
An Empirical Study of Explainable AI Techniques on Deep Learning Models For Time Series Tasks

Udo Schlegel

University of Konstanz
u.schlegel@uni-konstanz.de

Daniela Oelke

Offenburg University of Applied Sciences
daniela.oelke@hs-offenburg.de

Daniel A. Keim

University of Konstanz
keim@uni.kn

Mennatallah El-Assady

University of Konstanz
menna.el-assady@uni.kn

Abstract

Decision explanations of machine learning black-box models are often generated by applying Explainable AI (XAI) techniques. However, many proposed XAI methods produce unverified outputs. Evaluation and verification are usually achieved with a visual interpretation by humans on individual images or text. In this preregistration, we propose an empirical study and benchmark framework to apply attribution methods for neural networks developed for images and text data on time series. We present a methodology to automatically evaluate and rank attribution techniques on time series using perturbation methods to identify reliable approaches.

1 Introduction

Explainable AI (XAI) establishes a research field to bridge the gap between state-of-the-art deep learning and production-ready models in the industrial sector to explain and understand outputs. In research, deep learning achieves state-of-the-art performance in autonomous driving [16], speech assistance [6], and natural language processing [35] improving previous results by a margin. However, deep neural network models' black-box nature is often not suitable for a production-ready model [12]. For instance, the EU General Data Protection Regulation [10] forces companies to justify their employed algorithms' decisions. Especially, critical characteristics (fairness, privacy, reliability, trust [8]) for machine learning systems led to the dismissal of state-of-the-art deep learning models and the employment of interpretable models, e.g., in critical environments like health [25]. However, agencies like DARPA promoted research grants around explainable AI projects [12] to support the research into understanding deep learning models to facilitate their deployment. XAI techniques such as local interpretable model-agnostic explanations (LIME) [24] are developed to enable the interpretation of the model's decisions [11]. These XAI techniques promise to bridge the gap between black-box decision and comprehensible explanations of such models.

Most XAI techniques aiding in debugging and understanding a model's decisions are generally developed for domains such as images or text [11]. However, other types of data emerge as fast as images or text, e.g., due to an increasing amount of installed sensors. Such sensors produce massive amounts of time series signals and support, for instance, early detection of failures through failure prediction models [21]. An increasing amount of black-box models are deployed to tackle tasks with time series data. However, only a few works include temporal data in their XAI technique analysis, e.g., LIME [24], to create comprehensible explanations for such complex models. Such a development urges to evaluate already established XAI techniques applied on time series to identify applicable methods [26]. As time series analysis is often domain and tasks specific, a rigorous study of XAI techniques applied on such broad on time series can identify strengths and weaknesses to analyze baseline functionality for a time series XAI method [26].

In many cases, evaluating XAI techniques on a large scale is a tedious task, as a human evaluation is typical for most domains, which is relatively slow and sample-based [22]. However, a quantitative evaluation is necessary to identify reliable methods and verify their explanations on the number of data sensors produce. Thus, automatic quantitative analysis is needed to support human qualitative evaluations to present only critical samples for further investigation of the model and the XAI technique. Also, raw time series and sensor signals are often difficult for domain experts to analyze without converting them. Thus verification of explanations based on raw time series is not trivial even with human evaluations. In many cases, Fourier transformations help analysts to understand data characteristics. Still, with complex models working on the raw data, such a transformation often comprises loss of information, biases, or even faulty conclusions. As a result, automated evaluation and verification of XAI techniques on complex raw time series models are required to support humans in understanding, debugging, and improving their models.

In this preregistration, we propose an empirical study and benchmark framework to test and evaluate XAI techniques, focusing on attribution methods applied to time series data and models, thereby extending our previous work [26]. We present a methodology to automatically evaluate and rank XAI techniques on time series tasks using perturbation methods. Our empirical analysis tackles the following questions about the verification of XAI techniques on a selected set of network architectures.

