**1: Evaluation Setup & Baselines**

**Data Split**  
The dataset was split 80/20 at the file level to avoid leakage, with all the 1.2 second segments from all files remaining together in the same set. Multilabel stratification was applied to the aggregated segment labels, that were computed with get\_ground\_truth\_df(), so each class’s prevalence in training and validation matches the overall distribution. Reproducibility is ensured by a fixed random seed of 42. Each segment corresponds to a 1.2 second window, formed by aggregating 10 consecutive 100 ms frames (the base frame resolution of the features).

**Baseline Systems**  
Two naive baselines were used, an all-zero predictor, assigning 0 to every class in every segment, and a prevalence-based predictor that assigns 1 to classes with occurrence rate >= 0.5 in the training split, and otherwise 0.

**Baseline Costs**  
Using eval\_cost.py, which wraps the provided compute\_cost.py, both baselines were evaluated on the validation split. The all-zero and prevalence baselines each incur a total cost of **107.78** (cost per minute, summed over all classes) on the validation set. This establishes our lower bound for any classifier-based system to outperform.