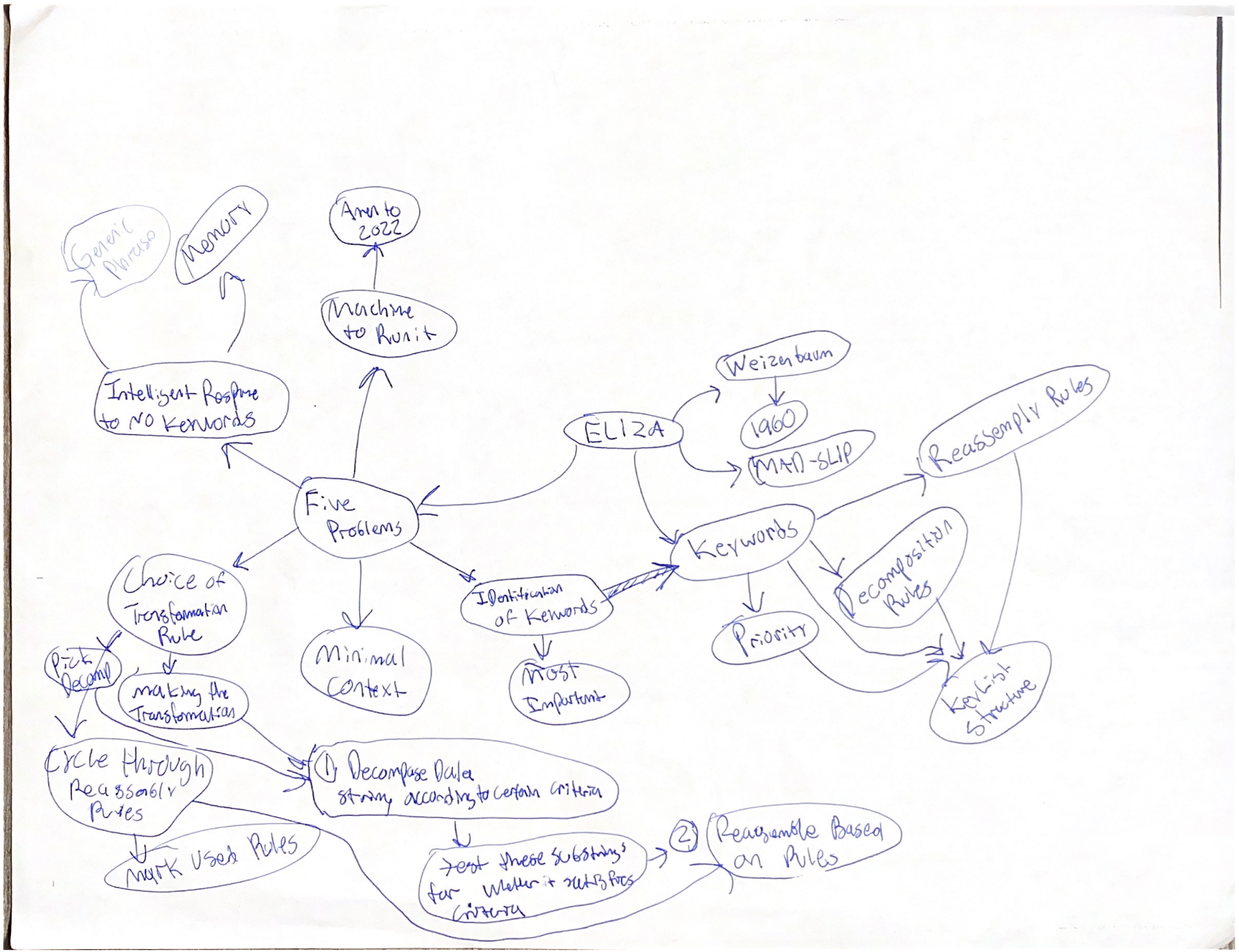
Generic Phrase
Memory
Intelligent Response to NO Keywords
Aren to 2022
Machine to Run it
Weizenbaum
1960
ELIZA
MAD-SLIP
Reassembly Rules
Five Problems
Keywords
Decomposition Rules
Choice of Transformation Rule
Identification of Keywords
Priority
Pick Decomp
making the Transformation
Minimal Context
Most Important
KeyList Structure
Cycle through Reassembly Rules
1) Decompose Data string according to certain criteria
mark used Rules
Test these substrings for Matches criteria
2) Reassemble Based on Rules

