Thanks for the detailed feedback! We are glad that reviewers found our results to be interesting.

R1:

> $\mathbb{P}(\mathbf{0}_L|x) = 0 \rightarrow$ I would prefer to not make such assumption.

This was done to simplify the definition of recall, since if no labels are relevant ($y = \mathbf{0}_L$), we would naïvely have to compute $\frac{0}{0}$. We can however remove this assumption, and note that $y = \mathbf{0}_L$ requires fixing a choice for the recall.

> Lemma 3 $\rightarrow$ There are undefined quantities in the lemma and typos in its proof.

We will explicate the meaning of $y_{\neg i}$ (which, as the reviewer correctly inferred, refers to all labels but the ith one), and also add the two forms of $\mathbb{P}(y'_i = 1 \mid x)$ as suggested.

We will incorporate the additional citations and other minor comments, which are appreciated.

R2:

> The authors also call for caution in interpreting the produced probabilities scores of the reduction techniques. But isn't it rather trivial? It is not a criticism; I'd just like to point it out in case I've missed something.

The fact discussed in Section 5.3 that most reductions do not output marginal label probabilities indeed follows immediately from our results. We simply wished to explicate that while OVA with logistic and PAL with softmax cross-entropy loss produce probability estimates, precisely *what* these probabilities measure are fundamentally different.

R3:

> I have some problems to understand the losses in Equations 6 and 7, because the ℓ_{BC} and ℓ_{MC} are never defined.

We work with abstract binary/multiclass losses $\ell_{\mathrm{BC}}/\ell_{\mathrm{MC}}$ to highlight that our results are not tied to specific choices. We tried to make the quantities concrete by providing examples of the logistic and softmax cross-entropy loss on Lines 142 and 148. We use their standard definitions: for binary $y_i \in \{0, 1\}$ the logistic loss is $\ell_{\mathrm{BC}}(y_i, f_i) = \log(1+e^{-(2y_i-1)\cdot f_i})$, while the softmax cross-entropy loss is as defined on Line 98. We will clarify this in our revision.

> I can't see the usefulness of Equation 8 . . . So, this seems to suggest that false positives should be heavily penalized.

To get some intuition, take the special case of square loss, $\ell_{\mathrm{BC}}(y_i, f_i) = (y_i - f_i)^2$. One may verify that $\ell_{\mathrm{OVA-N}}(y, f) = \sum_{i\in[L]}(y'_i - f_i)^2$ plus a constant, for $y'_i = \frac{y_i}{\sum y_j}$. Thus, the provided weighting scheme encourages f_i to estimate the "normalised labels" y'_i, rather than the raw labels y_i. One can obtain similar results for the logistic and hinge loss.

Observe also that the scale of $y'_i \in \{0, \frac{1}{100}\}$ in your example, which is much smaller than that of $y_i \in \{0, 1\}$. To model this compressed range of values, we thus need to shrink our predictions for the positives closer to 0. Placing a large weight on the negative term ($\ell_{\mathrm{BC}}(0, f_i)$) when $y_i = 1$ achieves precisely this. We will add a discussion in our revision.

> Equation 9 is also a strange variant. Here the denominator in the sum does not depend on i, so it can be moved in front of the sum. . . . PAL and PAL-N should therefore have the same risk minimizer.

To get some intuition, per Line 154, the effect of normalisation is to create a valid distribution y'_i over labels. The loss thus seeks to minimise the discrepancy between y'_i and the model distribution q_i over labels; e.g., for the cross-entropy loss, we choose q to minimise $-\sum_{i\in[L]} y'_i \cdot \log q_i$, or equally, $\mathrm{KL}(y'_i \| q_i)$.

It is true that $\sum_{j\in[L]} y_j$ can be moved outside the sum. However, it is not true that this is a constant weight in the risk: for any fixed x, we have to compute $\mathbb{E}_{y|x}\left[\frac{1}{\sum_{j\in[L]} y_j} \cdot \sum_{i\in[L]} y_i \cdot \ell_{\mathrm{MC}}(i, f)\right] \neq \frac{1}{\mathbb{E}_{y|x}[\sum_{j\in[L]} y_j]} \cdot \sum_{i\in[L]} \mathbb{E}_{y|x}[y_i \cdot \ell_{\mathrm{MC}}(i, f)]$ in general. Equality only holds when the number of labels is constant across x; we will make this point explicit.

> Traditionally, there are two ways to optimize task-based loss functions . . . For me, a big point of confusion is that the approaches are somewhat mixed in this paper. Wouldn't it be easier to analyze . . . using accuracy for ℓ_{BC} and ℓ_{MC}.

Ideally, it is always desirable to directly optimise the downstream task-specific measure of ultimate interest. In multilabel retrieval settings, these are typically the precision@k and recall@k; however, their direct optimisation is challenging. This has motivated the reductions proposed in prior work, which have been informally motivated as optimising *some* task-specific multilabel loss. It is precisely the motivation of this work to understand exactly what loss this is.

Both precision@k and recall@k implicitly use the top-k loss (Corollary 8). For $k = 1$ this is exactly the misclassification loss, which is in line with the reviewer's suggestion about using accuracy for ℓ_{BC} and ℓ_{MC}.