We thank all reviewers for their insightful and helpful feedback.

R2: It would be nice to see how FrugalML performs when limited to only using MLaaS services APIs, excluding GH.

A: We conducted an additional experiment on dataset FER+ using only MLaaS APIs excluding GH. To match the best API (Microsoft)'s performance, the learned FrugalML strategy always uses Face++ (5$) as the base service and occasionally calls Microsoft API (10$), leading to overall cost reduction of 17%. Alternatively, using the same cost target as the best API (10$), FrugalML achieves a 2% accuracy improvement. We'll add this to the revision.

R3: As the approach is strongly related to ensemble methods, one could additionally mention other seminal works (e.g. [1,2,3]) and not highlight mixture-of-experts alone, which, of course, is seminal as well.

A: Thanks for the suggestion; we will discuss these related works in the revision.

From the paper I extract that you learn a model which performs instance-wise predictions, correct? How much left-out training data of the particular dataset (or other datasets) do you use for this? How easy/difficult is this task and do the results vary on the used datasets?

A: Yes. Except for Figure 5, all experiments use 50% data for training and the remaining for evaluation. As Figure 5 shows, when the training sample size is larger than a few thousands, the performance becomes steady.

How would the results look like if only the best API would be called? Does this coincide with the results for MoE? How would the results look like if only the provided quality score is used? Following this thought, I am wondering why MoE on Facial Emotion Recognition always chooses the same API? How do you calculate the quality score for the Github CNN? And, for the datasets, how good is the quality score as conditional accuracy estimator?

A: The dot points in Figure 4 shows the accuracy if we only allow calling the best API. We are using the provided quality score from those APIs. We hypothesize that MoE chooses the same API because it relies on a linear model on the image features alone, on which the best API dominates. The GitHub CNN model is a VGG-19 variant which adopts a soft-max layer to compute the quality score. As shown in Figure R1 as below, the quality score has a high correlation to conditional accuracy.

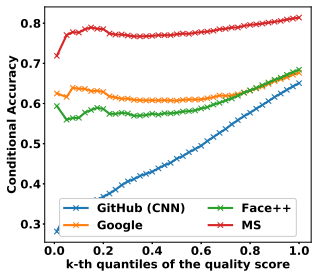

Figure R1: Accuracy conditional on quality score on dataset FER+.

Lastly, why is the proposed protocol of a strategy constrained to a single add-on service? When would the runtime of the proposed approach be a practical issue?

A: This is mainly because using more services may increase the cost and the sample complexity for training. Allowing more add-on services would be an interesting direction of future extension.

R4: I believe that the proposed method will work when the real label distribution is invariant and the same as the initial dataset. But is the proposed method robust when they are not the same? What will happen if the label distribution is changing over time? Should we switch the strategy at some point?

A: In this paper we do assume that the label distribution does not change during inference. When the distribution has changed, the performance of the trained strategies might drop down and retraining or domain adaptation is needed for better performance. We will add a discussion on this for the revision.

R6: The authors could have pushed a bit further on the comparison with other cascade architectures and a better understanding of how robust these results are would be nice. For example can a GAN mess up FrugalAI more so than a quality API.

A: Thanks for the suggestion; we will discuss robustness further in the revision. Our experiments on diverse real datasets from different domains suggest that FrugalML is robust. Testing it on GANs is a great idea for future work. One advantage of FrugalML over model cascade is that standard cascade methods incur a fixed cost while the proposed FrugalML allows for different budget requirements.