Reproducibility for distributional semantics - An applied example

Laura Raveling

2025-05-19

Reproducibility setup

- testing the repro set up with different tools: quarto, python and VS Code
- R Studio also renders quarto markdown files (with the jupyter engine, knitr only knows R)
- combination with latex style rendering
- there are several options to render latex in R Studio

Modeling semantic search processes

- How can a more "static" and a more "dynamic" approach to representations (of representations) be effectively combined? How to model processes over embeddings and how to model the interaction between the structure of the search space and the search processes systematically?
- see, for instance the debate about models and processes (e.g. [2])
- How to make the simulation of a search process more "biologically plausible"?
- test this approach: setup architecture to combine embeddings with systematic searches with agent based modeling or different methods. The notebooks are an attempt to work with the method in [1] who are proposing and agent based model for semantic memory search
- inspired by animal foraging model
- relevance of animal behavior for search heuristics:
- see for instance ([4]), e.g. citing Tinbergen: "It begins to be difficult, and even in some cases impossible, to say where ethology stops and neurophysiology begins." (Tinbergen, 1963)

- "A generative approach allows for the construction of a virtual environment that can be used to study different mechanisms and their interactions, providing a way to analyze systems that may be difficult to get data from (such as specific clinical populations with cognitive impairments)." [1, p. 1115]

Agent based modeling overview

- as more "dynamic" modeling approach
- [python library mesa] (https://mesa.readthedocs.io/latest/)
- (https://www.youtube.com/watch?v=mScpHTIi-kM) [Youtube Video about Prisoner's Dilemma]
- or sociology use case: [3]

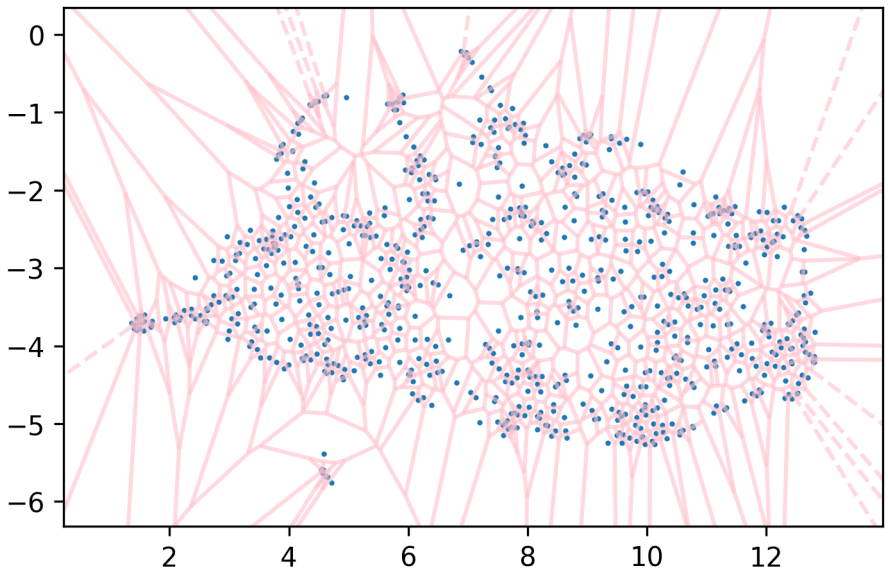
Tools and Data

- from osf repository of replicated study
- the code is partly redundant to abm_semantics.py to explain and visualize
- whole model is in abm_semantics.py
- for running the model in the python script: check if you need to install the requirements in the "requirements.txt" file with pip
- get the word embedding data: in this case the word frequencies, cosine similarities and labels of the animal semantic fluency production task
- for this dataset, there are a total of N = 760 words in the corpus

- python library "mesa": this is intended as a high level library for agent based models
- it is also possible to simply combine any "igraph"- structure with the search function
- for now i wanted to test mesa, to get a high level understanding of the modeling work-flow but i am not sure how practicable it is for the semantic search

Plot data

- decided to use umap instead of t-sne for ease of testing (not having to search for the code of another dimensionality reduction technique)
- but maybe this also makes sense on a content-level, but i have no exact reference as to why
- visualize the data as voronoi plot because it looks like a cute spider web



