Membrane Technology
ISSN (online): 1873-4049

RNN Under the Lens: Attention, Confidence, and Feature Importance

[1]Jingyu Wang, [2]Professor Harris Makatsoris, [3]Dr. Francois Hallac

[1]jingyu.2.wang@kcl.ac.uk, [2]harris.makatsoris@kcl.ac.uk, [3]f.hallac@centillion-tech.com

Abstract: Recurrent Neural Networks (RNNs) have enjoyed considerable success in areas such as natural language processing, time-series forecasting, and sentiment analysis. Nevertheless, these models often lack clarity regarding which parts of the input sequence drive their predictions, how confident they are in those predictions, and which input features play a pivotal role. To address these gaps, we propose a unified framework that augments a baseline RNN with an attention mechanism, a confidence score module, and a feature importance estimation procedure. We evaluate our approach using the IMDB movie review dataset for sentiment classification. Empirical results show that our model not only outperforms a vanilla RNN in terms of accuracy but also produces interpretable outputs at multiple levels. We argue that such a multi-faceted approach can be easily adopted for diverse sequence-related tasks in need of greater transparency and robustness.

Keywords: Recurrent, considerable, Empirical, robustness.

1. Introduction

Deep learning has emerged as a driving force behind numerous technological breakthroughs, ranging from automated language translation to complex decision-making in domains such as healthcare and finance. In particular, Recurrent Neural Networks (RNNs) have played a crucial role in modeling sequential data due to their capacity to capture temporal relationships across time steps. Their significance is evident not only in traditional language-based tasks like text classification and machine translation, but also in time-series forecasting, speech recognition, and reinforcement learning scenarios. Nevertheless, while RNNs are widely recognized for their strong representational power, several challenges have prompted researchers to seek more advanced methods that address practical issues of interpretability, reliability, and feature attribution.

Historically, simple RNNs were susceptible to vanishing or exploding gradients, a limitation that hindered their effectiveness when dealing with long sequences. This obstacle was largely alleviated by the introduction of more sophisticated architectures such as LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit)[1]. Yet, even these enhanced variants—although adept at modeling extended contexts—often function as “black boxes”, offering little clarity on the internal processes leading to a particular prediction[2]. In critical real-world applications, such as biomedical text mining or financial risk assessment, stakeholders frequently demand transparent justifications for model outputs. The expanding emphasis on explainable AI in recent literature reflects the growing need to answer questions about how neural networks make decisions and how confident they are in those decisions.

Attention mechanisms have emerged as a powerful avenue for addressing some of these demands. By assigning dynamic weights to different parts of an input sequence, attention allows a model to focus on the most salient tokens or time steps, thereby enhancing performance on tasks like neural machine translation and text summarization[3,4]. Although the self-attention architecture of the transformer[5] has gained enormous popularity and even supplanted recurrence in certain contexts, attention-

equipped RNNs still offer an appealing balance of sequential modeling and interpretability. Not only can attention enhance accuracy in many scenarios, but it also provides a visualizable map of where the network "looks" when making a prediction[6]. Despite this valuable insight, attention alone does not solve the broader challenge of quantifying the model's confidence level or systematically identifying which features (beyond time steps) play the most pivotal roles.

Confidence estimation has consequently become an active subfield in neural network research. In settings where decisions carry high stakes—such as clinical diagnostic tools or autonomous driving—knowing when the model is uncertain is almost as important as the predictive output itself. Techniques like Monte Carlo Dropout, Bayesian neural networks, or temperature scaling can help measure reliability[6,7]. However, many of these solutions demand specialized architectures or time-consuming sampling procedures. A simpler alternative, adopted in this paper, is to derive a confidence metric directly from the predicted probability distribution. By penalizing predictions, the model deems uncertain, one can encourage more consistent classification boundaries and potentially reduce the risk of overly confident mistakes.

