A Benchmark for Interpretability Methods in Deep Neural Networks

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, Been Kim
Google Brain
shooker,dumitru,pikinder,beenkim@google.com

김 성 철

Contents

Introduction

- **XAI (Explainable AI)**

  1. There is no ground truth.
  2. It is unclear which should select.

  → We need a framework to validate the relative merits and reliability

- **Commonly used strategy**

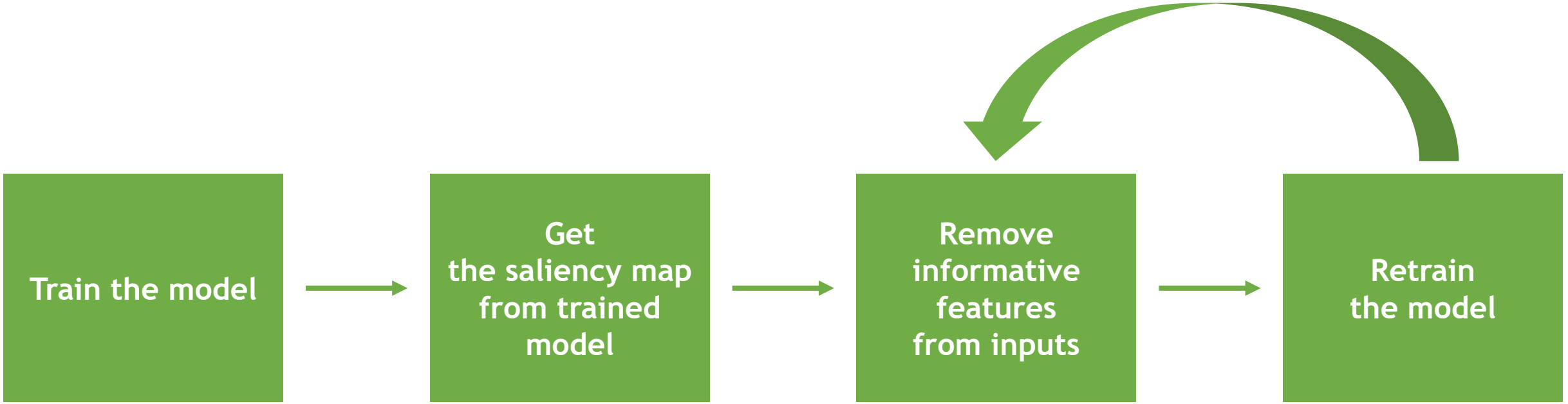
  : **Remove** the informative features & **look** at how the classifier degrades

  → Samples where the features are removed come from a different distribution!

  → It is unclear whether the degradation in model performance comes from the distribution shift.