Transferring attribution methods to time series:

- For each of the considered XAI attribution methods, can we transfer them to time series?
- What are the tweaks and changes needed to apply each method?
- Are some methods better suited for particular models and tasks?

Evaluating the applicability of transferred methods to time series tasks:

- Which metrics and measures of validation are required for a systematic evaluation of each method on the given tasks?
- How strong does an XAI attribution reflect the model’s predictions?
- Can we rank and identify strengths as well as weaknesses of attributions on time series for given tasks?

Benchmarking attribution methods on time series tasks:

- Given a concrete model architecture, dataset, and task; how sensitive are XAI methods?
- What is the overall strongest performing XAI method for each task?
- Can we reproduce the results of Schlegel et al. [26] showing SHAP as best performing?

We want to investigate how much the XAI technique outputs differ from each other and how much their proposed domain differs from time series tasks through these questions to investigate a detailed analysis of strengths and weaknesses. Such an analysis leads to insights into the model’s behaviors towards the time dependencies these complex models learn during their training.

2 Related Work

An increasing amount of XAI techniques are developed to support, reason, and explain the rising amount of AI model decisions [32]. Especially, methods like LIME [24] and Integrated Gradients [28] slowly begin to move into industrial contexts to facilitate the usage of production-ready deep learning models. Nevertheless, most XAI techniques are only evaluated on their data and task at hand to show their applicability [34, 28]. In some cases, new XAI techniques are compared to some others. However, these comparisons are relatively sparse and only against a few similar techniques [17]. There is often a lack in comparison to more than just similar methods and techniques. Further, general evaluation is often only done using, e.g., the pointing game for object detection [37] which checks for every object if the maximum attribution of the XAI technique is contained in the bounding box.

Nevertheless, Hooker et al. [15] propose the benchmark framework ROAR (RemOve And Retrain) on images to investigate XAI techniques and their feature attributions. ROAR first uses the attributions to perturb relevant pixels to uninformative values to create a new data set. Next, it tests the model’s accuracy on the created data and retrains the model on it. After the retraining, ROAR again creates attributions and perturbs the relevant pixels to get a new data set to test the accuracy. Hooker et al. [15] find that most of their applied techniques produce unconvincing attributions on the retrained model which do not change the test accuracy. The benchmark has a few weaknesses, for instance, complex models, which need a lot of time for their training, are hard to analyze as the whole benchmark

run needs to train the model twice which could lead to days or weeks for results. Also, ROARs results show that after retraining the model on the changed data, the accuracy does not get worse by perturbing the data by applying the XAI technique again. Such a result show that these techniques not always show the correct explanations, however, these results could also show that the XAI technique is not suitable for the model architecture.

Schlegel et al. [26] propose two verification methods to apply perturbation methods onto time series data and present preliminary results applying XAI techniques on time series. They propose time point perturbations, which change relevant time points to zero or the original value’s inverse. To test the time dependencies learned by a classifier, they present time interval perturbations in which they reorder an interval around relevant time points or set the whole interval to the its mean. Through these verification methods, Schlegel et al. [26] demonstrate in a preliminary evaluation the application of Saliency Maps [30], LRP [3], DeepLIFT [28], LIME [24], and SHAP [20] on time series. However, they neglect other XAI techniques to investigate the verification methods they propose and show the results only for one gradient-based approach. Further, an analysis between score-based and gradient-based techniques could lead to strengths and weaknesses for specific models.

Overall, there is a lack of XAI technique evaluations as the research is still only applied in some areas like computer vision [15, 17]. Thus, in this work, we tackle the time series domain to apply and analyze other fields’ XAI techniques to evaluate suitability and explanations.

3 Methodology

To assess the quality of an explanation of an XAI technique, we measure the difference of a quality metric applied on the models output for test data and on the output of an adaption of the test data based on the explanation.

