Spelling Correction

Build a spelling corrector that corrects corrupted words according to a noisy-channel model composed of a unigram model and weighted-Levenshtein-distance error model. You may assume that each word is corrupted by at most one edit.

You may use this word frequency data and the provided data files.

Your implementation should provide a single-word spelling-correction function of the form `correct(original: str) -> str`.

Document any modeling assumptions that you make.

Invent some specific scenarios in which your spelling corrector works well and some scenarios where you think it could do better. Explain why it is making poor decisions and suggest one or more ways it could be improved.

You should turn in a document (`.txt`, `.md`, or `.pdf`) answering all of the **red** items above. You should also turn in Python scripts (`.py`) for *each* of the **blue** items. Unless otherwise specified, you may use only `numpy` and the standard library.

References

- Jurafsky and Martin, 2024, appendix D