AI INSTITUTE #AIISC
UNIVERSITY OF SOUTH CAROLINA

UNIVERSITY OF SOUTH CAROLINA

CSCE 580: Introduction to AI
CSCE 581: Trusted AI

Lecture 16: Machine Learning – Trust Issues

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

17TH OCT 2023

Carolinian Creed: "I will practice personal and academic integrity."

Credits: **Copyrights of all material reused acknowledged**

Organization of Lecture 16

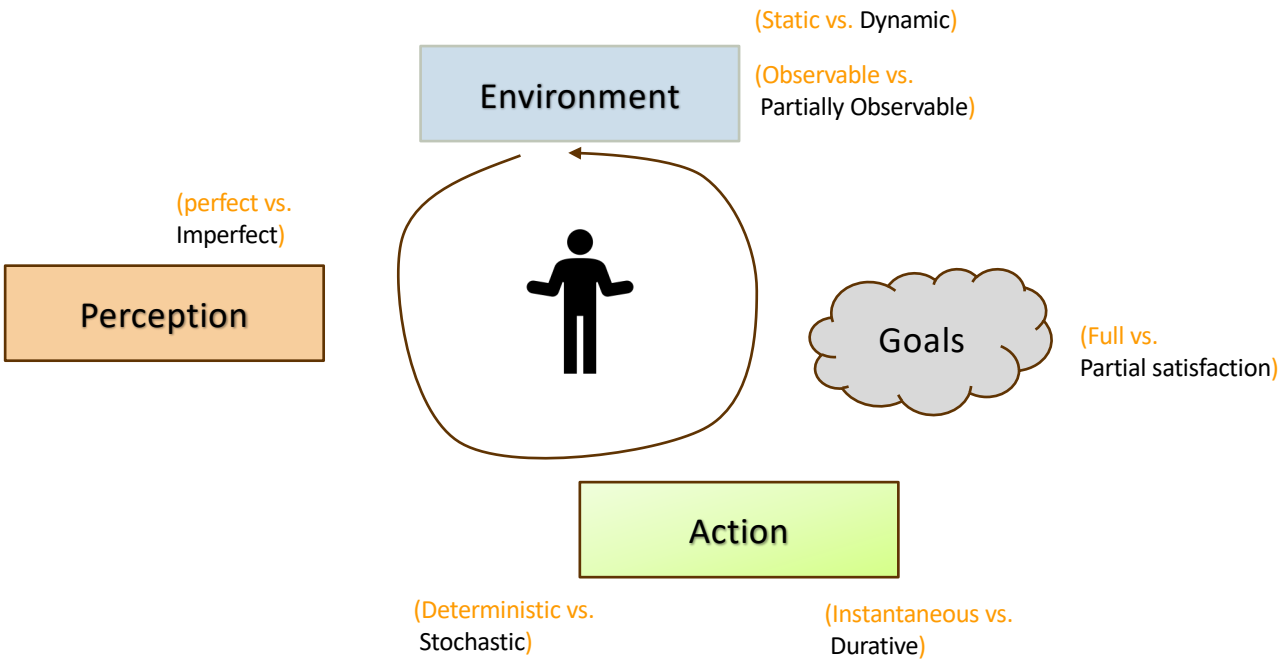
- Introduction Segment
  - Recap of Lecture 15
- Main Segment
  - Finish (Sprint 1) Project Presentations
  - Trust Issues
    - Explainability
  - LIME tool
- Concluding Segment
  - Course Project Discussion
  - About Next Lecture – Lecture 17
  - Ask me anything