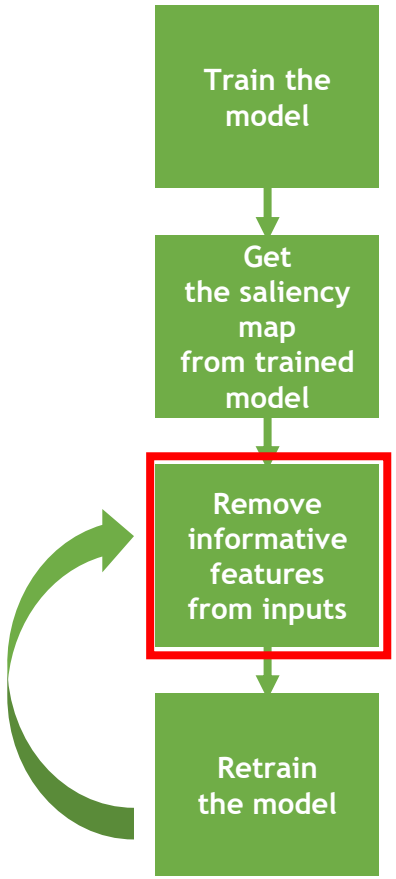
Introduction

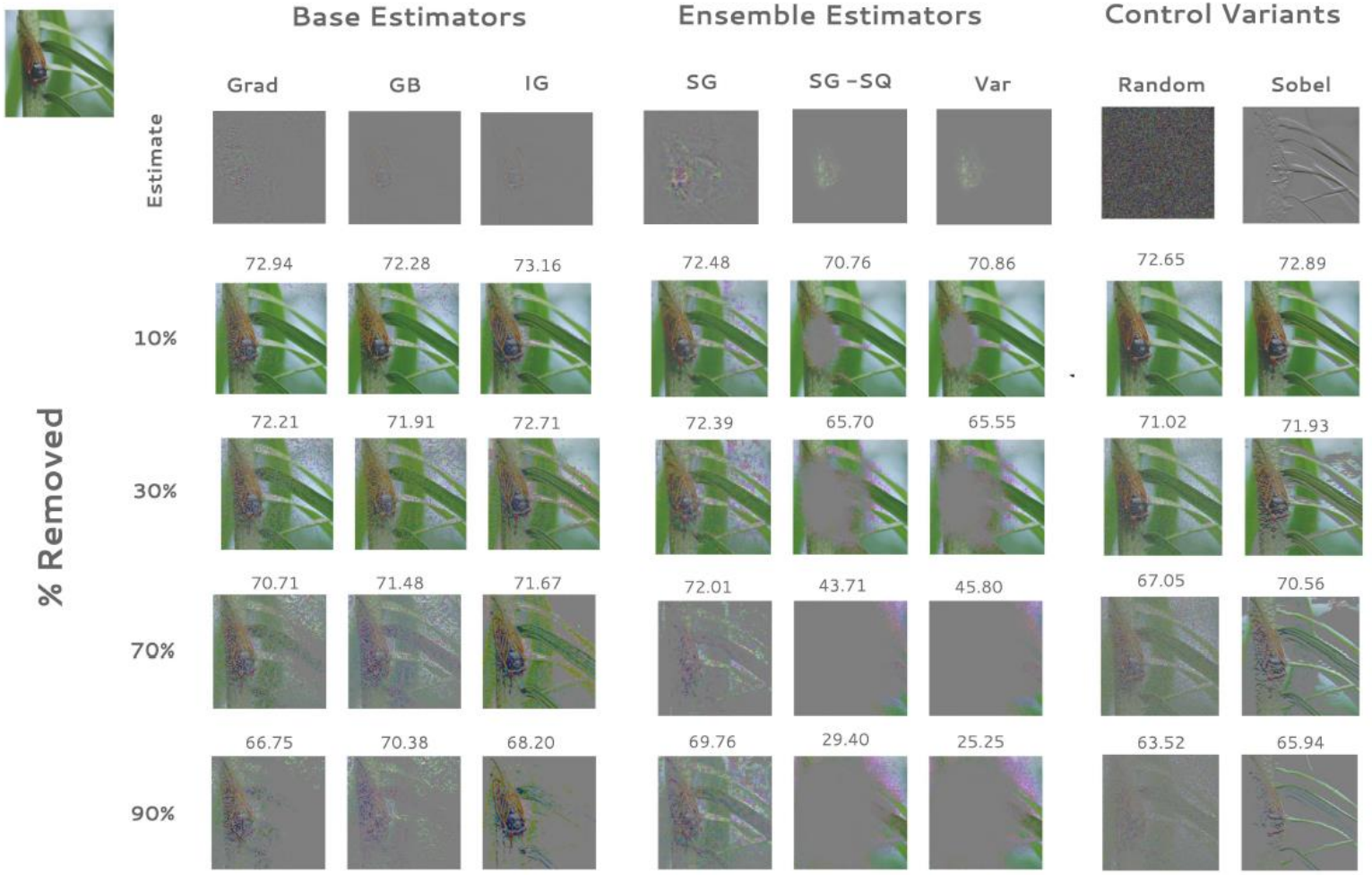
- RemOve And Retrain (ROAR)



Introduction

- RemOve And Retrain (ROAR)





