

PS1

● Graded

Student

Chetan Hiremath

Total Points

98 / 100 pts

Question 1

What is Easy, What is Hard

15 / 15 pts

- | | | |
|--|------------|-----------|
| 1.1 | (no title) | 3 / 3 pts |
| <div>✓ + 3 pts Present, Complete, Reasonable</div> | | |
| 1.2 | (no title) | 3 / 3 pts |
| <div>✓ + 3 pts Present, Complete, Reasonable</div> | | |
| 1.3 | (no title) | 3 / 3 pts |
| <div>✓ + 3 pts Present, Complete, Reasonable</div> | | |
| 1.4 | (no title) | 3 / 3 pts |
| <div>✓ + 3 pts Present, Complete, Reasonable</div> | | |
| 1.5 | (no title) | 3 / 3 pts |
| <div>✓ + 3 pts Present, Complete, Reasonable</div> | | |

Question 2

What is Easy, What is Hard (Take 2)

15 / 15 pts

2.1 (no title) 3 / 3 pts

✓ + 3 pts Present, Complete, Reasonable

2.2 (no title) 3 / 3 pts

✓ + 3 pts Present, Complete, Reasonable

2.3 (no title) 3 / 3 pts

✓ + 3 pts Present, Complete, Reasonable

2.4 (no title) 3 / 3 pts

✓ + 3 pts Present, Complete, Reasonable

2.5 (no title) 3 / 3 pts

✓ + 3 pts Present, Complete, Reasonable

Question 3

The Turing Test

20 / 20 pts

3.1 (no title) 5 / 5 pts

✓ + 5 pts Correct

3.2 (no title) 5 / 5 pts

✓ + 5 pts Correct

3.3 (no title) 5 / 5 pts

✓ + 5 pts Correct

3.4 (no title) 5 / 5 pts

✓ + 5 pts Correct

Question 4

Shannon's Mind Reading Game

29 / 30 pts

4.1 (no title) 10 / 10 pts

✓ + 10 pts Both drawings present and correct

4.2 (no title) 15 / 15 pts

✓ + 15 pts Correct

4.3 (no title) 4 / 5 pts

✓ + 5 pts Correct

✓ - 1 pt point is missing

Question 5

Babbling with a MC Model

19 / 20 pts

5.1 (no title) 5 / 5 pts

✓ + 5 pts Correct

5.2 (no title) 5 / 5 pts

✓ + 5 pts Correct

5.3 (no title) 5 / 5 pts

✓ + 5 pts Correct

5.4 (no title) 4 / 5 pts

✓ + 5 pts Correct

✓ - 1 pt Wrong Conclusion

Q1 What is Easy, What is Hard

15 Points

FIRST: Note that this Problem Set has a [FAQ](#) that addresses textbox answers.

PURPOSE: Start to think about what is easy and what is hard for computers to do and algorithms/techniques that might be used to solve different problems.

Re-order these popular NYT games by how easy you believe it would be for an AI to do (easiest to hardest), and add a couple sentences (i) justifying that and (ii) also saying how you might construct a program/system to solve each one:

- Spelling Bee
- Wordle
- Connections
- Letter Boxed
- Sudoku

If you unfamiliar with any of these games, check out screenshots of rules and puzzles in the [FAQ](#) that addresses textbox answers. Else, a simple web search should supply the necessary details.

Q1.1

3 Points

Easiest

Wordle is the easiest game that will be solved by AI because it finds hidden words by using the process of elimination, testing different word combinations, and learning the feedback of the previous guess until it guesses the correct word. I can construct a program/system by generating guesses with English letter frequencies and combinations and refining them with valuable feedback until the word is guessed correctly in a certain number of tries or attempts because it is a very simple and straightforward task.

Q1.2**3 Points**

Next Easiest

Spelling Bee is the next easiest game that will be solved by AI because it can generate and evaluate words that have the required and the specific letters by word frequency, dictionary validity, and a vast language model until it finds the right word. I can construct a program/system by generating various combinations of letters and using a dictionary, which contains many words, to create and identify proper words.

Q1.3**3 Points**

Middle

Sudoku is the middle game that will be solved by AI because it is a completely different game that requires logical number placement and doesn't repeat any number in any row or column by using logical rules and constraints to make sure that the grids are solved without the repetition of numbers. I can construct a program/system by backtracking the numbers that can efficiently solve these puzzles and fill the numbers iteratively due to the constraints until the rows and the columns of the Sudoku grids have no repeated numbers.