Introduction Section

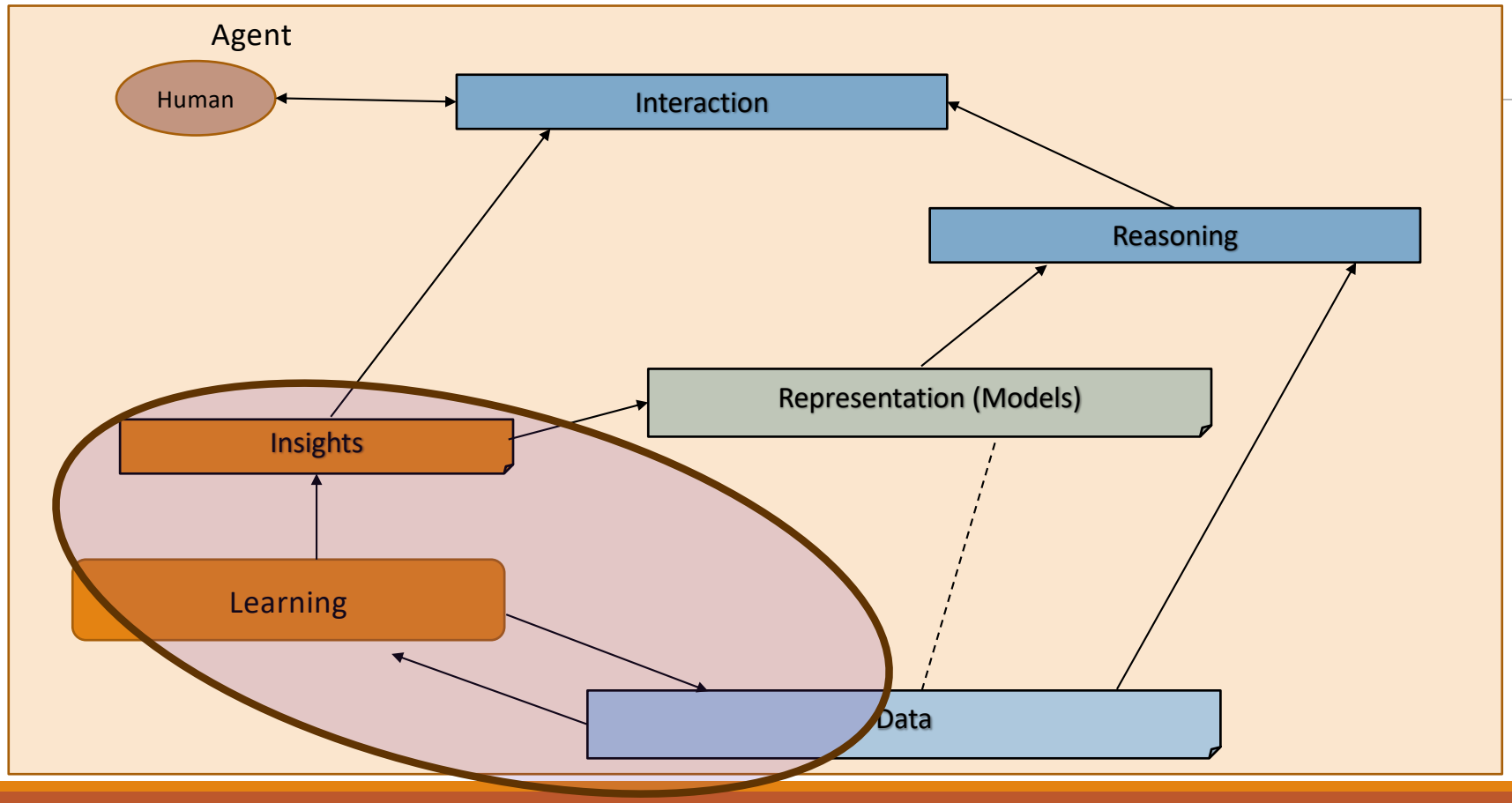
Recap of Lecture 15

- Topic discussed
  - Student presentations on Sprint 1 – to be completed

Intelligent Agent Model



Relationship Between Main AI Topics



Where We Are in the Course

CSCE 580/ 581 – In This Course

- Week 1: Introduction, Aim: Chatbot / Intelligence Agent
- Weeks 2-3: Data: Formats, Representation and the Trust Problem
- Week 4-5: Search, Heuristics - Decision Making
- Week 6: Constraints, Optimization – Decision Making
- Week 7: Classical Machine Learning – Decision Making, Explanation
- Week 8: Machine Learning - Classification
- Week 9: Machine Learning - Classification – Trust Issues and Mitigation Methods
- Topic 10: Learning neural network, deep learning, Adversarial attacks
- Week 11: Large Language Models – Representation, Issues
- Topic 12: Markov Decision Processes, Hidden Markov models - Decision making
- Topic 13: Planning, Reinforcement Learning – Sequential decision making
- Week 14: AI for Real World: Tools, Emerging Standards and Laws; Safe AI/ Chatbots

Main Section

Credit: Retrieved from internet

What are the Components of Trust (Technology)

1. Competent – does what it is supposed to do
2. Reliable – including, well tested
3. Upholds human values, social good
    1. Fairly and ethically used
    2. Adequate data management & preserves privacy
4. Allows human-technology interaction
    1. Explainable, transparent
    2. How does the system give its result?