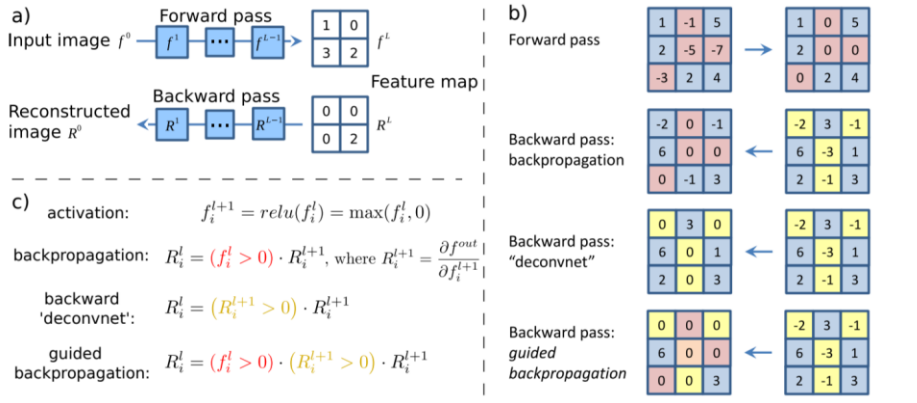
Interpretability methods

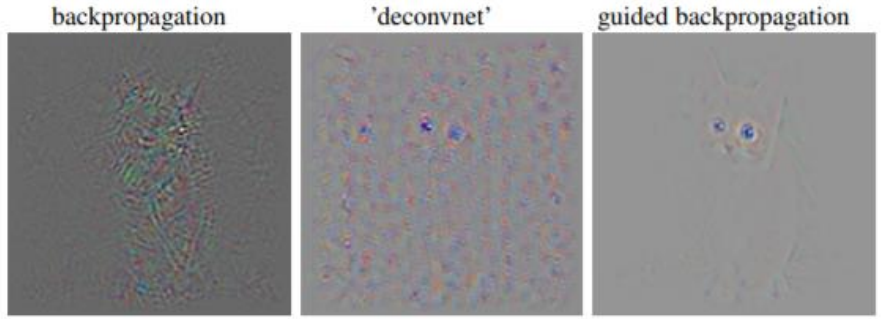
- Base estimators
  - Gradients or Sensitivity heatmaps (GRAD)

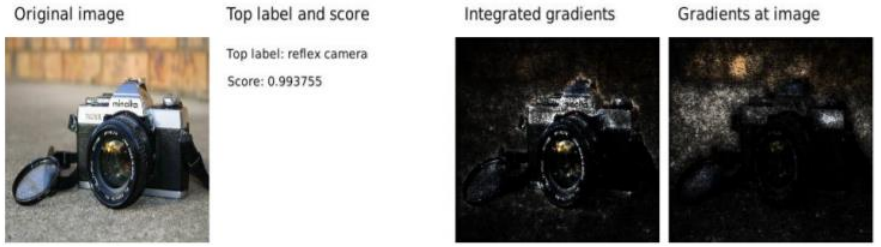
$$e = \frac{\partial A_n^l}{\partial x_i}$$

  - Guided Backpropagation (GB)
  - Integrated Gradients (IG)

$$e = (x_i - x_i^0) \times \sum_{i=1}^{k} \frac{\partial f_w(x^0 + \frac{i}{k}(x - x^0))}{\partial x_i} \times \frac{1}{k}$$







https://arxiv.org/pdf/1412.6806.pdf
https://arxiv.org/pdf/1703.01365.pdf

Interpretability methods

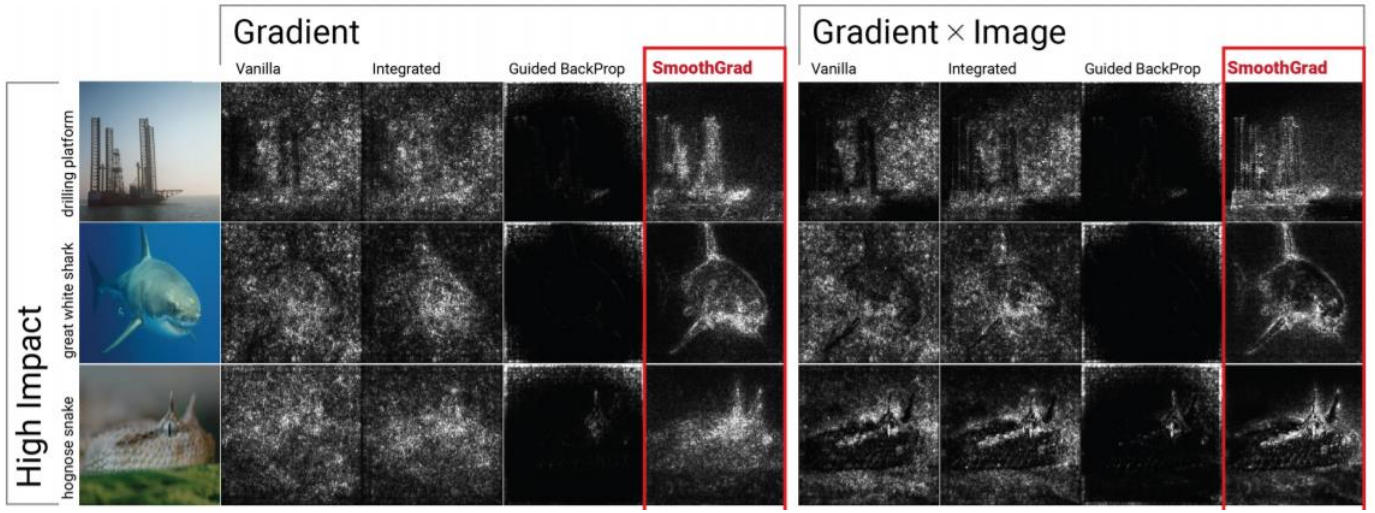
- Ensembling methods
  - Classic SmoothGrad (SG)