Q1.4**3 Points**

Next Hardest

Letter Boxed is the next hardest game that will be solved by AI because it requires the formation of words that connect and loop around sets of letters in squares by utilizing word manipulation and guessing letters until these letters form proper words. I can construct a program/system by using spatial reasoning, which will traverse possible paths and make sure that the words are connected by the letters, and ignoring repetitions that won't have letters from the same sides.

Q1.5

3 Points

Hardest

Connections is the hardest game that will be solved by AI because it connects letters to form words and requires a great understanding of semantics by using semantic networks and knowledge graphs to establish shared groups among words until the words are included in their respective groups with different levels of difficulty correctly. I can construct a program/system by including NLP/natural language processing and semantic analysis because they will allow me to guess, find, and include words in their respective groups correctly.

Q2 What is Easy, What is Hard (Take 2)

15 Points

PURPOSE: Start to think about what is easy and what is hard for computers to do and how that might shape the future economy.

Re-order these professions by how much AI could disrupt/displace them and say one to two sentences justifying your choice by explicitly stating how each one would be affected or pieces of the job that would be hard for AI to replace.

- customer service provider
- copy editor
- journalist
- computer programmer
- truck driver

Q2.1

3 Points

Easiest to Disrupt/Displace

Copy Editor is the easiest profession that will be replaced by AI because this profession involves fine-tuning language, tone, and style that are accomplished by human creativity and understanding of the context. AI can assist in grammar, spelling checks, automating proofreading tasks, and style adherence. But human creativity, nuanced language understanding, and the ability to grasp contextual preciseness will be challenging for AI to replace the profession of copy editor.

Q2.2**3 Points**

Next Easiest to Disrupt/Displace

Journalist is the next easiest profession that will be replaced by AI because this profession requires critical thinking, investigation, and storytelling to write various articles and stories. AI can help with data analysis and generate some news content. But the ethical considerations, human connection, investigative journalism, interview subjects, complex information, and contextual understanding that require human skills in journalism won't be automated by AI.

Q2.3**3 Points**

Middle to Disrupt/Displace

Computer Programmer is the middle profession that will be replaced by AI because this profession uses programming skills to write and test code for results and observations. AI automates certain aspects and tasks of programming and some coding tools. But the creativity, problem-solving skills, and decision-making skills that are required for innovative programming solutions are areas that are suitable for human expertise that easily outperforms AI.

Q2.4**3 Points**

Next Hardest to Disrupt/Displace

Customer Service Provider is the next hardest profession that will be replaced by AI because this profession handles routine customer queries and answers questions from various customers. AI can perform some queries with the help of chatbots and virtual assistants. But the empathy, emotional intelligence, and nuanced understanding that are required for complicated customer interactions are challenging for AI to respond effectively. Therefore, human touch is very important in customer service.

Q2.5

3 Points

Hardest to Disrupt/Displace

Truck Driver is the hardest profession that will be replaced by AI because this profession involves complex real-world decision-making, adaptability to unpredictable situations, and diverse/strong physical skills. Autonomous vehicles that use AI are advanced. But AI faces challenges in responding to unpredictable road conditions, navigating diverse terrains, and ensuring public safety in a wide range of scenarios. So, the replacement of human truck drivers by AI will not be a wise idea.

Q3 The Turing Test

20 Points

PURPOSE: *Understand the Turing Test and the current state-of-the-art in LLMs in that context.*

Read Turing's paper Turing's "[Computing Machinery and Intelligence](#)". Also, read [Donald Knuth's writeup](#).

Conduct your own experiment, where you ask Chat GPT a series of questions. Assess its answers with respect to whether you believe it has passed the Turing test or not. Give a detailed

(i) prompt,

(ii) response, and

(iii) analysis

similar to what Knuth did but **only for 4 questions**, each in a separate box below.

Q3.1

5 Points

Your Question 1 Prompt, Response, and Analysis

Question 1

Prompt- I have asked Chat GPT to give me 3 ideas to plan good New Year's resolutions like personal resolutions, family resolutions, and professional resolutions. (Question- Give me 3 ideas about how to plan good New Year's resolutions. Give me some that are personal, family, and professionally oriented.)

Response- Here is Chat GPT's generated response:

Certainly! Here are three ideas for New Year's resolutions in the personal, family, and professional domains:

Personal Resolution: Cultivate Mindfulness- Set a goal to incorporate mindfulness practices into your daily routine. This could include meditation, deep breathing exercises, or simply taking moments throughout the day to be fully present and engaged in the current moment. Cultivating mindfulness can enhance your overall well-being and help manage stress.