Components of Trust - Illustration

1. Competent – does what it is supposed to do
2. Reliable – including, well tested
3. Upholds human values
    1. Fairly and ethically used
    2. Adequate data management & preserves privacy
4. Allows human-technology interaction
    1. Explainable, transparent
    2. How does the system give its result?

	Car – cruise control	Nuclear Energy
Competent	X	X
Reliable	X	X
Upholds human values	-	?
Allows human interaction	X	-

x: yes; -: not applicable; ?: questionable

Components of Trust for AI

1. Competent – does what it is supposed to do
2. Reliable – including, well tested
3. Upholds human values
   1. Fairly and ethically used
   2. Adequate data management & preserves privacy
4. Allows human-technology interaction
   1. Explainable, transparent
   2. How does the system give its result?

	AI – Word Tag Cloud	AI – Image Search	AI – Self-driving Car	AI-powered Chatbot: Medical Guide
Competent	x	x	?	x
Reliable	x	?	?	?
Upholds human values	?	?	?	?
Allows human interaction	x	x	?	?

x: yes; -: not applicable; ?: questionable

Group Discussion (5 mins): What are the Components of Trust for People

1. Examples
    1. Competence – do they get work done?
    2. Reliable – are they consistent?
    3. Uphold human values
    4. …
2. Are the components for people different from those for technology?

Main AI Ethics Issues

DATA GOVERNANCE AND PRIVACY

FAIRNESS AND INCLUSION

HUMAN AND MORAL AGENCY

VALUE ALIGNMENT

ACCOUNTABILITY

TRANSPARENCY AND EXPLAINABILITY

TECHNOLOGY MISUSE

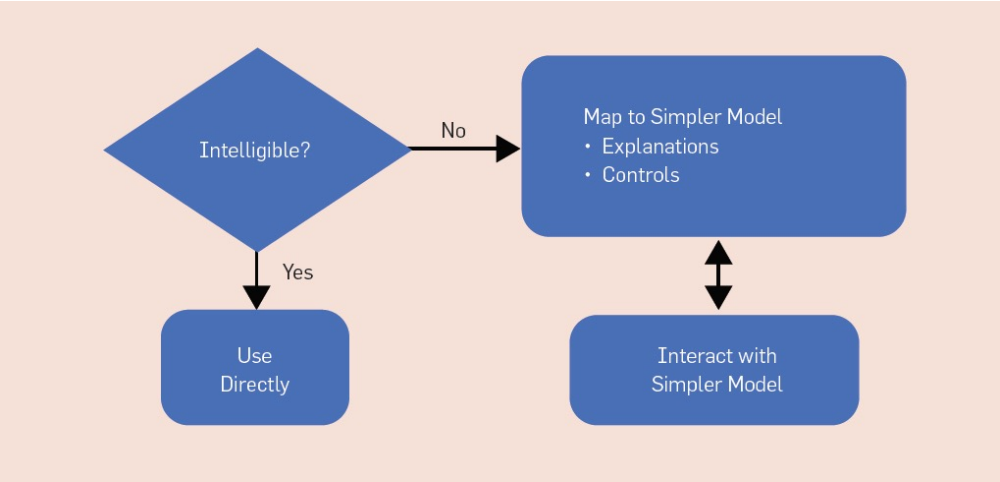
Credits:
Tutorial on Trusting AI by Testing and Rating Third Party Offerings at IJCAI 2020, Biplav Srivastava, Francesca Rossi, Jan 2021

Generating Explanations

What is the Purpose of Explanations

- Explanation and understanding
  - Frank C Keil, https://pubmed.ncbi.nlm.nih.gov/16318595/
- Purposes for explanations in **psychology**
  - To predict similar events in the future: *slippery roads can cause a fall*. Use information later.
  - For diagnosis: *why a system failed and then repair a part to bring it back to its normal function*
  - To affix blame: *for a crime*
  - To justify or rationalize an action: *sweet to an enemy because of the strategic value of being nice on that occasion*
  - In the service of aesthetic pleasure