$$e = \sum_{i=1}^{J} (g_i(x + \eta, A_n^l))$$

  - SmoothGrad^2 (SG-SQ)

$$e = \sum_{i=1}^{J} (g_i(x + \eta, A_n^l)^2)$$

  - VarGrad (Var)
    - $e = \mathrm{Var}(g_i(x + \eta, A_n^l))$



https://arxiv.org/pdf/1706.03825.pdf

Interpretability methods

- Control Variants
  - Random
  - Sobel Edge Filter

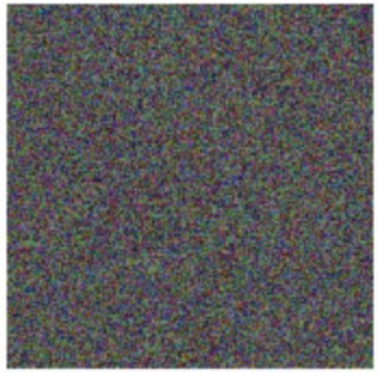
-1	0	+1
-2	0	+2
-1	0	+1

x filter

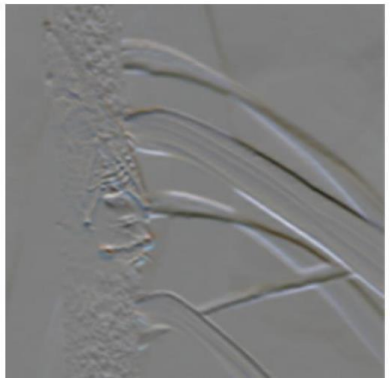
+1	+2	+1
0	0	0
-1	-2	-1

y filter

Random



Sobel



https://m.blog.naver.com/PostView.nhn?blogId=roboholic84&logNo=220482877717&proxyReferer=https%3A%2F%2Fwww.google.com%2F

ROAR : RemOve And Retrain

- Mechanism
  1. Get an estimate e of feature importance from every input
  2. Rank each e into an ordered set $\{e_i^o\}_{i=1}^N$
  3. Replace the corresponding pixels in the raw image with the per channel mean
  4. Generate new train and testset at different degradation levels t = [0., 10., ···, 100]
  5. Retrain!

ROAR : RemOve And Retrain

- **Why do we need to retrain the model?**

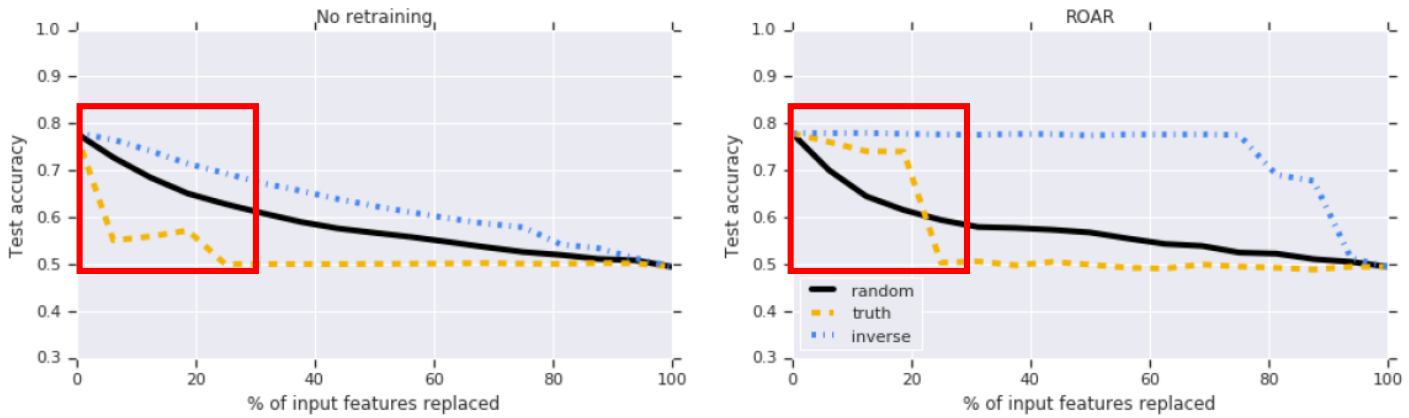
1. The train and testset must have **a similar distribution!**
2. 