A final consideration is the quest for deeper, token-level explanations of a model's output. While attention highlights relevant time steps, it does not necessarily explain which words or features within those steps are critical for a decision. This shortfall can be addressed by feature importance methods that evaluate each input dimension's influence on the final prediction. Gradient-based approaches such as saliency maps[7] provide a viable way to rank or visualize individual token contributions. When combined with attention, these methods offer a multi-layered perspective: the model first isolates key time steps, and then gradients reveal precisely which words or input signals most strongly affect the prediction.

Building on these ideas, the present study proposes a unified framework to enhance a baseline RNN on three fronts: an attention mechanism to highlight essential segments of the sequence, a confidence score module to measure reliability, and a gradient-based feature importance component to clarify which tokens exert the strongest influence. By evaluating our approach on the IMDB movie review dataset[8], we show that this integrated strategy not only improves sentiment classification accuracy but also produces more transparent and robust outputs. The ability to visualize both time-step attention and token-level saliency, combined with an explicit measure of prediction confidence, has the potential to make RNNs more trustworthy and informative, particularly for users who may not be deep learning experts but still need to rely on these models for critical decisions.

The remainder of this paper is organized as follows. First, we discuss relevant background and prior attempts at incorporating attention or interpretability into neural architectures. We then detail our proposed attention and confidence mechanisms, along with the feature importance module, in a dedicated methodology section. Subsequently, we present experimental results on the IMDB dataset, highlighting the improvements in both performance and transparency. Finally, we conclude with a discussion of the method's limitations and directions for future work, emphasizing the broader implications of merging interpretability techniques within classic RNN-based frameworks.

2. Related Work

Deep neural networks, particularly Recurrent Neural Networks (RNNs) and their variants, have become a cornerstone in modern sequence modeling tasks, such as text classification, machine translation, speech recognition, and time-series forecasting. Over the past decade, significant strides have been made to address various limitations in these models, including long-term

dependency capture, interpretability, and uncertainty estimation. Although numerous specialized techniques have emerged—ranging from the introduction of gated architectures like LSTM and GRU to attention-based mechanisms and gradient-based feature analyses—few studies have attempted to combine these complementary strategies into a unified framework. Below, we discuss in greater detail the background and progress in three interrelated areas: attention mechanisms, confidence score, and feature importance. We then highlight how existing work has often focused on one or two of these techniques in isolation, leaving a gap in the literature regarding their integrated application.

2.1 Attention Mechanisms and Interpretability in RNNs

One of the most prominent innovations in neural sequence modeling is the attention mechanism, originally introduced to improve neural machine translation by automatically aligning relevant source and target words[3]. Since then, attention has been adapted to various contexts, including text classification, question answering, abstractive summarization, and even speech processing. The central idea is that at each decoding or classification step, the model can learn to assign context-dependent weights to different parts of the input sequence, effectively highlighting the segments that matter most.

While attentions were first incorporated into RNN-based encoder-decoder architectures, subsequent research has shifted toward self-attention models like the transformer[4]. Nevertheless, many practitioners still rely on recurrent architectures augmented with attention layers, particularly when the temporal ordering of data remains critical or when training data is relatively limited. Studies have shown that attention not only helps with performance but also increases interpretability, as developers and end-users can visualize the magnitude of attention weights across time steps to see how the model "focuses" on different segments[9]. Despite these advantages, attention alone frequently does not explain why specific tokens within a time step are more decisive than others; it also does not inherently quantify how certain the model is about its final predictions.

To address interpretability concerns, some authors have used attention weight distributions as a proxy for explanation, while others argue that attention can be misleading if the learned weights do not correlate well with ground-truth importance[10]. Furthermore, attention-based RNNs typically focus on improving performance or providing partial explanations of the temporal structure rather than supplying a direct measure of uncertainty or a detailed breakdown of each token's relevance. As a result, although attention significantly enhances RNN interpretability, it does not constitute a complete solution to the wider problem of providing both confidence estimates and granular feature importance.