Setting and Terminology: Intelligible Models and Explanations

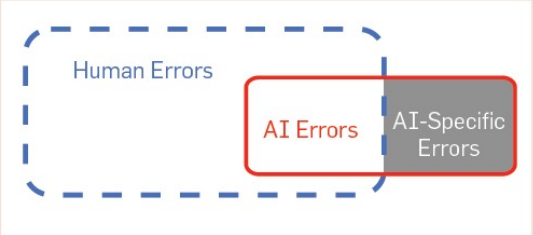


- Transparency: providing stakeholders with relevant information about how a model works
- Explainability: Providing insights into model's behavior for specific datapoints

Sources:
1. The Challenge of Crafting Intelligible Intelligence, Daniel S. Weld, Gagan Bansal, Communications of the ACM, June 2019, Vol. 62 No. 6, Pages 70-79, 10.1145/3282486
2. Explainable Machine Learning in Deployment, FAT* 2020.

Need for Intelligibility

The red shape denotes the AI's mistakes; its smaller size indicates a net reduction in the number of errors. The gray region denotes AI-specific mistakes a human would never make. Despite reducing the total number of errors, a deployed model may create new areas of liability (gray), necessitating explanations.



- **AI may have the wrong objective:** is AI right for the right reasons?
- **AI may be using inadequate features:** understand modeling issues
- **Distributional drift:** detect when and why models are failing to generalize
- **Facilitating user control:** guiding what preferences to learn
- **User acceptance:** especially for costly actions
- **Improving human insight:** improve algorithm design
- **Legal imperatives**

Source: The Challenge of Crafting Intelligible Intelligence, Daniel S. Weld, Gagan Bansal, Communications of the ACM, June 2019, Vol. 62 No. 6, Pages 70-79, 10.1145/3282486

Types of Explanations

- **Feature-based**: from the features of the data, which feature(s) were most important for given decision output
  - Example: For a loan, is it income or the person's age ?
- **Sample-based**: from data in training, which data points were important for given test point; helps understand sampling and its representation in wider population
  - Example: For a loan, what instances similar to the loan application would have gotten the loan ?
- **Counter-factual**: what-ifs – what do you change about the input to change the decision output
  - Example: For a loan, does getting an additional borrower insurance increase chance of getting the loan?
- Natural language

Source: Explainable Machine Learning in Deployment, FAT* 2020

LIME – Local Interpretable Model-Agnostic Explanations

Paper: "Why Should I Trust You?" Explaining the Predictions of Any Classifier, Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin, ACM's Conference on Knowledge Discovery and Data Mining, KDD2016

Blogs:

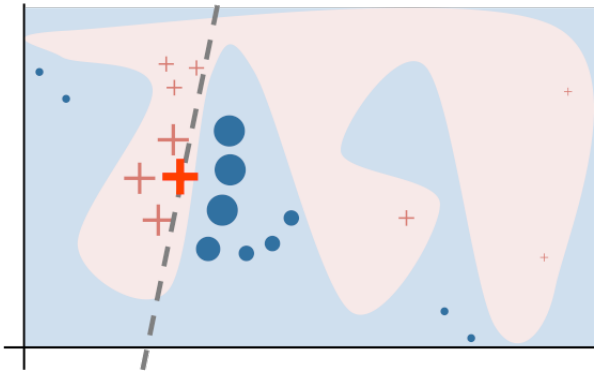
- https://homes.cs.washington.edu/~marcotcr/blog/lime/
- https://www.oreilly.com/content/introduction-to-local-interpretable-model-agnostic-explanations-lime/

Code: https://github.com/marcotcr/lime

Figures credit: Marco Tulio Ribeiro

LIME Key Idea

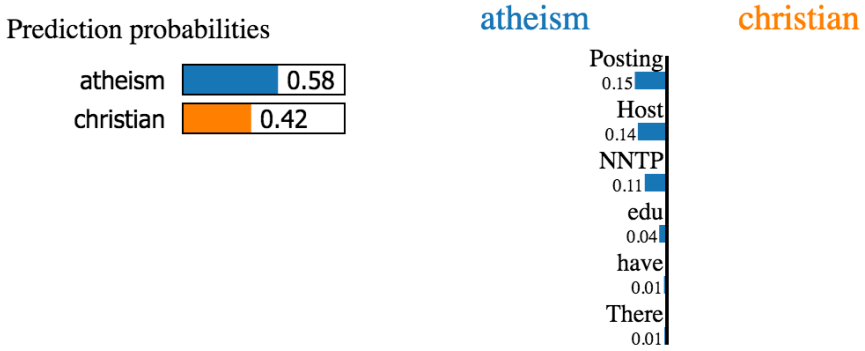
- Generate a local, linear explanation for any model
- How
  - Perturb near the neighborhood of a point of interest, X (**Local**)
  - Fit a linear function to the model's output (**Linear**)
  - Interpret coefficients of the linear function (**Explain**)
  - **Visualize**
- Applicability
  - Any classification model!