3.1 Perturbation on Time Series

One of the most used techniques to test impacts on the classification of models are perturbations of the data [26, 36]. A common technique in computer vision to either evaluate a model on the importance of pixels in an image is to set these to black [36] or test an XAI technique by adapting the relevant pixels to an uninformative value (e.g., mean) [15]. Schlegel et al. [26] proposed two verification methods to use similar perturbation techniques on time series as altering a time point to zero often leads to biased data, which can help to classify. Such perturbations at single time points do not test if a classifier learns just single time points or if time dependencies are learned to predict unseen data. Thus, Schlegel et al. [26] propose a verification method on time intervals around relevant time points in which they reorder the interval or set the interval to the mean of it.

In our experiment, we adapt the proposed verification methods of Schlegel et al. [26] and add some tweaks to extend these to seven methods. For our time point perturbation, we take the relevant time points and set them to zero, to their inverse, and to the mean of the time series sample. The used perturbation takes a time series $t = (t_0, t_1, t_2, \dots, t_n)$ and changes relevant time points to, e.g., zero resulting in the changed time series $t^c = (t_0, 0, t_2, \dots, 0, t_n)$ for relevant time points $i = 1, n - 1$. Our time interval perturbation sets an interval around a relevant time point to zero, to the mean of the interval, to the inverse of every time point interval, and to a reordered version like Schlegel et al. [26]. These interval perturbations differ from the point perturbation by changing a whole intervals in a time series $t = (t_0, t_1, t_2, \dots, t_n)$ to, for instance, the mean of the interval resulting in $t = (t_0, \mu_2, \mu_2, \mu_2, \dots, t_n)$ with a relevant time point at $i = 2$ and an interval range of 1. Through these types of perturbations, we achieve a divers range of testing setups with various possible hypothesis. For instance, we can analyze models and XAI techniques on key features, e.g., models focusing on time intervals rather than points or techniques showing only time points as relevant.

3.2 Evaluation Methodology

After defining our verification methods, our evaluation methodology consists in the first step of the selection process for a task, a dataset, and a model (e.g., task: forecasting, data: weather, and model: RNN with LSTM [14] layers). Our evaluation pipeline is composed of three stages (model training, model explanation, and explanation evaluation) to calculate a score for every XAI technique we consider for the model architecture.

Model training – In the first step, we train our selected architecture on the selected data with the assigned task with a chosen cost function, e.g., RMSE. The model is initialized with a static seed and is trained for a previously chosen amount of epochs, neglecting early stopping to have a reproducible outcome. If the model’s training time does not exceed a specific limit, the process is changed to a random seed and ten trained models. In this case, the score is calculated by the average of the ten models. Finally, we use the quality metric to evaluate the XAI technique on a test set to get a baseline. Such a baseline can consist, for instance, of the accuracy or the RMSE depending on the task.

Model explanation – After training the model, we apply our XAI techniques on every sample of the test data and calculate the attributions. We normalize the attributions on single samples to identify the most relevant time points. Next, we select relevant time points for verification of our test data in three ways. First, we sort the resulting attributions of every sample and take the top k attributions. The sorting helps us to get the positions of the most relevant k time points. The parameter k varies with the input length and is relatively chosen to be five percent of the length with $len * 0.05$. Second, we calculate a threshold with $max - (max - mean) * 0.1$ of the attributions and take all time points with an attribution above as relevant. Such a threshold leads to the 95 percentile if the attributions reflect a normal distribution; else, it leads to a more robust selection as it is more dynamic than the top k. Third, we take a fixed threshold (e.g., 0.8) and select all time points with a higher attribution value than the threshold as relevant time points. After the selection of points, we perturb the selected time points at the resulting attribution positions for all three processes according to our verification methods. These three perturbations lead to $3 * v$ new test sets t^c with v being the number of verification methods applied and c an identifier for the change. As a last, we create again $3 * v$ new test sets t_r^c by selecting time points randomly as relevant based on the number of relevant points by the previous selections as well as ten more and ten less percent. Such three random baselines enable better comparison options.