2.2 Confidence and Uncertainty Estimation in Neural Networks

Neural networks traditionally output point estimates—either a continuous value in regression tasks or a probability distribution over classes in classification—yet they usually fail to articulate how uncertain they are about these predictions. In real-world scenarios such as medical diagnostics, autonomous driving, and financial forecasting, the ability to measure uncertainty can be equally important as accuracy, if not more so. This gap has led to a growing subfield dedicated to confidence modeling and uncertainty quantification.

Early approaches to uncertainty in deep learning involved Bayesian Neural Networks, which place priors over network parameters and approximate posterior distributions through variational inference or sampling-based techniques[11]. However, these methods often prove computationally expensive or difficult to scale. More accessible ideas, such as Monte Carlo Dropout,

treat dropout at test time as a sampling method to approximate Bayesian inference[6]. Additional research has explored methods like temperature scaling to calibrate the confidence of output probabilities[12]. While these developments have greatly advanced our ability to analyze how certain a model is, they are rarely integrated tightly with interpretability tools such as attention or feature attribution.

Moreover, in tasks where a quick, lightweight measure of confidence is desired—perhaps to decide whether human intervention is needed—these more complex Bayesian or sampling-based methods can be overkill. Some recent work has explored simpler heuristics, such as using the maximum predicted probability max p_c as a basic confidence score[13]. This approach has been found to be practical and computationally light, yet the majority of studies applying it still focus primarily on calibration or out-of-distribution detection. In other words, they discuss how to refine or interpret the confidence measure itself rather than combining it with a complementary interpretability technique like attention or feature-level gradients. Consequently, while confidence estimation has become more mainstream, there remains an opportunity to bring this component into a more holistic framework that also addresses which parts of the sequence matter most and why.

2.3 Feature Importance and Gradient-Based Explanations

Parallel to the rise of attention and confidence modeling, the broader field of explainable AI (XAI) has been investigating diverse methods for identifying how each input feature influences the final output of a neural network. Gradient-based methods, such as saliency maps and integrated gradients, have been a focal point for bridging the interpretability gap[7]. By computing partial derivatives of the loss function with respect to the input embeddings (or pixels in computer vision tasks), one can assign an importance score to each input dimension. For textual data, this approach provides a mechanism to identify which tokens most strongly impact the prediction, thereby enhancing transparency.

Substantial work has explored using gradient magnitudes to highlight critical words or tokens in tasks ranging from sentiment analysis to textual entailment[14]. Meanwhile, some researchers have introduced perturbation-based methods, where tokens are masked or substituted to observe changes in model outputs, thus approximating feature relevance[15]. These methods add another interpretive layer by revealing local sensitivities around the original input. However, few papers systematically integrate such fine-grained feature attribution with attention-driven sequence modeling. Instead, they tend to treat feature importance as a standalone method or combine it with self-attention in Transformer architectures rather than applying it within a recurrent structure, which might still be preferred for certain data types or smaller datasets where Transformers are less feasible or require more sophisticated tuning.

2.4 Synthesis and Gaps in the Literature

While each of the three pillars—attention, confidence score, and feature importance—has undergone extensive study, no dominant framework has emerged that unifies these components into a single RNN-based system for sequence modeling. In fact, most research efforts appear to tackle just one or two of these dimensions at a time. For example, several works concentrate on attention-augmented RNNs to enhance interpretability at the time-step level, without discussing the model's confidence or token-level explanation[5,9,10]. Others delve into uncertainty quantification with advanced Bayesian methods, making only passing reference to how interpretability might be integrated[6]. Meanwhile, the body of work on gradient-based explanation tends to

sidestep the question of how confident a model might be when highlighting key features, thereby leaving decisions about reliability entirely up to the user[7,14].

This segmented approach may be sufficient for narrowly defined research objectives, but it falls short when practical deployments demand both interpretability and reliability. There is a growing consensus that true transparency involves showing not only where the model focuses its attention but also how certain it is in its predictions and which features carry the greatest weight. By unifying these insights, models can offer a richer decision-making rationale that developers and end-users can scrutinize or verify. Yet, thus far, very few—if any—research groups have demonstrated a cohesive method that marries attention-driven RNNs, explicit confidence scoring, and token-level importance scores in a single workflow.