Figure 2: A comparison between not retraining and ROAR on artificial data. In the case where the model is not retrained, test-set accuracy quickly erodes despite the worst case ranking of redundant features as most important. This incorrectly evaluates a completely incorrect feature ranking as being informative. ROAR is far better at identifying this worst case estimator, showing no degradation until the features which are informative are removed at 75%. This plot also shows the limitation of ROAR, an accuracy decrease might not happen until a complete set of fully redundant features is removed. To account for this we measure ROAR at different levels of degradation, with the expectation that across this interval we would be able to control for performance given a set of redundant features.

ROAR : RemOve And Retrain

- Results

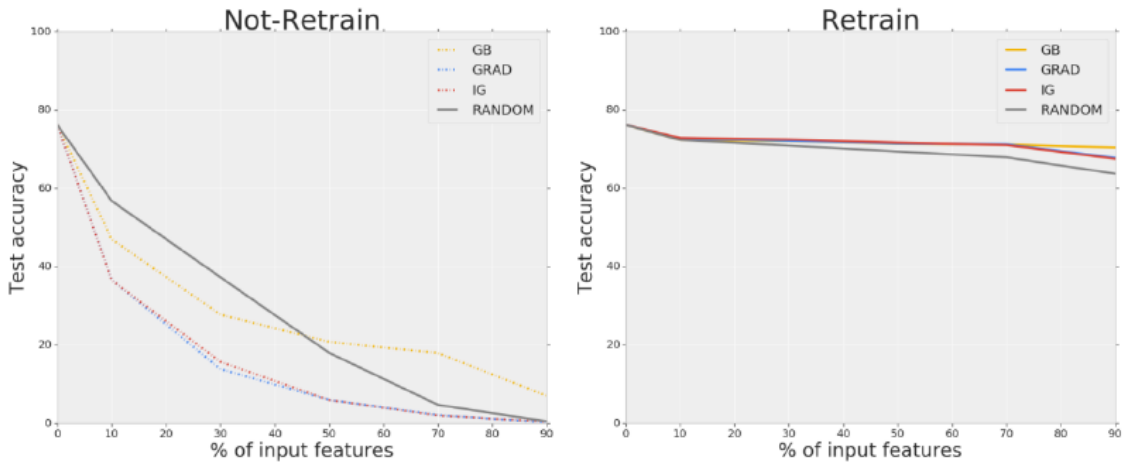


Figure 3: On the left we evaluate three base estimators and the random baseline without retraining. All of the methods appear to reduce accuracy at quite a high rate. On the right, we see, using ROAR, that after re-training most of the information is actually still present. It is also striking that in this case the base estimators perform worse than the random baseline.