Explanation evaluation – In our last step, we take our newly $12 * v$ created test sets t^c as well as t_r^c and our model to predict the quality metric to compare the results to the baseline. The assumption $qm(t) \geq qm(t_r^c) > qm(t^c)$, with the quality measure qm , the test set t , the randomly changed points t_r^c , and the XAI relevant time points t^c , holds, if the XAI technique captures relevant information the model learns to distinguish samples.

4 Experimental Protocol

Our current experimental setting involves the usage of 15 XAI techniques, six DNN models, seven verification techniques, and three time series tasks with various data sets. As such a setting, our design space of techniques and algorithms consists of at least $15 * 6 * 7 * 3$ possible variations. However, as the setup will also be provided as an extendable benchmark framework, more XAI techniques, models, data sets, and verification methods can be added later on to test further parameters.

XAI Techniques – The XAI techniques will involve gradient, score, and surrogate based attribution methods, namely LRP [3], LIME [24], SHAP [20], IG [34], Saliency [30], Occlusion [36], DeepLIFT [28], Input*Grad [29], PatternAttribution [36], Shapely Sampling [4], GradCAM [27], Guided Backprop [33], NoiseTunnel [1], DeepTaylor [23], and SmoothGrad [31]. We will use default parameters for all XAI techniques or will use parameters, as mentioned in the original papers. For our benchmark framework, we will also provide functions to evaluate and test different parameters, e.g., changing the kernel size for LIME or the window size for Occlusion.

DNN Models – Our models will consist of various deep neural networks with recurrent neural networks (LSTM [14], GRU [5]), convolutional neural networks (Conv1D, Conv2D, Conv3D), transformers for time series [19, 35], and a deep neural network with residual connections [13]. We will initialize our models with uniform distribution of weights and biases near zero. The models will be trained using Adam [18] as a optimizer. Our benchmark framework will also enable to change the models to predefined architectures with pre-trained parameters.

Verification Techniques – As a base for the verification, we include the methods presented in the previous section, namely point perturbations (zero, inverse, mean) and interval perturbations (swap, mean, zero, inverse) with accuracy (1) and root mean squared error (2) as deployed quality measures.

$$ACC = \frac{\text{correct predictions}}{\text{amount of samples}} \quad (1) \quad RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2)$$

Accuracy has a problem with unbalanced data sets as a model could only learn to predict the majority class and the accuracy would still be good. However, in our case, even if it learns to predict the majority class every time, we can argue that a working attribution should find relevant time points. Even if our classifier predicts the same for everything, then every XAI technique (even the random change) should have zero impact on the classification, which indicates a flawed model.

Time Series Tasks – We will focus on three major time series tasks for our experiment setting, namely classification, forecasting, and regression. As there is often ambiguity in these terms, we define them as follows: A classification model labels segmented non-overlapping time series t with a class c . A forecasting model slides over a time series t and predicts for every step a separate value y not included in the time series. A regression model predicts the next time point t_{i+1} or interval based on the previous time series $t_{0,\dots,i}$. For the classification tasks, we will use the UCR Benchmark repository [7] with 128 different data sets. Our forecasting data set will consist of weather forecasting ¹, air quality [38], and store demand forecasting [9]. And lastly, our regression task will incorporate finance data ² and household power consumption [9].

Implementation – For our experiment, we will use PyTorch ³ and Captum ⁴ as primary libraries. PyTorch will be used to build and train the selected models on the selected data sets. Captum will create the attributions we selected to analyze based on the trained PyTorch model. In some cases, like LIME and SHAP, we will use the implementations provided by the authors. We extend the Pytorch LRP implementation for LSTMs with the implementation provided by Arras et al. [2]. For reproducibility, we fix all possible random seeds to 13.

