Introduction to Python and Poetry

Digital Integration Teaching Initiative

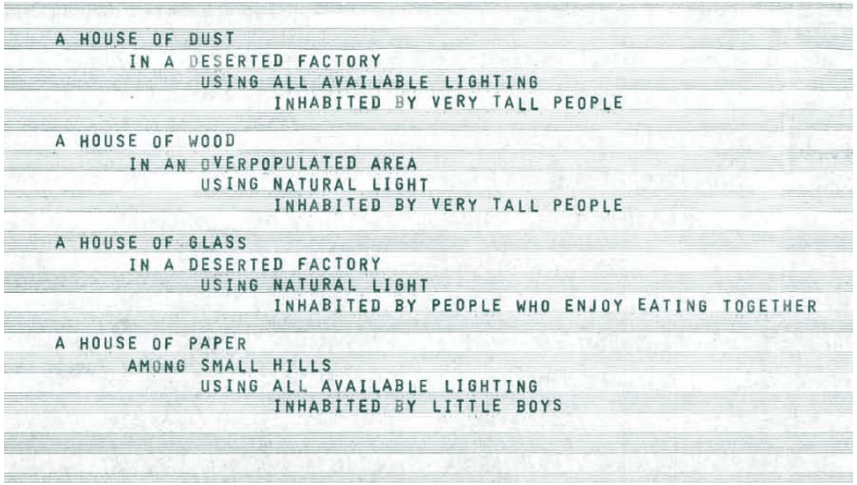
Northeastern University
NULab for Texts, Maps, and Networks

Workshop Agenda

- Computational poetry example and discussion
  - TheHouseOfDust_ExampleComputationalPoem.ipynb
- Python poetry
- Introduction to Python and Google Colab
  - Colab&IntroToPythonPoetry_Lesson.ipynb
  - PythonPoetry_ComputationalPoemTemplate.ipynb
- Generative AI
- Concluding Discussion

Example: “The House of Dust”

- Poem by Alison Knowles and James Tenney (1967)
- Code reimplemented in Python by Nick Montfort and updated as teaching example: TheHouseOfDust_ExampleComputationalPoem.ipynb



Printout of “The House of Dust,” Gebr König Verlag, Cologne, 1967

“The House of Dust” Discussion

- Based on the code for “The House of Dust”, what are the four main building blocks of the poem?
- Can you tell which decisions were made by the author and which are random?
- Can you describe the process of how this poem was written?

Writing Poetry in Python

- Computational poetry using predefined words and lines
  - "The House of Dust" by Alison Knowles and James Tenney (1967)
  - "A Travesty Generator for Micros" by Hugh Kenner and Joseph O'Rourke (1984)
  - *Travesty Generator* by Lillian-Yvonne Bertram (2019)
- AI-generated poetry

Python & Google Colaboratory

Northeastern University
NULab for Texts, Maps, and Networks

Python Summary

The Python code in this workshop covers these topics:

- Variables
- Strings
- Lists
- Dictionaries
- Selecting data from lists and dictionaries
- Print function
- Import random module
- Random.randint() function

Python Google Colab Notebooks

Please refer to the below notebooks to learn more about Python and Google Colab.

- Colab&IntroToPythonPoetry_Lesson.ipynb: This notebook introduces the fundamentals of Python and provides example code for creating computational poetry.
- PythonPoetry_ComputationalPoemTemplate.ipynb: This notebook is a template with some starter code to help you create your own computational poem.

Generative AI

Northeastern University
NULab for Texts, Maps, and Networks

Important AI vocabulary

- Artificial Intelligence (AI): A technology that combines datasets and computer science to solve problems and mimic human intelligence
- Supervised machine learning: An algorithm that classifies or predicts based on its prior training with a labeled dataset
- Unsupervised machine learning: An algorithm that finds patterns or groups in data without prior training
- Generative AI: An algorithm that produces content
- Markov Chains: A series of occurrences where each one depends only on the one directly before
- Word Embedding: A numerical representation of a word

Generative AI Summary

- Uses unsupervised machine learning and other computational methods, such as Markov chains and embeddings, to learn how to generate content
- The type of dataset used to develop the generative AI determines what it can do

Image by DALL-E 3 found in "Text Embeddings: Comprehensive Guide" by Mariya Mansurova

Example: Verse by Verse

- Google Verse by Verse
  - Uses a generative model to create lines of poetry
  - Uses a semantic model to determine the best next line of poetry
- About Google Semantic Experiences

Sample of poets whose works are included in Verse by Verse

AI Ethics Resources

- The Institute for Experiential AI at Northeastern
- United Nations Educational, Scientific and Cultural Organization (UNESCO) Global Forum on the Ethics of AI 2024

Image found in Changing the Landscape of AI Governance, UNESCO

Post-exploration group discussion

- Do you have any reflections on using Python for constructing poetry?
- How does the writing process differ from traditional poetry?
- How might the code impact the readers perception of the poem?
- How might you use this in the future?

Thank you!

—Developed by Sara Morrell, Dipa Desai, and Kasya O'Connor

Grant

- For more information on the DITI, please see: https://bit.ly/diti-about
- Schedule an appointment with us! https://bit.ly/diti-meeting
- If you have any questions, contact us at: nulab.info@gmail.com

Northeastern University
NULab for Texts, Maps, and Networks

Learn More

- Bertram, Lillian-Yvonne. *Travesty Generator*. Noemi Press, 2019.
- Turkel, William J., and Adam Crymble. "Manipulating Strings in Python." *Programming Historian*, 17 July 2012.
- Santillan, Marvin C., and Arnulfo P. Azcarraga. "Poem generation using transformers and doc2vec embeddings." *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020.