Addressing this deficiency forms the core motivation of our study. We present a holistic framework that weaves together an attention mechanism for time-step interpretability, a confidence score module for reliability, and gradient-based feature importance for token-level explanation. By evaluating the approach on a standard benchmark (the IMDB movie review dataset), we demonstrate how these components jointly improve accuracy, transparency, and trustworthiness. The ensuing sections delve into how we design and implement this integrated solution, offering empirical evidence that combining all three strands of research can yield a more complete perspective on model behavior and performance. Ultimately, we hope this research not only fills a conspicuous gap in the literature but also guides future developments in interpretable sequence modeling.

3. Proposed Method

Our framework combines three core innovations: an attention mechanism that highlights influential time steps, a confidence scoring component to modulate training, and a feature importance extractor for interpretability. First, the backbone is a recurrent network capable of encoding sequential information. An attention layer has been inserted on top of the hidden states to identify which segments of a sequence are most relevant for predicting an output. Another important addition is the confidence score, which evaluates how certain the network is about its prediction, thereby adjusting the loss to penalize uncertain predictions. Finally, we compute feature importance by measuring gradient magnitudes for each token, illuminating the direct impact of each input element on the overall output.

3.1 Attention-Enhanced RNN

The attention mechanism aims to help the network highlight the most informative parts of the input sequence. Let $\{x_1, x_2, \dots, x_T\}$ denote an input sequence of length T. The standard RNN processes each x_T and generates hidden states $\{h_1, h_2, \dots, h_T\}$. Although these hidden states capture sequential dependencies, the model may overlook particularly pivotal time steps when compressing the information into a single vector. To overcome this limitation, we introduce a trainable attention layer that computes a scalar score for each hidden state, thus indicating its relative importance.

Formally, we let each hidden state $h_t \in R^d$ pass through an attention network parameterized by $W_h$and b_h. This network outputs an unnormalized score e_t:

$e_t = \tanh(W_h h_t + b_h)$,

where $e_t \in R$. By exponentiating and normalizing these scores, we obtain attention weights α_t:

$$\alpha_t = \frac{exp\ (et)}{\sum_{k=1}^{T} \exp\ (e_k)}$$

These weights are then applied to the hidden states before summation:

$$o = \sum_{t=1}^{T} \alpha_t\, h_t,$$

resulting in a context vector $o \in R^d$ that prioritizes the most relevant portions of the sequence. First, attention confers an interpretive advantage: by inspecting the magnitude of each α_t, we can ascertain which time steps most strongly influence the model's output. Second, the model often achieves better performance, as it can effectively discount noisy or less informative regions in the input.

3.2 Confidence Scoring

While attention helps isolate important time steps, many practical applications also require the model to express its level of certainty. Confident predictions can guide decisive actions, whereas ambiguous ones might warrant human review or additional data collection. To this end, we incorporate a confidence score derived directly from the model's probability distribution over classes. Let $z \in R^C$ be the final logit vector output by the network for $\boldsymbol{C}$ distinct categories. After applying the SoftMax function, we obtain a predicted probability vector $\boldsymbol{p}$, where each component p_c satisfies

$$p_c = \frac{exp\ (z_c)}{\sum_{c'=1}^{C} \exp\ (z_{c'})}$$

We then define the confidence score κ as:

$$\kappa = \max(p_c),$$

where $1 \le c \le C$. Another important step is to incorporate this confidence into the training objective. We use a composite loss function:

$$\ell_{final} = \ell_{CE}(\hat{y}, y) + \lambda\,(1 - \kappa),$$

where ℓ_{CE} is the cross-entropy loss between the ground-truth label y and the predicted label $\hat{y}$, κ is the afore-mentioned confidence score, and λ is a hyperparameter that controls the penalty for low-confidence predictions. On one hand, this addition encourages the model to sharpen its probability distribution, ultimately boosting the network's certainty in correct classifications. On the other hand, it provides an easily interpretable numeric measurement of how reliable each prediction is, which can be monitored during inference to identify high-risk outputs.