Experiment Run – Our experiment run will consist of independent docker containers that apply our methodology for a selected time series task, data set, and model. Such a configuration enables to fix a time series dataset and a model to investigate the various attribution methods. We run our experiments strictly using the previously presented methodology using our three-stage evaluation. After a successful run, the container will store the trained model and the results in an external database. We can achieve a highly efficient experiment setup through the independent docker containers as hardware constraints of our servers only limit us.

Result Analysis – After our experiment run, we will analyze the results to investigate the XAI techniques’ strengths and weaknesses towards architectures and tasks to come up with a best-practices guideline. Based on these findings, we will either prove or deny general views about XAI techniques applied on other data than images or text answering the questions we brought up in previously. We will also focus on the previously presented exploratory questions and investigate exploratory tasks to identify surprising results. A closer look into these questions enables us to enhance our preregistration with a section of unforeseen outcomes as related work misses hypotheses about the XAI techniques. However, we will reproduce our preliminary results [26] and investigate our previous hypothesis of SHAP being the overall best XAI technique. Our extended set of random baselines will facilitate either prove or reject a significance towards this hypothesis. With this preregistration and the exploratory results and findings, we will establish a useful benchmark for attribution techniques on time series to foster the research into novel methods.

5 Extension as a benchmark framework

As new XAI techniques are getting developed rapidly, we will extend our methodology and experimental setup into a benchmark framework to support novel approaches’ analysis and verification. By enabling users to add their own models, data sets, verification methods, and XAI techniques, the benchmark framework supports to analyze and compare novel methods and models against prominent and state-of-the-art techniques. We will further extend the framework with the concept of ROAR (RemOve And Retrain) by Hooker et al. [15] to enhance the functionality of the benchmark and to be able to also test XAI techniques before production ready models. Through such an extension, our framework will be able to test combinations of XAI techniques to dig deep into the interplay of these. Good working interplays lead to robust explanations of ensembles of XAI techniques.

¹Deutscher Wetterdienst: <https://opendata.dwd.de/>

²Yahho Finance: <https://finance.yahoo.com/quotes/OCR,dataset/view/v1/>

³PyTorch Deep Learning Framework: <https://pytorch.org/>

⁴Captum Model Interpretability for PyTorch: <https://captum.ai/>

Acknowledgments and Disclosure of Funding

This work was partly funded by the European Union’s Horizon 2020 research and innovation programme under grant agreements No 826494.

References

- [1] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, 2018.
- [2] L. Arras, A. Osman, K.-R. Müller, and W. Samek. Evaluating Recurrent Neural Network Explanations. In *Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2019.
- [3] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek. On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PLoS ONE*, 10(7):e0130140, July 2015.
- [4] J. Castro, D. Gómez, and J. Tejada. Polynomial calculation of the Shapley value based on sampling. *Computers & Operations Research*, 36(5):1726–1730, 2009.
- [5] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.
- [6] G. E. Dahl, D. Yu, L. Deng, and A. Acero. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on audio, speech, and language processing*, 20(1):30–42, 2011.
- [7] H. A. Dau, E. Keogh, K. Kamgar, C.-C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, Yanping, B. Hu, N. Begum, A. Bagnall, A. Mueen, and G. Batista. The UCR Time Series Classification Archive. www.cs.ucr.edu/~eamonn/time_series_data/, Oct. 2018.
- [8] F. Doshi-Velez and B. Kim. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv preprint arXiv:1702.08608*, 2017.
- [9] D. Dua and C. Graff. UCI machine learning repository. <http://archive.ics.uci.edu/ml>, 2017.
- [10] European Union. European General Data Protection Regulation. Technical report, 2018.
- [11] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi. A Survey Of Methods For Explaining Black Box Models. *ACM Computing Surveys*, 51(5), 2018.
- [12] Gunning, D. Explainable Artificial Intelligence (XAI) DARPA-BAA-16-53. Technical report, Defense Advanced Research Projects Agency (DARPA), 2016.
- [13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.
- [14] S. Hochreiter and J. Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8), 1997.
- [15] S. Hooker, D. Erhan, P.-J. Kindermans, and B. Kim. A benchmark for interpretability methods in deep neural networks. In *Advances in Neural Information Processing Systems*. 2019.
- [16] B. Huval, T. Wang, S. Tandon, J. Kiske, W. Song, J. Pazhayampallil, M. Andriluka, P. Rajpurkar, T. Migimatsu, R. Cheng-Yue, et al. An empirical evaluation of deep learning on highway driving. *arXiv preprint arXiv:1504.01716*, 2015.
- [17] A. Kapishnikov, T. Bolukbasi, F. Viégas, and M. Terry. Xrai: Better attributions through regions. In *IEEE International Conference on Computer Vision*, pages 4948–4957, 2019.
- [18] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