Family Resolution: Quality Time Commitment- Resolve to spend more quality time with your family. This could involve planning regular family activities, such as game nights, movie nights, or outdoor outings. Setting aside dedicated time for family bonding can strengthen relationships, foster communication, and create lasting memories.

Professional Resolution: Skill Development- Identify a specific skill or set of skills that would enhance your professional growth. This could be learning a new programming language, improving your project management abilities, or developing better communication skills. Invest time in professional development, whether through online courses, workshops, or networking events, to boost your career prospects.

Remember to make your resolutions specific, measurable, achievable, relevant, and time-bound (SMART) to increase the likelihood of success. Tailor them to your personal preferences and circumstances, and don't hesitate to adjust them throughout the year based on your progress and changing priorities.

Analysis- This response proves that Chat GPT can provide good ideas about the

resolutions that will be helpful to me. Because it has gathered data from humans before it has given me ideas for my resolutions. It generates responses that are very similar to the responses of regular humans, who know resolutions. It tells me to have different ways of exercising for mindfulness, spend quality time with family, and learn and gain new skills. It is fascinating that Chat GPT has provided additional tips and ideas that won't hesitate me to make progress during this year, so it passes the Turing Test.

Q3.2

5 Points

Your Question 2 Prompt, Response, and Analysis

Question 2

Prompt- I have asked Chat GPT to know its thoughts about this year's Super Bowl's winning football team. (Question- Who do you want to win this year's Super Bowl?)

Response- Here is Chat GPT's generated response:

I don't have personal preferences or opinions. My purpose is to provide information and assistance. If you have any questions about the Super Bowl or related topics, feel free to ask!

Analysis- This response proves that Chat GPT doesn't know the answer of this question because it doesn't predict the future of the winner of the Super Bowl. Also, it doesn't have its personal preferences and opinions since it is not a human. If I ask this question to a human, then that human will share a possible choice of the winner of the Super Bowl. But an AI chatbot like Chat GPT won't have its perspectives because it provides information and data to me. It doesn't have its preferences, so it won't share its personal thoughts with me and doesn't pass the Turing Test.

Q3.3

5 Points

Your Question 3 Prompt, Response, and Analysis

Question 3

Prompt- I have asked Chat GPT to tell me the easiest major of any college.
(Question- What is the easiest major in college?)

Response- Here is Chat GPT's generated response:

The perception of what constitutes the "easiest" major can vary among individuals, as it depends on personal strengths, interests, and aptitudes. What might be easy for one person could be challenging for another. Additionally, difficulty levels can differ between universities and programs.

Some people may find majors in the humanities or social sciences to be more accessible, while others might excel in fields like business or natural sciences. It's essential to choose a major based on your interests and career goals rather than solely on perceived ease.

If you have a specific field or area of study in mind, I can provide more information tailored to that.

Analysis- This response proves that Chat GPT provides some details about the easiest majors to me even though choosing the easiest majors depends on the individual's choices. It can offer me information on the easiest majors that are recommended by many people, so it basically talks about the majority. It generates and sends information that is provided by humans, so the response mainly depends on the humans' preferences because they have experiences of different majors. Chat GPT provides that information according to humans, so it passes the Turing Test.

Q3.4

5 Points

Your Question 4 Prompt, Response, and Analysis

Question 4

Prompt- I have asked Chat GPT to recognize the most impossible task that won't be performed by any humans. (Question- What is the most impossible task that won't be performed by humans?)

Response- Here is Chat GPT's generated response:

It's challenging to pinpoint the single most impossible task that won't be performed by humans, as the concept of impossibility can be subjective and may evolve over time with technological advancements. However, certain tasks, such as violating fundamental laws of physics (like perpetual motion machines of the first kind) or achieving true time travel in a way that doesn't violate known principles, are often considered extremely difficult or currently impossible based on our current understanding of the universe. Keep in mind that advancements in science and technology may change our perspectives on what is considered impossible in the future.

Analysis- This response proves that Chat GPT can answer this question properly because it is very impossible for humans to violate fundamental laws of physics or create time traveling. It is scientifically true because many people can't harm or alter science that has been proven by many scientists over several years. I have watched time travel in movies, but it is impossible to achieve time travel in real life. Chat GPT can provide information from the internet that has a vast number of resources, so it gives me the answer that is provided from the internet and passes the Turing Test.

