overall performance. Here, the strategy of aggregation-attention weights dropout is necessary when training AAN models. In Table 9, we show that AAN models without randomly dropping aggregation-attention weights attain poor interpretability in concept-based explanation

4.4 Heat-Maps and Case-Level Concept-Based Explanation

First, we investigate if AAN attends to relevant keywords when it is making predictions, which can be accomplished by visualizing attention weights (see Figure 2). This is a *Business* news article from Newsroom and we observe that the most relevant keyword that AAN detects is *boston*. Other

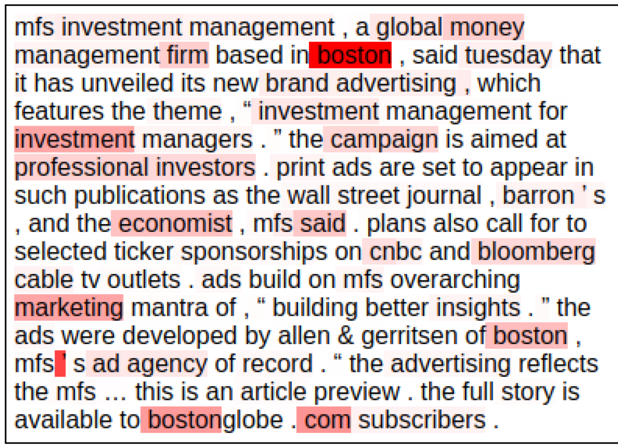


Fig. 2. Attention-weight visualization for an interpretable attention-based classification model.

important keywords include *investment*, *economist*, *marketing,* and *com*. Compared with Figure 2, our *case-level concept-based explanation* provides more informative results. From Table 4, we observe that AAN makes the prediction based on several different aspects, such as corporations (e.g., com), occupations (e.g., economist), terminology (e.g., marketing), and so on. Moreover, *boston* may be related with corporation (e.g., bostonglobe or gerritsen of boston) or city, thus, it appears in both concepts 8 (corporations) and 9 (locations).

4.5 Corpus-Level Explanation

Corpus-level explanation aims at finding the important keywords for the predictions. In Table 5, we show 20 most important keywords for each predicted label and we assume these keywords determine the predictions. In the last section, we will demonstrate this assumption by the consistency analysis. The scores of keywords have been shown in Figure 3.

In addition to causal relationships, we can also use these keywords to check if our model and datasets have bias or not. For example, *boston* and *massachusetts* play an important role in predicting business, which indicates the training set has bias. By checking our data, we find that many business news articles are from *The Boston Globe*. Another obvious bias example is that the numbers *8, 7,* and *9* are important keywords for IMDB sentiment analysis. This is because the original ratings scale from 1 to 10 and many reviews mention that "*rate this movie 8 out of 10*".

Moreover, from Figure 3(a) and (b), we find that for a randomly split corpus, distributions of keywords across training/development/test sets are similar to each other. This guarantees the model achieves outstanding performance on testing sets. If we apply a model trained on IMDB to Beauty (see Figure 3(c)), it can only leverage the cross-domain common keywords (e.g., *disappointed* and *loved*) to make predictions. However, we achieve 71% accuracy, which is much better than random predictions. In Table 5, we use bold font to highlight these common keywords.

4.6 Corpus-Level Concept-Based Explanation

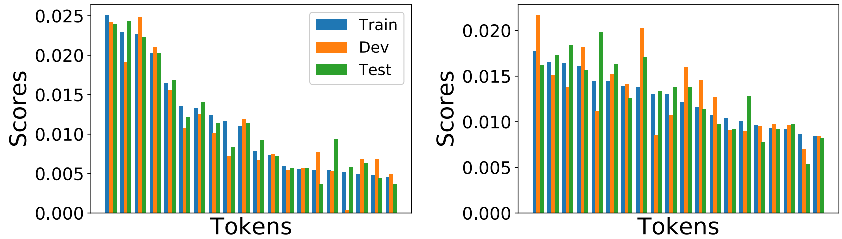
Corpus-level concept-based explanation further improves the corpus-level explanation by introducing clustering structures to keywords. In this section, we still use the AAN trained on Newsroom as an example for this task. Table 6 shows concepts and relevant keywords for AAN when it assigns an article to *Business*. Here, we observe that the first-tier salient concepts consist of concepts 8 (corporations) and 1 (business terminology in general). The second-tier concepts 7, 6, and