ROAR : RemOve And Retrain

- Results

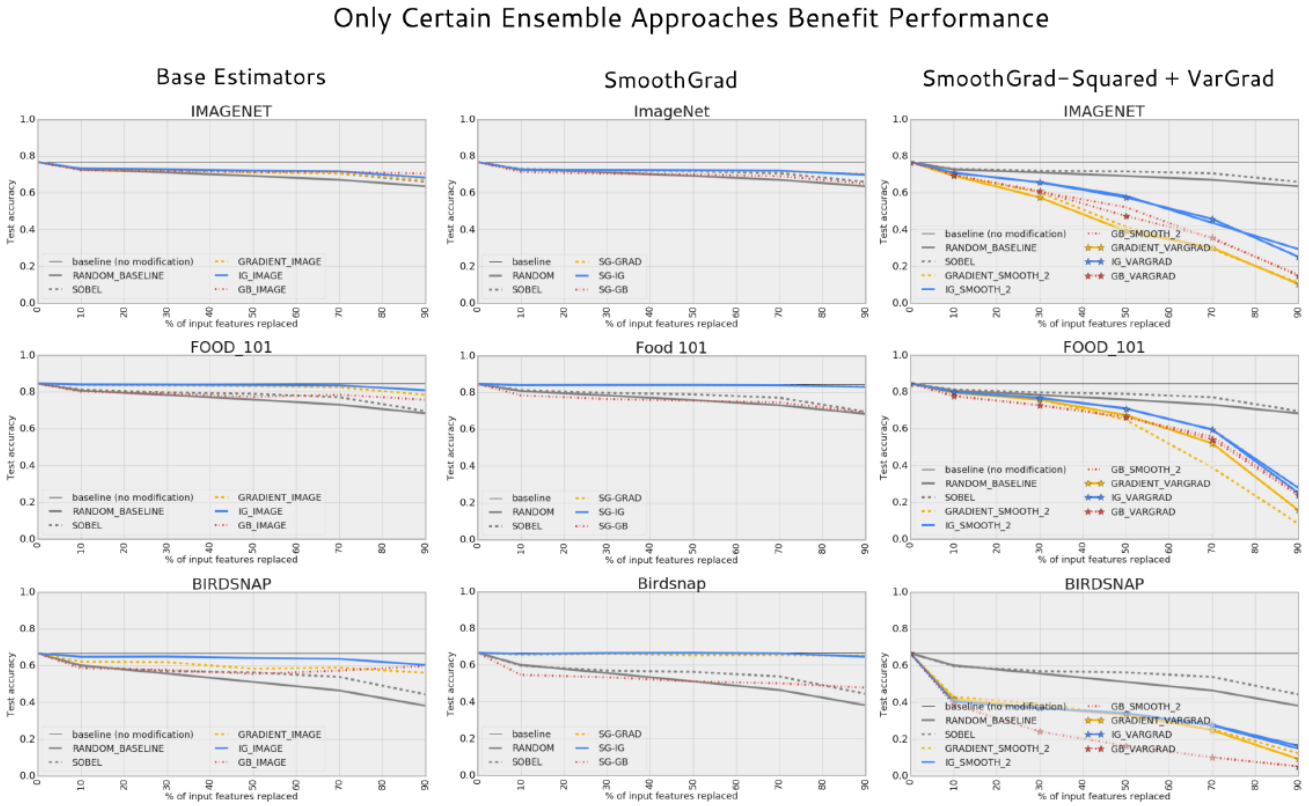
