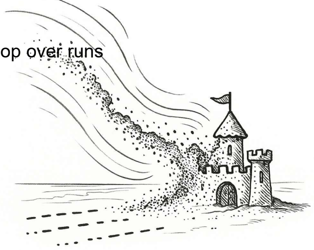
Degeneration Simulation *(pipelineResearch.py)*

Experiment with four “check-tasks” that model differing **error processes** in manual copying and discuss how reliably we can still “carbon-date” a text by distance.

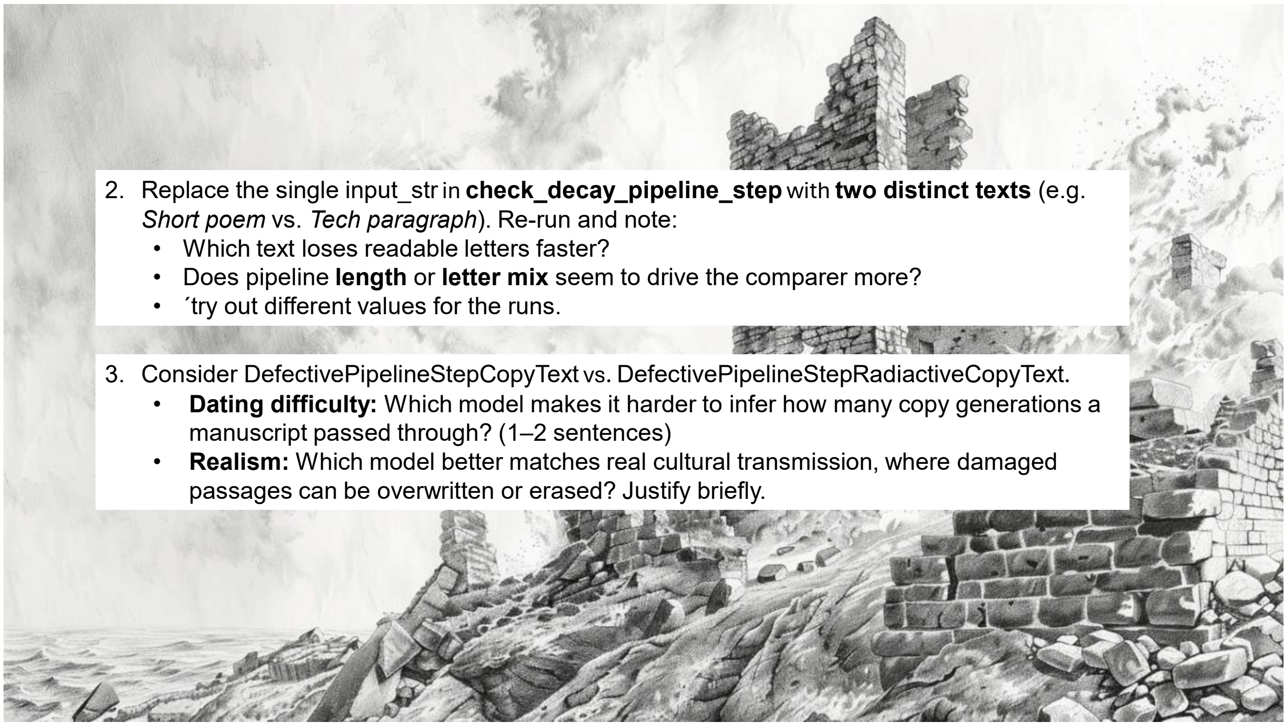
Reminder: SeveralRunsPipeline re-uses the **same** Pipeline idea you met in *pipeline9.py*—only the loop over runs repeats each step.



1 Run the Four Built-in Checks (~10 min)
Execute **all** of:
check_appender()
check_defective_pipeline_step()
check_defective_pipeline_step_radiative()
check_decay_pipeline_step()

Fill the table with the console verdict (which pipeline “wins”) **and** one short note on *why*:

Check	Input(s) length	Runs p1 / p2	Comparer	Verdict	Why?
Appender			count("e")		
Defective (copy)			count("_")		
Defective (radiative)			estimate_runs		
Decay			% of "_"		



2. Replace the single input_str in **check_decay_pipeline_step** with **two distinct texts** (e.g. *Short poem* vs. *Tech paragraph*). Re-run and note:
   - Which text loses readable letters faster?
   - Does pipeline **length** or **letter mix** seem to drive the comparer more?
   - ´try out different values for the runs.

3. Consider DefectivePipelineStepCopyText vs. DefectivePipelineStepRadiactiveCopyText.
   - **Dating difficulty:** Which model makes it harder to infer how many copy generations a manuscript passed through? (1–2 sentences)
   - **Realism:** Which model better matches real cultural transmission, where damaged passages can be overwritten or erased? Justify briefly.