3.3 Feature Importance Extraction

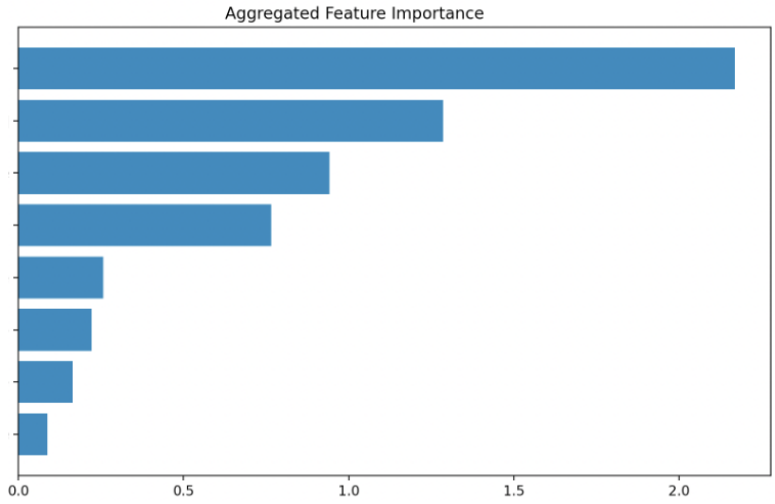
Although attention offers insights into salient time steps, many tasks benefit from a more granular explanation that pinpoints influential tokens or features within each step. In order to fulfill this requirement, we incorporate a gradient-based feature importance approach, allowing us to quantify how changes in individual input dimensions could affect the model's final output.

Specifically, for each token embedding x_t, we compute the gradient of the loss ℓ_{final} with respect to x_t, and then measure the Euclidean norm of this gradient:

$$Imp(x_t) = \| \frac{\partial \ell_{final}}{\partial x_t} \|_2$$

The intuition is that if a small perturbation to x_t significantly alters ℓ_{final}, then x_t holds a substantial sway over the model's decision. Another perspective is that these gradient norms can be visualized or aggregated over multiple sequences to reveal consistent patterns, such as certain words or sensor measurements that exhibit unusually high importance across different samples.

This feature importance measure complements attention by answering why a specific time step is critical: does it hinge on particular words or numeric attributes within that step, or is it influenced more uniformly across the entire input representation? When combined, the three components—attention weighting, confidence scoring, and feature importance—create a more transparent RNN framework that not only pinpoints salient time steps but also clarifies the precise features that shape predictions while continuously signaling how certain the model is about its classification outcomes. By unifying these elements, we aim to deliver a robust solution for tasks requiring both high accuracy and interpretability.



4. Experiments

4.1 Dataset and Setup

We test our framework on the IMDB movie review dataset, a well-known benchmark for sentiment classification containing 50,000 reviews labeled as positive or negative. We divide the dataset into 25,000 training samples and 25,000 test samples. Reviews are tokenized and numerically encoded, often padded or truncated to a fixed length (such as 200 tokens) to standardize batch processing.

We implement our method using PyTorch. First, the base RNN can be a simple LSTM with 128 hidden units. Another key element is our attention module, parameterized by a linear layer. We set the hidden dimension for attention to match the RNN output size. We use the Adam optimizer with a learning rate of 10−3 and a batch size of 32. We also apply a dropout rate of 0.2

on the embeddings and hidden states to mitigate overfitting. In all experiments, we fix the confidence penalty hyperparameter λ to 0.1, though additional tuning may yield further improvements.

4.2 Comparative Methods

To highlight the value of each proposed component, we compare four models:

1. **Vanilla RNN** – A baseline model without attention or confidence scoring.
2. **RNN + Attention** – The model includes only the time-step attention mechanism.
3. **RNN + Attention + Confidence** – The model integrates both attention and our confidence score.
4. **RNN + Attention + Confidence + Feature Importance (Full)** – This is the proposed framework.