Table 5. This Table Shows 20 Most Important Keywords for Model Predictions on Different Training Sets

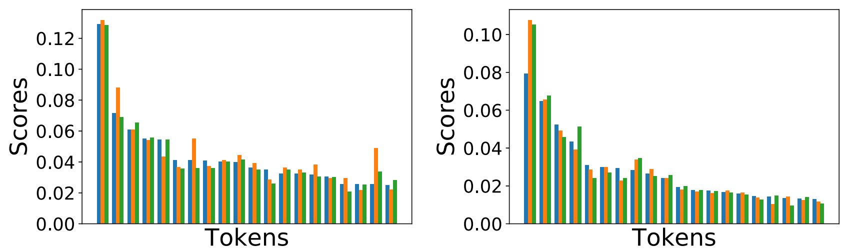
Dataset	Label	Keywords
IMDB	Negative	worst, awful, terrible, bad, **disappointed**, boring, **disappointing**, **waste**, **horrible**, sucks, fails, disappointment, lame, dull, poorly, poor, worse, mess, dreadful, and pointless
	Positive	8, 7, **excellent**, **loved**, 9, enjoyable, superb, enjoyed, **highly**, **wonderful**, entertaining, best, beautifully, good, **great**, brilliant, terrific, funny, hilarious, and fine
Beauty	Negative	disappointed, nothing, unfortunately, made, not, waste, disappointing, terrible, worst, horrible, makes, no, sadly, disappointment, t, awful, sad, bad, never, and started
	Positive	great, love, highly, amazing, pleased, perfect, works, best, happy, awesome, makes, recommend, excellent, wonderful, definitely, good, glad, well, fantastic, and very
Newsroom	Business	inc, corp, boston, massachusetts, economic, cambridge, financial, economy, banking, auto, automotive, startup, company, mr, finance, biotechnology, somerville, retailer, business, and airline
	Entertainment	singer, actress, actor, star, fox, comedian, hollywood, sunday, rapper, fashion, celebrity, contestant, filmmaker, bachelor, insider, porn, oscar, rocker, host, and monday
	Sports	quarterback, coach, basketball, baseball, soccer, nba, sports, striker, tennis, hockey, nfl, nhl, football, olympic, midfielder, golf, player, manager, outfielder, and nascar
	Health	dr, health, pediatric, obesity, cardiovascular, scientists, researcher, medicine, psychologist, diabetes, medical, psychiatry, aids, fitness, healthcare, autism, psychology, neuroscience, fox, and tobacco
	Technology	tech, cyber, electronics, wireless, lifestyle, silicon, gaming, culture, telecommunications, scientist, company, google, smartphone, technology, francisco, broadband, privacy, internet, and twitter

Keywords are ordered by their scores. For IMDB, we use bold to highlight the common keywords shared with Beauty.

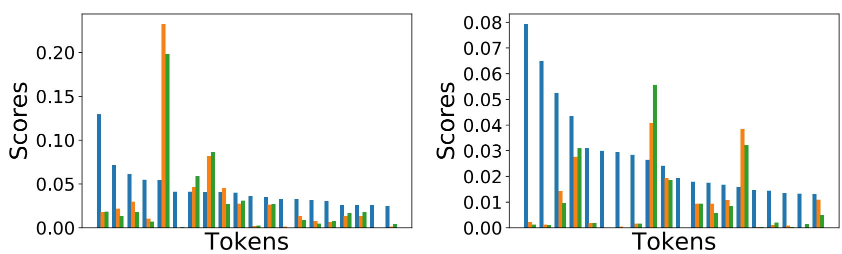
4 are related to economy, finance, mortgage, and banking, which are domain-specific terminology. They share many keywords. Concepts 9 and 2 are associated with locations and occupations, respectively, which receive relatively lower scores. Concepts 5, 3, and 10 are not quite meaningful. We have also shown results for Newsroom sports in Table 7, where we find that 1 (sports terminology) and 7 (leagues and teams) are the first-tier salient concepts. The second-tier salient concepts 6 and 4 are about games and campaigns. Concepts 7, 6, and 4 also share many keywords. Concepts 8 (corporations and channels), 2 (occupations and roles), and 9 (locations) are the third-tier salient concepts. Concepts 5, 3, and 10 are also meaningless. From these tables, we summarize some commonalities: (1) Domain-specific terminologies (i.e., concepts 1, 7, 6, and 4) play an important role in predictions. (2) Locations (i.e., concept 9) and Occupations/Roles (i.e., concept 2) are less important. (3) Meaningless concepts (i.e., concepts 5, 3, and 10), such as punctuation, have the least influence.



