Coding Manual:

Iterated Learning Project

Authors: Madeline Meyers, Harsh Singh, Daniel Yurovsky

*The University of Chicago*

**1 Transcript Overview**

Why do languages change? One possibility is they evolve in response to two competing pressures: (1) to be easily learned, and

(2) to be effective for communication. In a number of domains (e.g. kinship categories, color terms), variation in the world’s natural languages appears to be accounted for by different but near-optimal tradeoffs between these two pressures (Regier, Kemp, & Kay, 2015). Models of these evolutionary processes have used transmission chain paradigms in which errors of learning by one agent become the language input for the subsequent generation. However, a critical feature of human language is that children do not learn in isolation. Rather, they learn in communicative interactions with caregivers who can draw inferences from their errorful productions to their in-

tended interests. In a set of iterated learning experiments, we show that this supportive context can have a powerful stabilizing role in the development of artificial languages, allowing them to achieve higher levels of asymptotic complexity than they would by vertical transmission alone.

The spreadsheets have the following column heads:

**Chain:**

**Identifies experiment**

**Trial:**

**Identifies Puzzle**

**Generation:**

**Identifies attempt**

**No Relation:**

**Errors found are random**

**Trnsltion:**

**Errors found include a shifting of a block or chunk of blocks in the left, right, upward, or downward direction**

**For translations greater than 1 row/column shift,**

**Take the form: 1. #of rows/column shifted, 2. #rows/column shifted…**

**Rotation:**

**Errors found include a rotation of exactly 90, 180 or 270 degrees of a chunk of blocks**

**Mirroring:**

**Errors found include an exact flip of a chunk of blocks across a vertical or horizontal axis**

**Un-Chunking:**

**Errors found include a splitting or dispersion of blocks from a single chunk of blocks to simplify the overall design of the puzzle**

**Chunking:**

**Errors found include a unifying of several blocks together to form a single chunk of blocks to simplify the overall design of the puzzle**

**Diagonal Chunking:**

**Errors found include a unifying of several blocks together to form a diagonal chunk to simplify the overall design of the puzzle**

**For diagonal chunking greater than or equal to 3 blocks,**

**Take the form: 1. #of blocks in chunk, 2. #of blocks in chunk…**

**Edges & Corner:**

**Errors found include a pushing of blocks to either edges or corners to simplify the overall design of the puzzle**

**Shape Change:**

**Errors found include an alteration of a chunk of blocks’ shape to another particular shape**

**Iconicity:**

**Errors found include an intentional formation of a sign or symbol to simplify the overall design of the puzzle**

**Symmetry:**

**Errors found include symmetry along a point, block, or chunk of blocks to simplify the overall design of the puzzle**

**2 Coding Goals**

We are interested in identifying and describing every error found as the child attempts to copy from memory the target puzzle onto the blank grid.

**3 Potential Areas of Confusion**

Although it is tempting to code any rotation that occurs at exactly 180 degrees as a mirroring, refrain unless the entire chunk of blocks has rotated exactly 180 degrees about an axis or block.

A block in the shape of a staircase should be considered an instance of iconicity.

If diagonal chunking occurs it should not be marked twice for chunking and diagonal chunking

**4 Examples**