For our evaluation, we use the StackOverflow dataset from [**?**]. We preprocess the dataset to lexicalize both the broken and fixed code snippets, then filter the dataset by length and edit distance, in which all Python snippets whose broken form is fewer than 80 lexical tokens and whose human fix is under four Levenshtein edits is retained.

For our first experiment, we run the sampler until the human repair is detected, then measure the number of samples required to draw the exact human repair across varying Levenshtein radii.

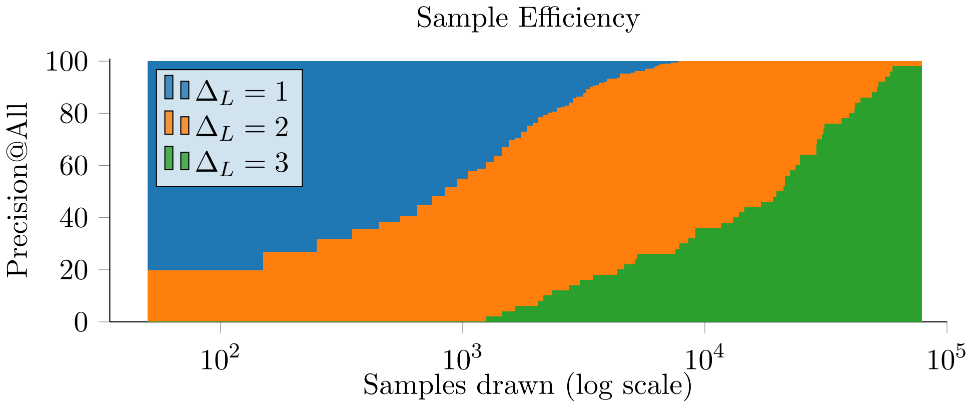

Figure 1: Sample efficiency of LBH sampler at varying Levenshtein radii.

Next, measure the precision at various ranking cutoffs for varying wall-clock timeouts. Here, P@{k=1, 5, 10, All} indicates the percentage of syntax errors with a human repair of $\Delta = \{1, 2, 3, 4\}$ edits found in $\leq p$ seconds that were matched within the top-k results, using an ngram likelihood model.

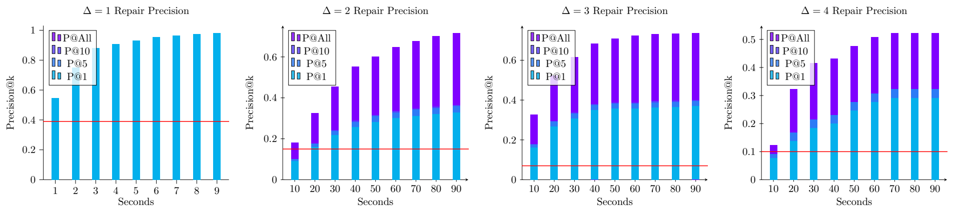

Figure 2: Human repair benchmark. Note the y-axis across different edit distance plots has varying ranges.