(a) Newsroom. Left: Business. Right: Sports



(b) IMDB. Left: Negative. Right: Positive



(c) IMDB-to-Beauty. Left: Negative. Right: Positive

Fig. 3. Distribution of keywords on training, development, and testing sets. Scores are calculated by $p_\theta(w, l, C)$. The orders of tokens are the same as those in Table 5.

4.7 Consistency Analysis

In this section, we leverage the method proposed in Section 3.2.3 to respectively build an NBC for BERT-SAN and BERT-AAN on the training set. Then, we apply them to the testing set to compare if NBC predictions and the model predictions are consistent with each other. We approximate the numerator of Equation (5) with five words (can repeat) with highest attention weights in each document. In Equation (4), γ is set to be 1,000. In Equation (10), we set $\lambda = 1.2$ for text categorization and $\lambda = 1.0$ for sentiment analysis. d' consists of five words with highest attention weights.

We use the accuracy (consistency score) between labels predicted by NBC and the original model to evaluate the consistency. Table 8 shows that around 85% of predictions are consistent. This demonstrates that keywords obtained by the corpus-level and concept-based explanation methods are important to predictions. They can be used to interpret attention-based models. Moreover, from CP and NCP scores, we observe a significantly higher probability that the model makes an incorrect prediction if it is inconsistent with NBC prediction. This finding suggests us to use consistency score as one criterion for *uncertainty estimation.*

4.8 Dropout of Aggregation-Attention Weights

For AAN, we apply dropout to aggregation-attention weights during training. In Table 9, we show an example without using the attention weight dropout mechanism. We observed that the weight

Table 6. Concept-Based Explanation (Business)

ID	Score	Keywords
8	0.173	inc, corp, massachusetts, boston, mr, ms, jr, ltd, mit, and q
1	0.168	economy, retailer, company, startup, ##maker, airline, chain, bank, utility, and billionaire
7	0.151	biotechnology, banking, tech, startup, pharmaceuticals, mortgage, financial, auto, commerce, and economic
6	0.124	economic, health, banking, finance, insurance, healthcare, economy, housing, safety, and commerce
4	0.107	financial, economic, banking, auto, automotive, securities, housing, finance, monetary, and biotechnology
9	0.086	boston, massachusetts, cambridge, washington, detroit, frankfurt, harvard, tokyo, providence, and paris
5	0.056	-, ##as, -, -, itunes, inc, corp, northeast, -, and llc
2	0.054	economist, executive, spokesman, analyst, economists, ##gist, ceo, director, analysts, and president
3	0.026	-, -, -,), ##tem, ##sp, the, =, t, and ob
10	0.000	-, comment,), insurance, search, ’, tesla, graphic, and guitarist,

Scores are calculated using Equation (6).

Table 7. Concept-Based Explanation (Sports)

ID	Score	Keywords
1	0.176	quarterback, player, striker, champion, pitcher, midfielder, outfielder, athlete, goaltender, and forward
7	0.165	nhl, mets, soccer, nets, yankees, nascar, mls, reuters, doping, and twitter
6	0.147	tennis, sports, soccer, golf, doping, hockey, athletic, athletics, injuries, and basketball
4	0.139	baseball, basketball, nba, nfl, sports, football, tennis, olympic, hockey, and golf
8	0.119	jr, ", n, j, fox, espn, nl, u, boston, and ca
2	0.100	coach, manager, commissioner, boss, gm, trainer, spokesman, umpire, coordinator, and referee
9	0.060	philadelphia, indianapolis, boston, tampa, louisville, buffalo, melbourne, manchester, baltimore, and atlanta
5	0.055	’, ’, ##as, -, ##a, ‘, sides, newcomers, chelsea, and jaguars
3	0.022	’, ’,), „ ##kus, ##gre, the, whole, lever, and ##wa
10	0.000	.,), finishes, bel, gymnastics, ’, ##ditional, becomes, tu, and united

Scores are calculated using Equation (6).