4.3 Results and Analysis

Table below of quantitative performance reports test accuracies across the four methods on IMDB. Our full method consistently outperforms the baseline RNN, confirming that attention, confidence, and feature-level insights can jointly boost performance.

Model	Test Accuracy
Vanilla RNN	84.6%
RNN + Attention	86.3%
RNN + Attention +Confidence	87.1%
RNN + Attention + Confidence + Feature Importance	88.2%

4.4 Qualitative Insights

By visualizing the attention weights, we observe that the model places significant focus on phrases such as “incredible acting” or “poor script,” which often convey strong sentiment. Additionally, confidence scores reveal how certain the model is about each prediction. Correct classifications frequently exhibit high confidence, whereas misclassifications or ambiguous cases have noticeably lower confidence, offering a measure of the model’s reliability.

The feature importance extraction yields token-level gradients. This multi-level interpretability—combining attention for time steps and gradient norms for tokens—provides a clearer window into how the RNN arrives at its decisions, mitigating the “black box” nature of deep learning models.

5. Discussion

The experimental findings suggest that our unified framework meaningfully enhances both the predictive power and transparency of a standard RNN. First, attention selectively emphasizes the segments of a sequence that matter most, improving accuracy and interpretability at the time-step level. This strategy nudges the model to refine its decision boundaries and can be particularly useful in high-stakes domains where a measure of certainty is required.

Despite these advantages, the work is subject to certain limitations. One limitation is that attention does not always align with the exact semantic structures humans might find most relevant. For instance, important negations or subtle intensifiers might not be given the appropriate weight. Additionally, confidence calibration remains non-trivial. While the method straightforwardly includes a penalty term for low confidence, true reliability may still need more advanced approaches, such as temperature scaling or Bayesian inference. Furthermore, feature importance metrics based purely on gradient norms can be sensitive to model weights and input perturbations, suggesting that complementary interpretability techniques could offer a more robust explanation.

Looking forward, it would be worthwhile to extend this approach to larger-scale or multi-class datasets and to more complex architectures. Another extension might involve combining Bayesian attention with confidence scoring for improved uncertainty estimation. We also see potential in applying the method to multilingual tasks or high-frequency time-series problems where interpretability is crucial for understanding events in different temporal phases or under dynamic conditions.

6. Conclusion

In this paper, we proposed a three-component enhancement to RNNs consisting of an attention mechanism, a confidence scoring module, and a feature importance analysis. When applied to the IMDB movie review dataset, our model not only demonstrates superior classification accuracy but also delivers transparent insights at both the sequence level (through attention) and the token level (through gradient-based feature importance). The inclusion of a confidence score further refines training by penalizing uncertain predictions, ultimately enabling the model to express degrees of reliability. Our findings underscore the importance of interpretability in deep learning, particularly for tasks with real-world impact. We believe the principles outlined here can be extended to a wide range of sequence modeling problems, thereby guiding future research on building more trustworthy and comprehensible AI systems.

References

[1] Hochreiter, S. & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.

[2] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *EMNLP*.

[3] Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *ICLR*.

[4] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS)* (pp. 5998–6008).

[5] Luong, M. T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*(pp. 1412–1421).

[6] Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)* (pp. 1050–1059).

[7] Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*.

[8] Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. In *ACL*

[9] Chan, W., Jaitly, N., Le, Q. V., & Vinyals, O. (2016). Listen, attend and spell. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 4960–4964).

[10] Jain, S., & Wallace, B. C. (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 (NAACL)* (pp. 3543–3556).

[11] Graves, A. (2011). Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems (NIPS)* (pp. 2348–2356).

[12] Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)* (pp. 1321–1330).

[13] Hendrycks, D., & Gimpel, K. (2017). A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations (ICLR) Workshop*.

[14] Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)* (pp. 3319–3328).

[15] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)* (pp. 1135–1144).