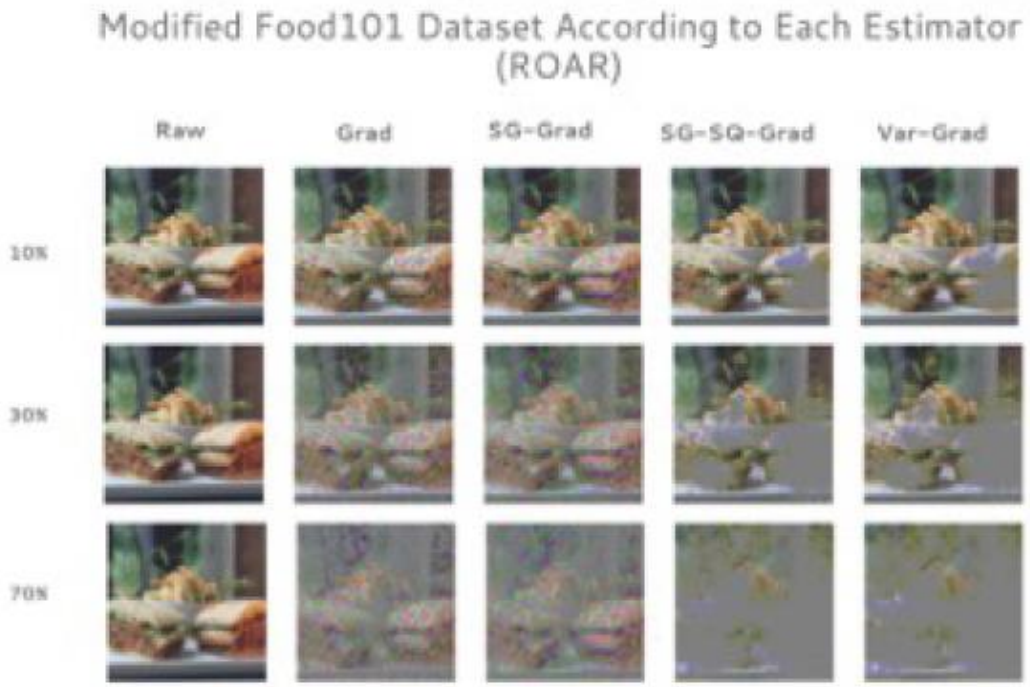
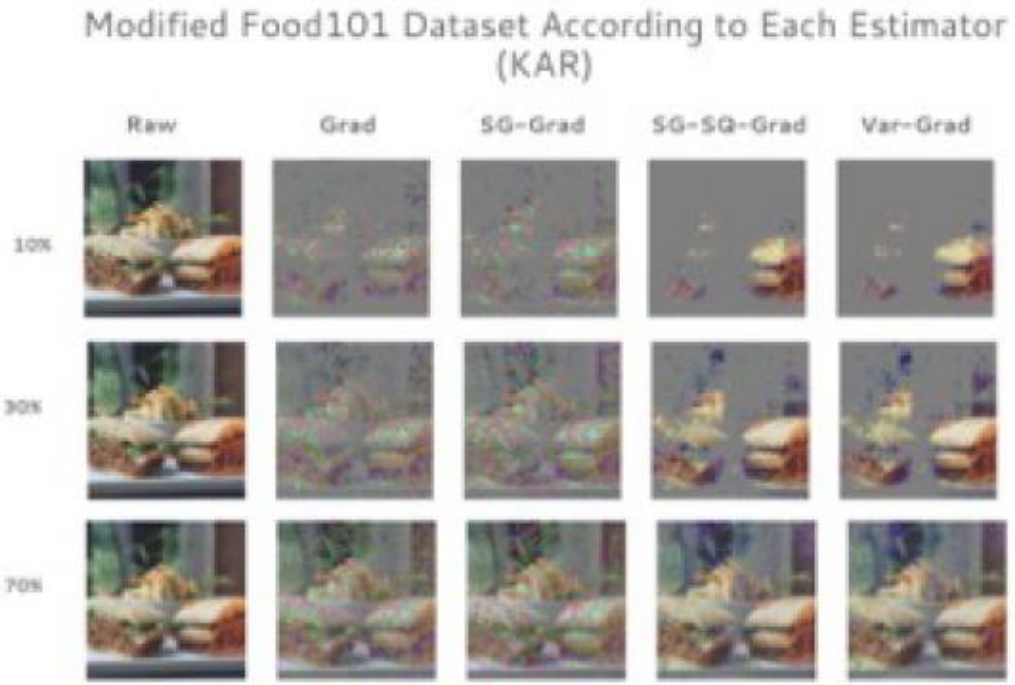