Table 8. Consistency between the Model and NBC

Model	Newsroom			IMDB			Beauty		
	CS	NCP	CP	CS	NCP	CP	CS	NCP	CP
BERT-SAN	83.96	21.59	4.72	86.02	17.17	5.81	85.45	16.30	4.56
BERT-AAN	84.36	20.20	5.57	85.46	21.18	5.05	84.72	16.04	4.51

CS represents consistency score, CP/NCP denotes percentage of incorrect predictions when NBC predictions are consistent/not consistent with model predictions.

Table 9. Concept-Based Explanation (Sports) for AAN without Applying Dropout to Attention Weights

CID	Weight	Keywords
1	0.8195	quarterback, athletic, olympic, basketball, athletics, qb, hockey, outfielder, and sports
7	0.0865	nascar, celtics, motorsports, nba, boston, augusta, nhl, tennis, leafs, and zurich
4	0.0370	mets, knicks, yankees, players, pitchers, lakers, hosts, coaches, forwards, and swimmers
3	0.0164	offensive, eli, bird, doping, nba, jay, rod, hurdle, afc, and peyton
2	0.0098	premier, american, mets, nl, field, yankee, national, aaron, nba, and olympic
10	0.0083	games, seasons, tries, defeats, baskets, players, season, contests, points, and throws
5	0.0015	dustin, antonio, rookie, dante, dale, dylan, lineman, ty, launch, and luther
8	0.0010	2016, 2014, college, tribune, card, press, s, -, this, and leadership
9	0.0004	men, -, grand, 9, s, usa, state, west, world, and major
6	0.0000	the, -, -, year, whole, vie, very, tr, too, and to

for concept 1 is much higher than the other concepts. In addition, keywords for each concept are not semantically coherent.

5 CONCLUSION

In this article, we proposed a general-purpose *corpus-level explanation* approach to interpret attention-based networks. It can capture causal relationships between keywords and model predictions via learning importance of keywords for predicted labels across the training corpus based on attention weights. Experimental results show that the keywords are semantically meaningful for predicted labels. We further propose a *concept-based explanation* method to identify important concepts for model predictions. This method is based on a novel AAN, which can automatically extract concepts, i.e., clusters of keywords, during the end-to-end training for the main task. Our experimental results also demonstrate that this method effectively captures semantically meaningful concepts/clusters. It also provides relative importance of each concept to model predictions. To verify our results, we also built an NBC based on an *attention-based bag-of-word document representation* technique and the causal relationships. Our consistency analysis results demonstrate that the discovered keywords are important to the predictions

REFERENCES

[1] Diego Antognini and Boi Faltings. 2021. Rationalization through concepts. In *Proceedings of the Findings of the Association for Computational Linguistics (ACL-IJCNLP'21)*. Association for Computational Linguistics, 761–775. DOI: 10.18653/v1/2021.findings-acl.68

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. arXiv:1409.0473. Retrieved from https://arxiv.org/abs/1409.0473.

[3] Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. arXiv:2004.05150. Retrieved from https://arxiv.org/abs/2004.05150.

[4] Francesco Bodria, Fosca Giannotti, Riccardo Guidotti, Francesca Naretto, Dino Pedreschi, and Salvatore Rinzivillo. 2021. Benchmarking and survey of explanation methods for black box models. arXiv:2102.13076. Retrieved from https://arxiv.org/abs/2102.13076.

[5] Diane Bouchacourt and Ludovic Denoyer. 2019. EDUCE: Explaining model decisions through unsupervised concepts extraction. arXiv:1905.11852. Retrieved from https://arxiv.org/abs/1905.11852.

