

%%%%%%%%%

# Hierarchical Active Inference for Language

Nested hierarchies of structures using Active inference for language recognition and reading

## Description

Each level of the hierarchy has a default structure if the lowest observable is a letter o(1)=letter a simple hierarchy can be described level 1 two factor: states = [composition of letters (e.g. word), locations of letters in the word] obs = [ letter, locations of letters in the word] than level 2 states = [composition of words (e.g. sentence), locations of words in the sentence] obs = [ word, locations of words in the sentence]

Extending the composition and the meaning of the composition it is possible to iterate this structure on N-levels

It is, also, possible to add classes for each level, that gives a context for each state recognized. For example for level 2 becomes: states = [sentence, locations of words in the sentence, context] obs = [ word, locations of words in the sentence, report]

## Getting started

Type HAI\_LANGUAGE\_pathsLoad; in main the directory to add necessary subpaths. then choose one of the main in directory "MAINS" to see a demonstration of code features

## Code

### Example in Simulation 1

Execute reading simulation (in folder simulation) to produce the outputs - four letters reading simulation: Sim1\_CM\_4l, Sim1\_DM1\_4l, Sim1\_DM2\_4l, Sim1\_DM\_4l - eight letters reading simulation: Sim1\_CM\_8l, Sim1\_DM1\_8l, Sim1\_DM2\_8l, Sim1\_DM\_8l Then execute Sim1\_Fig5a to produce Fig5(a) of the paper.

### (1) MAIN\_HAI\_DICTIONARY\_v0.m

HAI example with a simple dictionary 2-level hierarchy level 1 states = [ word, locations of letters in the word] obs = [ letter, locations of letters in the word] level 2 states = [sentence, locations of words in the sentence] obs = [ word, locations of words in the sentence]

DICTIONARY: Simple dictionary of 6 words composed by A B C D

### (2) MAIN\_HAI\_DICTIONARY\_v1.m

2-level hierarchy level 1 states = [ word, locations of letters in the word] obs = [ letter, locations of letters in the word] level 2 states = [sentence, locations of words in the sentence] obs = [ word, locations of words in the sentence]

DICTIONARY: Simple dictionary of English words (two syllable words of six letters)

### (3) MAIN\_HAI\_DICTIONARY\_v2.m

3-level hierarchy level 1 states = [syllable, locations of letter in the syllable] obs = [ letter, locations of letter in the syllable] level 2 states = [ word, locations of syllable in the sentence] obs = [syllable, locations of syllable in the sentence] level 2 states = [sentence, locations of word in the sentence] obs = [ word, locations of word in the sentence]

DICTIONARY: Simple dictionary of English words (two syllable words of six letters)

### (4) MAIN\_HAI\_DICTIONARY\_v3.m

2-level hierarchy level 1 states = [syllable, locations of letters in the syllable] obs = [ letter, locations of letters in the syllable] level 2 states = [ word, locations of syllable in the sentence] obs = [syllable, locations of syllable in the sentence]

DICTIONARY: Simple dictionary of English words (two syllable words of six letters)

## TRANSFORMES IN THE LOOP

### (5) MAIN\_HAI\_BERT\_LOOP\_s01.m,

three level structure, Dictionary provided by BERT

Read the produced BERT sentence 'THIS PAPER IS ALSO MENTIONED IN THE FAMOUS ENGLISH HISTORICAL NOVEL BY SIR ROBERT DE LA HAY'

level 1 states = [syllable, locations of letters in the syllable] obs = [ letter, locations of letters in the syllable] level 2 states = [ word, locations of syllables in the word] obs = [syllable, locations of syllables in the word] level 3 states = [sentence, locations of words in the sentence] obs = [ word, locations of words in the sentence]

### (6) MAIN\_HAI\_BERT\_LOOP\_s02.m,

Same as previous but given a context "We present a novel computational model that uses hierarchical active inference to simulate the reading process and eye movements during reading." read the BERT produced sentence:

THE COMPUTATIONAL MODEL IS ABLE TO PREDICT A TIME HORIZON FOR READING DURING A GIVEN TIME OR PLACE PERIOD

## (7) MAIN\_HAI\_BERT\_LOOP\_s03.m

Loop with BERT reading a sentence that has a word (BUTTER) that BERT does not provide

LOOP of HAI Code on a DICTIONARY predicted by BERT [https://it.mathworks.com/matlabcentral/fileexchange/107375-transformer-models?s\\_tid=FX\\_rc3\\_behav](https://it.mathworks.com/matlabcentral/fileexchange/107375-transformer-models?s_tid=FX_rc3_behav) add the corresponding path to use BERT model git clone <https://github.com/matlab-deep-learning/transformer-models>

## (8) main\_chatGPT\_SampleSentence.m

provided an API-KEY.txt for OPEN-AI chatGPT, given the same context of (6) "We present a novel computational model that uses hierarchical active inference to simulate the reading process and eye movements during reading."

produce a random sentence: e.g. THIS MODEL HAS BEEN DESIGNED TO ENABLE THE ACCOMMODATION OF A COMPREHENSIVE SET OF ADAPTIVE BEHAVIORS TO ACHIEVE BEST ACCURACY

## Suggested packages

1. matlab-tree package Package needed to enable tree visualization and computation on MDP <https://tinevez.github.io/matlab-tree/index.html>
2. spm12 statistical parametric mapping version 12 <https://www.fil.ion.ucl.ac.uk/spm/software/download/>
3. export\_fig enhanced routines for saving figures in MATLAB [https://github.com/altmany/export\\_fig](https://github.com/altmany/export_fig)
4. utilities <https://github.com/donnarumma/utilities/>

## Non exhaustive list of useful functions (in update...)

HAI\_disp -> print tree structure HAI\_compare -> compare two hierarchical structures

## Notes: Major differences between

VB\_MDP.m and spm12 spm\_MDP\_VB\_X.m

1) hidden states X are updated from t on. -Bayesian model averaging of hidden states- in spm\_MDP\_VB\_X.m the second cycle is from 1 to S in VB\_MDP.m it is possible to set it from t to S 2) this consequently means that in section - check for residual uncertainty (in hierarchical schemes) - the state on which the Entropy is computed in t and not 1

## Authors and acknowledgment

Francesco Donnaruma [francesco.donnarumma@istc.cnr.it](mailto:francesco.donnarumma@istc.cnr.it)

Mirco Frosolone [mirco.frosolone@istc.cnr.it](mailto:mirco.frosolone@istc.cnr.it)

Giovanni Pezzulo [giovanni.pezzulo@istc.cnr.it](mailto:giovanni.pezzulo@istc.cnr.it)

COgnition iN ActioN Laboratory (CONAN) <https://www.istc.cnr.it/it/group/conan-0>

## License

This code is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 1, or (at your option) any later version.

This code is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. A copy of the GNU General Public License can be obtained from the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.