Eliza Draft 1 – Charles Cutler

Print welcome message
Loop {
Looping waiting for human input

Take in put message
scan input for keywords
substitute necessary keywords (ex I → you, am → are)
Place items into priority stack Based on keyword priority

Pop top of priority keyword stack
Locate keyword in Array
Identify acceptable Deconstruction Rule
Identify acceptable Reassembly Rule
Reassemble message
print to screen

If no suitable Rule
go to next keyword in keyword stack

If keyword stack empty
memory response

Generic Response

} endLoop

Missing in Draft 1
- Update highest priority found so far
- comma / period delimiter
  - when keyword already been found, update or delete all subsequent text
  - No key all text prior and delimiter deleted
- Key word List structure
  - 3 Dim Array ?
- Check Keyword is the actual word that was Hashed
  - Dictionary
- Substitution Rule to go to other keyword Rules (Choose your own Adventure)

Eliza draft 2 - charles Cotter

{ Establish keyword dictionary }

[Establish memory stack and Rank picked word stack]

Print welcome message
Loop {
  Take an input string message { Each time wait here till recieve a message
  Scan input message for { keywords, delimiters }
    for each word in message:
      Keywords → substitute
      Keywords → Add to keyword stack (Add to stack either top or Bottom)
      Update Highest Priority found (

if ker stack empty, if no keyword found before, or.
    Delete All stuff Before, or. and, or. delimiter
  if keyword found ~~before~~ before, or.
    Delete all stuff after, or.

Pull top of keyword stack
if (If empty stack
  ☆ memory response
  ☆ Generic response

else (Otherwise
  if (Identify ~~first acceptable~~ Decomstructution Rule (If acceptable)
    Identify acceptable Reassembly Rule
    Reassemble message
    print to screen (Break Back to Loop to wait for next input)
    Update R# for Decon Rule

Loop till found

  else (If not acceptable
    go to next Deconstruction Rule

Clear stacks Of priority, Clear Highest, Keep memory!
} End Loop

Attempt 3: A clearer dictionary

KEYWORDS = {

"WORD" : [Sub Rule opt., RANK, [Decomp 1, R#, R_1 - R_n], ... [Decomp$_n$, R#, R_i - R_n]]

"word" : "

"word" : "

:

:

:

}

Do sub Rules during keyword check?

store words in Rank stack? ⇒ cause Dictionary?

Eliza draft 3 - Charles Cutler

{Establish key word Dictionary}

[Establish keyword stack and memory stack]

Print welcome message

Loop ("Thank You Eliza!") {

Take input string message

Scan input message for delimiters and keywords

for each ~~key~~word / delimiter ;

if (is delimiter) and (key stack empty)

delete all stuff prior to and the delimiter, or

else if (is delimiter) and (key stack Not empty)

delete all stuff after delimiter

else (means it is a word)

if (word in keywords) (else just move to next)

keyword substitution

~~update highest priority word found~~

if (keywords Rank > Highest rank)

Update Highest rank found

Put on top of keyword stack

else {

Put on Bottom of keyword stack

if (keystack empty)

memory Response

Generic Response

else

Pull top of keystack

LOOP {

Check if key word Decomp 1 works

if not go to keyword Decomp2

Loop until Decomp Rule is found

}

Pick Reassembly Rule

Update RT

Reassemble sentence

Print to screen

Clear ~~Ram~~ Keystack, Highest priority, Keep Memory!

} End Loop

Changes the message for later parts of program

Updates

Updates

Updates

(New key) return if no Decomp.

None (0) some phrases

1) Sorry
Don't
can't
won't
5) Remember
If
Dreamt
Dreams
How ✓
10) when ✓
Alike ✓
Same ✓
Certainly
Feel ✓
15) Think ✓
Believe ✓
Wish ✓
Memory My ✓
None
20) maybe
Name
Deutchs
Francois
Italiano
25) Espanol
X FREMD
Hello
Computer
Machine
30) Machines
Computer
am ✓

your ✓
were ✓
35) Are ✓
You're ✓
I'm
Myself ✓
Yourself ✓
40) Mother ✓
Mom ✓
Dad ✓
Father ✓
Sister ✓
45) Brother ✓
Wife ✓
Children ✓
I ✓
You
No
50) My ✓
Can
What ✓
Because
55) Everyone ✓
Everybody ✓
Nobody ✓
Nonone ✓
Always
60) Dit ✓
61) Are ✓
Was ✓

Grandmother ✓
Grandma ✓
Grandfather ✓
Grandpa ✓
Husband ✓
Uncle ✓
Aunt ✓