[6] Zhi Chen, Yijie Bei, and Cynthia Rudin. 2020. Concept whitening for interpretable image recognition. *Nature Machine Intelligence* 2, 12 (01 Dec 2020), 772–782. DOI: https://doi.org/10.1038/s42256-020-00265-z

[7] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research* 12 (Nov. 2011), 2493–2537.

[8] Arun Das and Paul Rad. 2020. Opportunities and challenges in explainable artificial intelligence (XAI): A survey. arXiv:2006.11371. Retrieved from https://arxiv.org/abs/2006.11371.

[9] 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*. Association for Computational Linguistics, 4171–4186. DOI: https://doi.org/10.18653/v1/N19-1423

[10] Nir Friedman, Dan Geiger, and Moises Goldszmidt. 1997. Bayesian network classifiers. *Machine Learning* 29, 2 (01 Nov 1997), 131–163. DOI: https://doi.org/10.1023/A:1007465528199

[11] Reza Ghaeini, Xiaoli Fern, and Prasad Tadepalli. 2018. Interpreting recurrent and attention-based neural models: A case study on natural language inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 4952–4957. DOI: https://doi.org/10.18653/v1/D18-1537

[12] Amirata Ghorbani, Abubakar Abid, and James Zou. 2019. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 3681–3688.

[13] Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. 2019. Towards automatic concept-based explanations. In *Proceedings of the Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.), Vol. 32. Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper/2019/file/77d2afcb31f6493e350fca61764efb9a-Paper.pdf.

[14] Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. 2018. Explaining explanations: An overview of interpretability of machine learning. In *Proceedings of the 2018 IEEE 5th International Conference on Data Science and Advanced Analytics*. 80–89. DOI: https://doi.org/10.1109/DSAA.2018.00018

[15] Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 708–719. DOI: https://doi.org/10.18653/v1/N18-1065

[16] Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, 507–517. DOI: https://doi.org/10.1145/2872427.2883037

[17] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of the Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (Eds.), Vol. 28. Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper/2015/file/afdec7005cc9f14302cd0474fd0f3c96-Paper.pdf.

[18] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (Nov. 1997), 1735–1780. DOI: https://doi.org/10.1162/neco.1997.9.8.1735

[19] Ting Hua, Chang-Tien Lu, Jaegul Choo, and Chandan K Reddy. 2020. Probabilistic topic modeling for comparative analysis of document collections. *ACM Transactions on Knowledge Discovery from Data* 14, 2 (2020), 1–27.

[20] 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*. Association for Computational Linguistics, 3543–3556. DOI: https://doi.org/10.18653/v1/N19-1357

[21] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. 2018. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Jennifer Dy and Andreas Krause (Eds.). PMLR, 2668–2677. Retrieved from http://proceedings.mlr.press/v80/kim18d.html.

[22] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 1746–1751. DOI: https://doi.org/10.3115/v1/D14-1181

[23] Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T. Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. 2019. *The (Un)reliability of Saliency Methods*. Springer International Publishing, Cham, 267–280. DOI: https://doi.org/10.1007/978-3-030-28954-6_14

[24] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv:1412.6980. Retrieved from https://arxiv.org/abs/1412.6980.

[25] Solomon Kullback. 1997. *Information theory and statistics*. Courier Corporation.

[26] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. arXiv:1703.03130. Retrieved from https://arxiv.org/abs/1703.03130.

[27] Frederick Liu and Besim Avci. 2019. Incorporating priors with feature attribution on text classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 6274–6283. DOI: https://doi.org/10.18653/v1/P19-1631

[28] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized bert pretraining approach. arXiv:1907.11692. Retrieved from https://arxiv.org/abs/1907.11692.

[29] Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Curran Associates Inc., 4768–4777.

[30] Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 1412–1421. DOI: https://doi.org/10.18653/v1/D15-1166

[31] 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 *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 142–150. Retrieved from https://www.aclweb.org/anthology/P11-1015.

[32] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. In *Proceedings of the NeurIPS Autodiff Workshop*.