Q4 Shannon's Mind Reading Game

30 Points

Note that this problem has a [FAQ](#)

PURPOSE: Understand Markov chains as a means to model/predict probabilistic processes in a simple case.

- a. Toss a coin 32 times and record the outcome (as a string of H and T).
- b. Compute the experimentally observed probability of heads over tosses 2 through 21, inclusive (20 outcomes).
- c. Compute the Markov chain transition probabilities over the first 21 tosses (viz., the first 20 transitions).
- d. Repeat (a)-(c) above with a sequence verbally derived from a friend not in the class who does not know the underlying model you are trying to construct.
- e. Using any language you wish, implement a computer program that uses the Markov chain model to predict the final ten transitions of each data set (throws 23-32 given the values of throws 22-31, respectively). That is, given the previous state (throw i), compute the next state (throw $i+1$) using the model, compare with the actual data, and tally the error function (# of wrong guesses).

Q4.1

10 Points

Upload a file containing (i) your flip data and (ii) the drawings for BOTH of your Markov chains (MCs) here (5 points each data/MC)

▼ Scannable Document.png
Download

Problem Set 2

1. T H H T H T T H T T H H H H H T T T T T H H H H H H H H

2. H H H T H T T H T T T H T H H H H T T T

$P(\text{Heads}) = \frac{\text{Number of heads}}{\text{Number of outcomes}} = \frac{10}{20} = \frac{1}{2} = 0.5 = 50\%$

3. HT = 5 HH = 5 TT = 5 TH = 5

5/100 (50%) (H) $\xleftrightarrow{5/100 (50\%)} (T) \xleftrightarrow{5/100 (50\%)} (H)$

4. T H T H T T H H T H H T H H T T T H T H T H H T T H T H

5. H T H T T H H T H H T H H T T T H

$P(\text{Heads}) = \frac{\text{Number of heads}}{\text{Number of outcomes}} = \frac{10}{20} = \frac{1}{2} = 0.5 = 50\%$

6. HT = 6 HH = 3 TT = 4 TH = 7

3/9 (33%) (H) $\xleftrightarrow{6/9 (67\%)} (T) \xleftrightarrow{7/11 (64\%)} (H)$

(click to upload a jpg -- no need for high res! -- or a pdf scan)

Q4.2

15 Points

Turn in a cut-and-paste of your code and any input/output pairs used in testing here

Python Code-

#Source- This code is modified, used, and borrowed from EECS 649: Lec 2 - The Turing Test's Simulated Luenberger's Weather Model's Python Code.

```
import random
```

```
#States are Heads (H) and Tails (T).
```

```
def func(set, state, HT, TH):
```

```
    incorrect = 0
```

```
    for i in range(10):
```

```
        r = random.random() #Pick a random number from 0 to 1.
```

```
        if (state == "H"):
```

```
            if r <= HT: #HT probability state changes to tails. If not, then the state is unchanged.
```

```
                nextstate = "T"
```

```
            else:
```

```
                nextstate = "H"
```

```
        elif (state == "T"):
```

```
            if r <= TH: #TH probability state changes to heads. If not, then the state is unchanged.
```

```
                nextstate = "H"
```

```
            else:
```

```
                nextstate = "T"
```

```
            if nextstate != set[i]: #The number of wrong guesses is recorded if states and data set's states are not same.
```

```
                incorrect+=1
```

```
            state = nextstate
```

```
            if state == "H": #The state H is added in the string of H and T if the condition is true.
```

```
                print("H", end=' ')
```

```
            elif state == "T": #The state T is added in the string of H and T if the condition is true.
```

```
                print("T", end=' ')
```

```
            print("")
```

```
            print(incorrect, "wrong guesses") #Tally the number of wrong guesses.
```

```
print("")
```

```
actual_firstset = ["T", "H", "H", "H", "H", "T", "H", "H", "T", "H"] #The last 10  
transitions and their values of the first data set.
```

```
actual_secondset = ["H", "T", "T", "H", "H", "T", "T", "H", "T", "H"] #The last 10  
transitions and their values of the second data set.
```

```
initial_state = "H" #The initial state or H.
```

```
print(*actual_firstset) #The first data set.
```

```
output = func(actual_firstset, initial_state, 0.5, 0.5) #The output for the first data  
set. (HT Probability = 0.5, and TH Probability = 0.5)
```

```
print(*actual_secondset) #The second data set.
```

```
output2 = func(actual_secondset, initial_state, 0.7, 0.6) #The output for the  
second data set. (HT Probability = 0.7, and TH Probability = 0.6)
```