- [19] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, and X. Yan. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. In *Advances in Neural Information Processing Systems*, pages 5243–5253, 2019.
- [20] S. Lundberg and S.-I. Lee. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems*, volume 16, pages 426–430, May 2017.
- [21] R. K. Mobley. *An Introduction to Predictive Maintenance*. Plant Engineering. Butterworth-Heinemann, Burlington, second edi edition, 2002.
- [22] S. Mohseni, N. Zarei, and E. D. Ragan. A Survey of Evaluation Methods and Measures for Interpretable Machine Learning. *arXiv preprint arXiv:1811.11839*, 2018.
- [23] G. Montavon, S. Lapuschkin, A. Binder, W. Samek, and K.-R. Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65, 2017.
- [24] M. T. Ribeiro, S. Singh, and C. Guestrin. "Why Should I Trust You?". In *International Conference on Knowledge Discovery and Data Mining*, 2016.
- [25] C. Rudin. Please Stop Explaining Black Box Models for High Stakes Decisions. *arXiv preprint arXiv:1811.10154*, Nov. 2018.
- [26] U. Schlegel, H. Arnout, M. El-Assady, D. Oelke, and D. A. Keim. Towards a Rigorous Evaluation of XAI Methods on Time Series. In *ICCV Workshop on Interpreting and Explaining Visual Artificial Intelligence Models*, 2019.
- [27] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *International Conference on Computer Vision*, volume 2017-Octob, pages 618–626, 2017.
- [28] A. Shrikumar, P. Greenside, and A. Kundaje. Learning Important Features Through Propagating Activation Differences. *International Conference on Machine Learning*, 2017.
- [29] A. Shrikumar, P. Greenside, A. Y. Shcherbina, and A. Kundaje. Not Just A Black Box: Learning Important Features Through Propagating Activation Differences. *arXiv preprint arXiv:1605.01713v2*, 2016.
- [30] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv preprint arXiv:1312.6034*, 2013.
- [31] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg. SmoothGrad: Removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.
- [32] T. Spinner, U. Schlegel, H. Schäfer, and M. El-Assady. explAiner: A Visual Analytics Framework for Interactive and Explainable Machine Learning. *IEEE Transactions on Visualization and Computer Graphics*, 2019.
- [33] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.
- [34] M. Sundararajan, A. Taly, and Q. Yan. Axiomatic Attribution for Deep Networks. In *International Conference on Machine Learning*, pages 3319–3328. JMLR. org, Mar. 2017.
- [35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention Is All You Need. *Advances in Neural Information Processing Systems*, 30, 2017.
- [36] M. D. Zeiler and R. Fergus. Visualizing and Understanding Convolutional Networks. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 818–833. Cham, 2014. Springer International Publishing.
- [37] J. Zhang, S. A. Bargal, Z. Lin, J. Brandt, X. Shen, and S. Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10):1084–1102, 2018.
- [38] S. Zhang, B. Guo, A. Dong, J. He, Z. Xu, and S. X. Chen. Cautionary tales on air-quality improvement in Beijing. *Royal Society A: Mathematical, Physical and Engineering Sciences*, 2017.