Figure 4: **Left:** Grad (GRAD), Integrated Gradients (IG) and Guided Backprop (GB) perform worse than a random assignment of feature importance. **Middle:** SmoothGrad (SG) is less accurate than a random assignment of importance and often worse than a single estimate (in the case of raw gradients SG-Grad and Integrated Gradients SG-IG). **Right:** SmoothGrad Squared (SG-SQ) and VarGrad (VAR) produce a dramatic improvement in approximate accuracy and far outperform the other methods in all datasets considered, regardless of the underlying estimator.

KAR : Keep And Retrain





KAR : Keep And

- Results

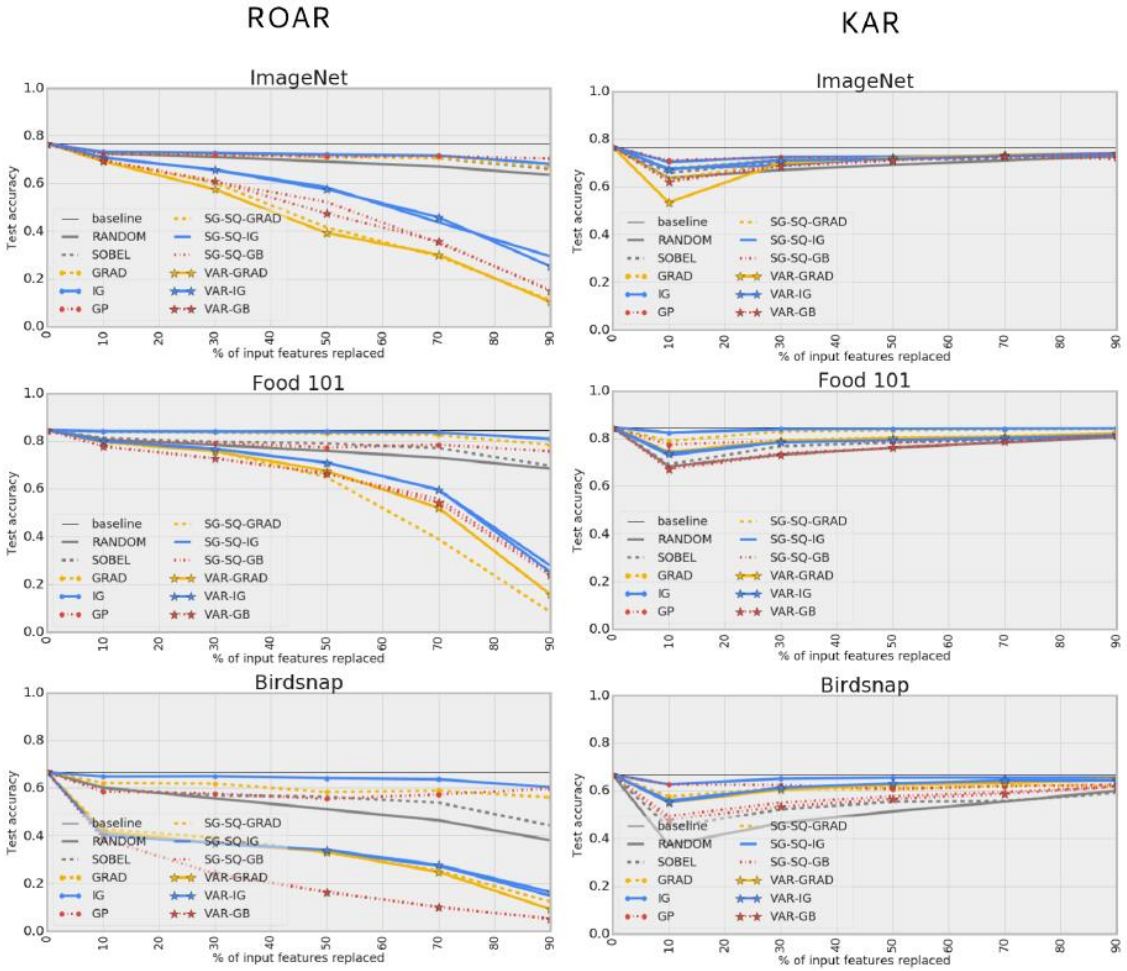


Figure 5: Evaluation of all estimators according to Keep and Retrain KAR vs. ROAR. **Left inset:** For KAR, **K**eep **A**nd **R**etrain, we keep a fraction of features estimated to be most important and replace the remaining features with a constant mean value. The most accurate estimator is the one that preserves model performance the most for a given fraction of inputs removed (the highest test-set accuracy).**Right inset:** For **ROAR**, **R**em**0**ve **A**nd **Retrain** we remove features by replacing a fraction of the inputs estimated to be most important according to each estimator with a constant mean value. The most accurate estimator is the one that degrades model performance the most for a given fraction of inputs removed. Inputs modified according to KAR result in a very narrow range of model accuracy. ROAR is a more discriminative benchmark, which suggests that retaining performance when the most important pixels are removed (rather than retained) is a harder task.

Conclusion

In this work, we propose ROAR to evaluate the quality of input feature importance estimators. Surprisingly, we find that the commonly used base estimators, Gradients, Integrated Gradients and Guided BackProp are worse or on par with a random assignment of importance. Furthermore, certain ensemble approaches such as SmoothGrad are far more computationally intensive but do not improve upon a single estimate (and in some cases are worse). However, we do find that VarGrad and SmoothGrad-Squared strongly improve the quality of these methods and far outperform a random guess. While the low effectiveness of many methods could be seen as a negative result, we view the remarkable effectiveness of SmoothGrad-Squared and VarGrad as important progress within the community. Our findings are particularly pertinent for sensitive domains where the accuracy of a explanation of model behavior is paramount. While we venture some initial consideration of why certain ensemble methods far outperform other estimator, the divergence in performance between the ensemble estimators is an important direction of future research.

감 사 합 니 다