Q4.3

5 Points

Comment on your results here

Result-

T H H H H T H H T H

T T T H T H H H H T

6 wrong guesses

H T T H H T T H T H

T H H T H H H T H H

8 wrong guesses

Here is the result of this Python Code. The first line of the result is the actual data of the first set of the last 10 transitions, the second line is the predicted data that is generated by the code, and the third line is the number of wrong guesses when the program has compared the actual data and the predicted data of the first set. If the actual data's state and the predicted data's state are different, then the difference is considered as a wrong guess that will be recorded. The other 3 lines follow the same procedure when the second set is used. If you use this code to generate other results, then the results will be completely random and different due to the random library of the Python Code. But the results will properly count the respective numbers of wrong guesses by any input pairs.

Q5 Babbling with a MC Model

20 Points

PURPOSE: *Further understand Markov chains as a means to model/predict probabilistic processes in a more complicated case more similar to how LLMs like chat-GPT work.*

Experiment with this [Star Wars GPT Python Notebook](#).

Q5.1

5 Points

Explain in your own words how the `build` and `babble` functions work

Build Function- The build function builds the dictionary and removes the upper-case letters in the corpus text by using the tokenize functions. Its parameters/inputs are `pred_length` that defines the ORDER and `punctuation_mode` that defines the MODE. The `pred_length` is controlled by the for loop from the range of 1 to 6, and the punctuation mode is "tokenize" that will tokenize the text and use the tokenize function that removes punctuation and converts uppercase letters into lowercase letters for the corpus text. Its result/output is the corpus text with lower-case letters that will be used for the babble function that will be explained in the next paragraph since the babble function is an important function that is incomplete without the build function and will provide the final output of this program.

Babble Function- This babble function produces n characters and tokenizes the corpus text before it provides the output. Its parameters/inputs are `seed_string` that defines the SEED or the input text, `pred_length` that defines the ORDER, `punctuation_mode` that defines the MODE, and `maxlen` that defines the maximum length. The `pred_length` is controlled by the for loop from the range of 1 to 6, and the punctuation mode is "tokenize" that will tokenize the text and use the SEED string and the tokenize function that removes punctuation and converts uppercase letters into lowercase letters for the corpus text. Its data is stored in a seed variable that will print the output if the length of the variable is less than the maxlen value. Its result/output is the corpus text with lowercase letters that will be displayed on the terminal.

Q5.2**5 Points**

Comment on how the order of the model affects its performance.

The order of the model affects its performance since the corpus text with lowercase letters is changed when the order changes or increases. The corpus text mainly depends on the order and number of tokens, which will be tokenized. If the number of tokens changes, then the corpus text will be completely different. Also, the performance is affected by the random library that will convert the random number of tokens into lowercase letters for the corpus text. Therefore, you will get accurate results with the random number of tokens and orders.

Q5.3**5 Points**

Comment on the expected (relative) performance of a model that used letters (instead of words) for tokens, and how text completion for say search and also chat-GPT's performance would be different if they had been trained on letters as tokens instead of words.

The expected/relative performance of the model that uses letters for tokens is that this program uses the random library and various functions that change the corpus text and its letters that will be converted from uppercase into lowercase letters. If the text completion and Chat-GPT are trained on letters as tokens, then their respective performances will be very different because using letters as tokens will have effective implications for capturing semantic and contextual information. Text generation can produce textual content, but Chat-GPT can produce accurate results since it understands the natural language understanding. You can train the model with letters as tokens to generate data and accurate text, but words are more suitable than letters generally for model training since words have semantic meanings.

Q5.4

5 Points

Comment on the difficulty of text generation being perceived as correct versus a computer program or mathematical proof.

The difficulty of the text generation that is perceived as a correct text and the computer program or a mathematical proof will have several differences because the text generation and the computer program/mathematical proof will provide different results, not same results. The text generation can provide contextually textual content, but it lacks true understanding and might produce incoherent information since it doesn't act like a human. Also, it mainly depends on the language comprehension and context. Computer program or mathematical proof is operated by rules and logical conditions, so their results are accurate. But text generation involves predictions that depend on correctness due to textual patterns. Therefore, language uncertainty and mimicking human information will be challenging for text generation.