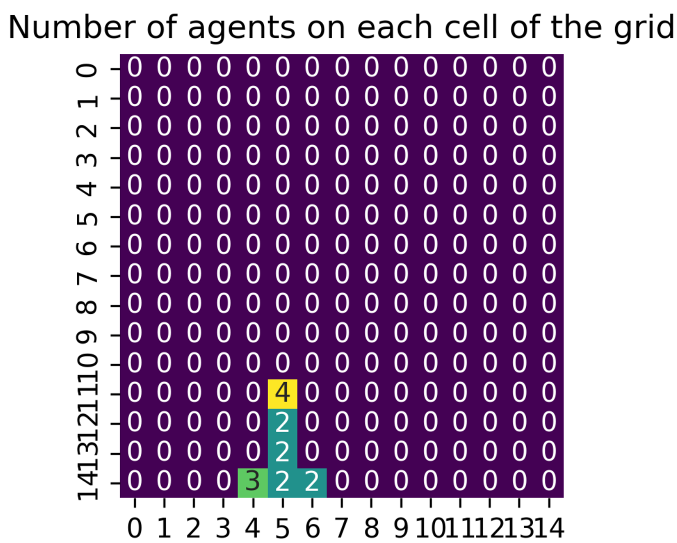
The model:

- one “agent” searches for the words in a semantic space according to the semantic scent algorithm
- if you want to run the original versions of [1]s models, you have to download NetLogo (https://ccl.northwestern.edu/netlogo/)
- sidenote: the NetLogo environment needs the python environment path variable specified to version that has the required packages preinstalled, at this point did know how to integrate this software into the reproducibility workflow for different operating systems

Set up the model landscape

- build the model landscape: sample words weighted by production frequency
- the model landscape is a 2 dimensional grid that is the search space
- are there other plausible versions of this grid such as the voronoi grid and how this could be integrated into the model ?
- format the data
- create a grid and add the words at coordinates defined by the 2-D embedding
- the (graph) properties of these “fixed agents” are the word frequencies and their names

- create the "word seeker" agent that moves through the embedding grid according to the search function
- the search function (as in [1]): semantic scent
- semantic scent function: "likelihood of switching from a patch based on the item just produced, and the proximity to neighborhood items" (p. 1116)
- the model switches between a local and a global search strategy based on the semantic scent decision criterion
- this is supposed to be more plausible than random walks
- the code of this notebook (not displayed in the pdf file) is a exemplary model setup, without the search functions just to see how it should work and to get plots
- after creating the model, there should 15 agents, 1 word seeker and 30 sampled words
- visualize the model landscape



- the agents are all in one corner and there are several agents per cell
- the search space is discrete so very similar agents land at the same coordinates
- i am note sure if this makes sense

Fitting the model

- big "?": "For the parameter optimization of the models, we used the Simulated Annealing method implemented in NetLogo's BehaviorSearch (Stonedah, 2010)." [1, p. 1117]

- i don't know how to do this in python
- no participant data available in the osf

Evaluation criteria

- i have listed these because they could maybe interesting criteria to consider for other evaluations of semantic models and participants data
- according to [1] there are several criteria that could be used for evaluation:
- the similarity between a word and the words preceding it
- "ratio of pairwise similarity over the subject's mean similarity by patch entry position"
- i am not sure what ratio of pairwise similarity could be. Does it check if the model at a global-local switch is close enough to the participants responses at occasions when they switch to different categories?
- the residual proximity (mean similarity to all possible remaining words) of an item to an item's position before or after a patch transition
- the mean ratio between the inter-item retrieval time (IRT) for an item and the participant's mean IRT over the entire task, relative to the order of entry for the item. - distribution of numbers of words, similarity and frequency values
- the average number of patches, and the average patch size.

References

[1] Diego Morales Bader, Enrique Canessa, and Sergio E Chaigneau. "An Agent-Based Model of Foraging in Semantic Memory". In: ().

[2] Thomas T. Hills and Yoed N. Kenett. "Is the Mind a Network? Maps, Vehicles, and Skyhooks in Cognitive Network Science". In: Topics in Cognitive Science 14.1 (Jan. 2022), pp. 189–208. ISSN: 1756-8757, 1756-8765. DOI: 10.1111/tops.12570. URL: https://onlinelibrary.wiley.com/doi/10.1111/tops.12570 (visited on 08/26/2024).

[3] Marijn A Keijzer and Michael Mäs. "The complex link between filter bubbles and opinion polarization". In: Data Science 5.2 (2022), pp. 139–166.

[4] Dean Mobbs et al. "Foraging for foundations in decision neuroscience: insights from ethology". In: Nature Reviews Neuroscience 19.7 (July 2018), pp. 419–427. ISSN: 1471-003X, 1471-0048. DOI: 10.1038/s41583-018-0010-7. URL: https://www.nature.com/articles/s41583-018-0010-7 (visited on 05/19/2025).