We would like to thank all the reviewers for their insightful comments and for finding this work interesting overall. We'll carefully incorporate your helpful suggestions into the paper revision.

Reviewer#1 Thank you for your very thoughtful feedback. We benefit a lot from it. We agree that there has been several interesting works applying Shapley values for different aspects of interpretability. We tried to discuss the most relevant literature and will add more discussion of the missing citations. One distinction we'd like to draw is that our method is designed for interpreting the individual elements/filter of the model while most the existing literature is designed for computing feature-importance scores. As discussed in the paper, we are aware that such methods (including Ancona et al and Lundberg et al) can be extended for computing neuron-importance, even though it was not used this way in the original papers. Assume each layer to be the input of the model and find the importance of neurons in that layer using one of these methods. Then repeat this process for every layer and get importance scores for all neurons. However, one of our desiderata is that the scores should incorporate the joint interactions of elements in different layers. Existing methods will only look at joint behavior of elements that are in the same layer. Directly modeling the network as a cooperative game between all of the neurons is a principled framework to account for all of the neuron interactions. Similar idea was also explored by Stier et al and Florin Leon for pruning small neural networks and our MAB approach enables us to compute scores for very large DNNs. **As the reviewer requested, we have added comparisons with Neuron Conductance for Fig2 during the response.** Neuron Shapley has a superior performance in finding the class-specific neurons. For conductance, removing top-10 and top-20 class specific neurons for the "Carousel" class reduced accuracy to 64% and 40% compared to 20% and 8% using Shapley (both methods use 25 carousel images). We will revise Fig 2 to add these and other comparisons. **Regarding model repair for adversarial attacks**, our results strengthen the overall message of the paper – adversarial vulnerability can be attributed to a small subset of neurons. Removing this subset substantially improved black-box robustness. While white-box attack is harder to defend, in many applications adversaries only have black-box access and pruning can still be an effective repair. Overall our results highlight how the sparsity of Neuron Shapley values can facilitate model repair (which has not been explored as much) and interpretation. Model reduction through pruning is a different application is an interesting future direction of research (and there's some work on this) which is separate from the scope of our current work. The reviewer suggested **comparing the number of passes in the model between neuron Shapley and conductance**. We will add this to the revision. Overall the two methods require the similar number of passes. In neuron Shapley, each computation of $V(.)$ on Inception-v3 (i.e. each forward pass) is performed on a batch of 128 random samples (out of 25000 images). By running the algorithm for 3000 iterations, given that most iterations are truncated after removing less than 1500 filters, Neuron-Shapley requires around 4.5×10^7 forward-passes. For neuron conductance, given the original suggested number of steps of 50 for Riemann approximations, the method requires around 4.6×10^7 gradients (i.e. forward+backward passes). The Shrikumar et al's paper mentioned by the reviewer has similar computational complexity to our implementation of neuron-conductance work. The reviewer mentions the discrepancy in the number of images used for neuron-conductance and Shapley. We used 100 images for conductance because that's what was used in the original work. Moreover, the number of images used for Shapley led to the two methods having the same computational cost. **For the response, we ran neuron-conductance with 200 images and found no significant improvement.** For Inception-v3 model, removing 57 filters reduced accuracy to random (from 59 using 100 images). For the fairness experiment, removing the top 105 unfair filters, accuracy on PPB increased to 88.5% (88.7% using 100 images). The Shapley results are still substantially better.

Reviewer#3 Thanks for your insightful review and support of the paper! Regarding the location and interaction of the important filters, Fig.1 (lower panel) shows how many important neurons are in each layer. We'll extend these results and include statistics of important neurons for other experiments. It's an interesting question for other ways to find the top-k filters. There are not many principled approaches for doing this to the best of our knowledge. There are strategies using gradient methods to assign neuron importance; conductance is a SOTA method in this class. Our experiments demonstrate that Shapley substantially outperforms conductance. Moreover Shapley uniquely satisfies several desirable mathematical properties. Thanks for mentioning the "Reward Structures" work; we'll cite and discuss this work.

Reviewer#4 Thank you for your positive comments and helpful suggestions. It's a great advice to make better connections to the bandit literature. We'll incorporate this in the revision and will edit the description of algorithm in our work to make this connection clear. We can also add theoretical guarantees for our algorithm. TMAB-Shapley is solving a top-k arm selection problem where each arm's distribution is bounded (between -1 and1) for which; we can directly adapt the regret bounds from the MAB literature to analyze our method. The reviewer mentions comparison with alternative neuron importance measures. We have provided comparisons with the Neuron-Conductance method which is the best performing of the existing neuron-importance measures. Shapley performed substantially better than conductance. We have also added more comparisons to conductance in the response (please see the last few comments to Reviewer 1). Your suggestion to combine top-k arm selection and static model-pruning is very interesting and is great for further research. This first work is mostly focused on applications of Neuron Shapley for model-repair and interpretability (rather than pruning) and our experiments are designed for these use-cases.