LIME on Text

Question: Why is a classifier with >90% accuracy predicting based on ?

Task: classifying religious inclination from email text



Text with highlighted words

From: johnchad@triton.unm.edu (jchadwic)
Subject: Another request for Darwin Fish
Organization: University of New Mexico, Albuquerque
Lines: 11
NNTP-Posting-Host: triton.unm.edu

Hello Gang,

There have been some notes recently asking where to obtain the DARWIN fish.
This is the same question I have and I have not seen an answer on the
net. If anyone has a contact please post on the net or email me.

"If we remove the words **Host** and **NNTP** from the document, we expect the classifier to predict **atheism** with probability 0.58 - 0.14 - 0.11 = 0.31"

Source: https://github.com/marcotcr/lime

Code Examples for Tabular Data

- LIME
  - Iris dataset and supervised classifiers – random forest and logistic regression, tabular data: https://github.com/biplav-s/course-tai/blob/main/sample-code/l9-explanations/LIME%20explanations%20on%20tabular%20data.ipynb

- Many other examples
  - https://github.com/biplav-s/course-d2d-ai/tree/main/sample-code/l12-explanability-autoai

LIME on Image

Question: Why is this a frog?

Divide image into interpretable components - contiguous superpixels

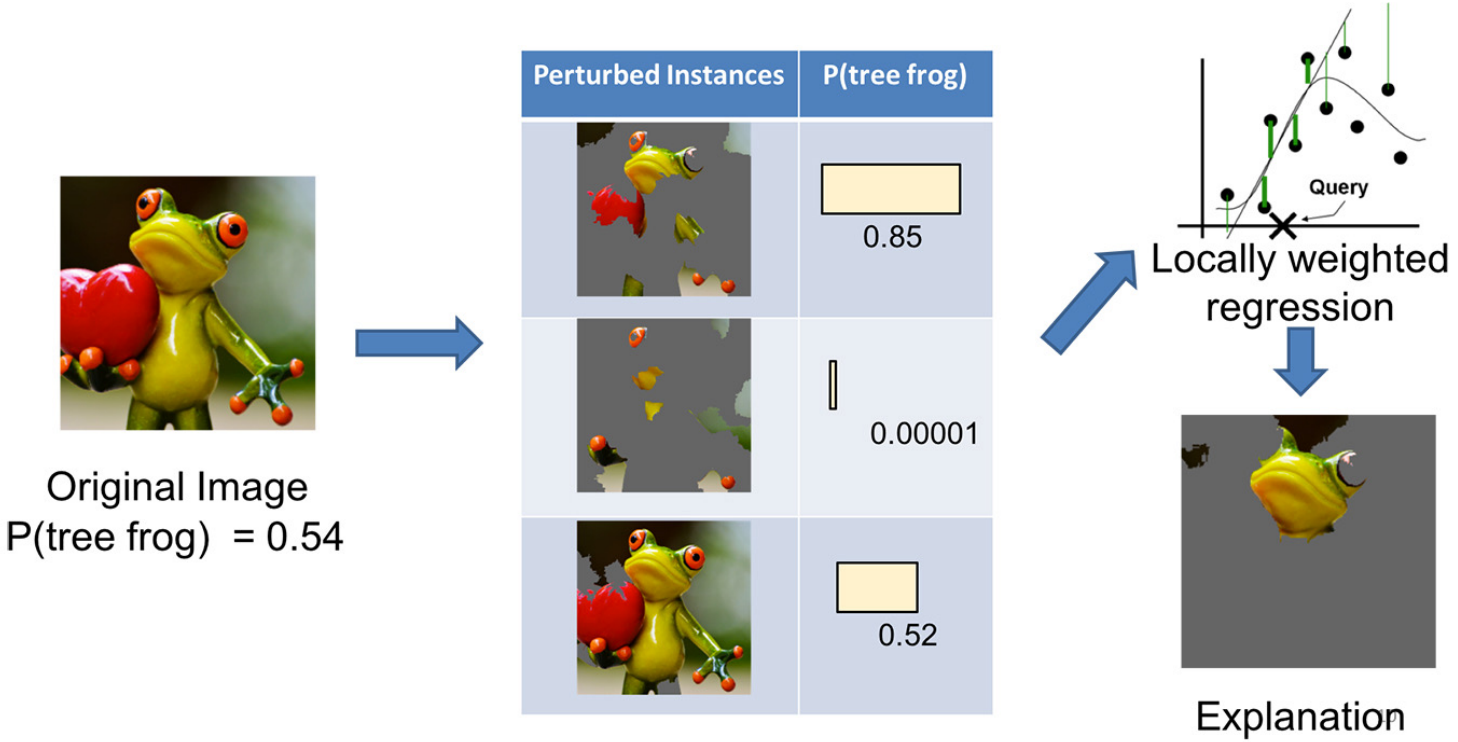
Original Image

Interpretable Components

Source: https://www.oreilly.com/content/introduction-to-local-interpretable-model-agnostic-explanations-lime/

LIME

1. Generate a data set of perturbed instances by turning some of the interpretable components “off” (gray).
2. For each perturbed instance, calculate probability that a tree frog is in the image according to the model.
3. Learn a simple (linear) model on this data set, which is locally weighted
4. Output regions with highest positive weights as an explanation, graying out everything else.



Source: https://www.oreilly.com/content/introduction-to-local-interpretable-model-agnostic-explanations-lime/

Explanation and Practical Implications

- Context
  - Problem: detect common cardiovascular conditions
  - Data: ECG data
  - Explanation: LIME
- References
  - Blog: https://www.ucsf.edu/news/2021/08/421301/ai-algorithm-matches-cardiologists-expertise-while-explaining-its-decisions
  - Paper: https://jamanetwork.com/journals/jamacardiology/article-abstract/2782549