[33] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 1532–1543. DOI: https://doi.org/10.3115/v1/D14-1162

[34] Matthew E. 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*. Association for Computational Linguistics, 2227–2237. DOI: https://doi.org/10.18653/v1/N18-1202

[35] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 Task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation*. Association for Computational Linguistics, 19–30. DOI: https://doi.org/10.18653/v1/S16-1002

[36] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Model-agnostic interpretability of machine learning. arXiv:1606.05386. Retrieved from https://arxiv.org/abs/1606.05386.

[37] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv:1910.01108. Retrieved from https://arxiv.org/abs/1701.00133.

[38] Sofia Serrano and Noah A. Smith. 2019. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2931–2951. DOI: https://doi.org/10.18653/v1/P19-1282

[39] Tian Shi, Kyeongpil Kang, Jaegul Choo, and Chandan K. Reddy. 2018. Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations. In *Proceedings of the 2018 World Wide Web Conference*. International World Wide Web Conferences Steering Committee, 1105–1114. DOI: https://doi.org/10.1145/3178876.3186009

[40] Tian Shi, Liuqing Li, Ping Wang, and Chandan K Reddy. 2021. A simple and effective self-supervised contrastive learning framework for aspect detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 13815–13824.

[41] Tian Shi, Ping Wang, and Chandan K Reddy. 2019. LeafNATS: An open-source toolkit and live demo system for neural abstractive text summarization. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*. 66–71.

[42] Hendrik Strobelt, Sebastian Gehrmann, Michael Behrisch, Adam Perer, Hanspeter Pfister, and Alexander M. Rush. 2019. Seq2seq-Vis: A visual debugging tool for sequence-to-sequence models. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2019), 353–363. DOI: https://doi.org/10.1109/TVCG.2018.2865044

[43] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*. JMLR.org, 3319–3328.

[44] Erico Tjoa and Cuntai Guan. 2020. A survey on explainable artificial intelligence (XAI): Toward medical XAI. *IEEE Transactions on Neural Networks and Learning Systems* PP (October 2020). DOI: https://doi.org/10.1109/tnnls.2020.3027314

[45] Jesse Vig. 2019. A multiscale visualization of attention in the transformer model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Association for Computational Linguistics, 37–42. DOI: https://doi.org/10.18653/v1/P19-3007

[46] Oriol Vinyals, Ł ukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. 2015. Grammar as a foreign language. In *Proceedings of the Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (Eds.), Vol. 28. Curran Associates, Inc., 2773–2781. Retrieved from https://proceedings.neurips.cc/paper/2015/file/277281aada22045c03945dcb2ca6f2ec-Paper.pdf.

[47] Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 606–615. DOI: https://doi.org/10.18653/v1/D16-1058

[48] 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*. 11–20.

[49] 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*. Association for Computational Linguistics, Online, 38–45. DOI: https://doi.org/10.18653/v1/2020.emnlp-demos.6

[50] Weibin Wu, Yuxin Su, Xixian Chen, Shenglin Zhao, Irwin King, Michael R. Lyu, and Yu-Wing Tai. 2020. Towards global explanations of convolutional neural networks with concept attribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[51] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. *Advances in Neural Information Processing Systems* 32 (2019).

[52] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 1480–1489. DOI: https://doi.org/10.18653/v1/N16-1174

[53] Chih-Kuan Yeh, Been Kim, Sercan Arik, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. 2020. On completeness-aware concept-based explanations in deep neural networks. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 20554–20565. Retrieved from https://proceedings.neurips.cc/paper/2020/file/ecb287ff763c169694f682af52c1f309-Paper.pdf.

[54] Mohammad Nokhbeh Zaeem and Majid Komeili. 2021. Cause and effect: Concept-based explanation of neural networks. arXiv:2105.07033. Retrieved from https://arxiv.org/abs/2105.07033.

[55] Xuchao Zhang, Fanglan Chen, Chang-Tien Lu, and Naren Ramakrishnan. 2019. Mitigating uncertainty in document classification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 3126–3136. DOI: https://doi.org/10.18653/v1/N19-1316

[56] Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. 2018. Interpretable basis decomposition for visual explanation. In *Proceedings of the European Conference on Computer Vision*.

Received December 2020; revised May 2021; accepted July 2021