References for AI Explainability

Papers

- The Challenge of Crafting Intelligible Intelligence, Daniel S. Weld, Gagan Bansal, Communications of the ACM, June 2019, Vol. 62 No. 6, Pages 70-79, 10.1145/3282486
- "Why Should I Trust You?" Explaining the Predictions of Any Classifier, Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin, in ACM's Conference on Knowledge Discovery and Data Mining, KDD2016; https://homes.cs.washington.edu/~marcotcr/blog/lime/, https://www.oreilly.com/content/introduction-to-local-interpretable-model-agnostic-explanations-lime/
- Explainable Machine Learning in Deployment, FAT* 2020, https://arxiv.org/pdf/1909.06342.pdf; Video: https://www.youtube.com/watch?v=Hofl4uwxtPA

Tutorial: XAI tutorial at AAAI 2020, https://xaitutorial2020.github.io/

Tool: AIX 360

Tool: https://aix360.mybluemix.net/

Video: https://www.youtube.com/watch?v=Yn4yduyoQh4

Paper: https://arxiv.org/abs/1909.03012

Course Project

Project Discussion: What Problem Fascinates You ?

- Data
  - Water
  - Finance
  - …
- Analytics
  - Search, Optimization, Learning, Planning, …
- Application
  - Building chatbot
- Users
  - Diverse demographics
  - Diverse abilities
  - Multiple human languages

Project execution in sprints

- Sprint 1: (Sep 12 – Oct 5)
  - Solving: Choose a decision problem, identify data, work on solution methods
  - Human interaction: Develop a basic chatbot (no AI), no problem focus
- Sprint 2: (Oct 10 – Nov 9)
  - Solving: Evaluate your solution on problem
  - Human interaction: Integrated your choice of chatbot (rule-based or learning-based) and methods
- Sprint 3: (Nov 14 – 30)
  - **Evaluation:** Comparison of your solver chatbot with an LLM-based alternative, like ChatGPT

Project Discussion: Dates and Deliverables

Project execution in sprints

- Sprint 1: (Sep 12 – Oct 5)
  - Solving: Choose a decision problem, identify data, work on solution methods
  - Human interaction: Develop a basic chatbot (no AI), no problem focus
- Sprint 2: (Oct 10 – Nov 9)
  - Solving: Evaluate your solution on problem
  - Human interaction: Integrated your choice of chatbot (rule-based or learning-based) and methods
- Sprint 3: (Nov 14 – 30)
  - **Evaluation:** Comparison of your solver chatbot with an LLM-based alternative, like ChatGPT

- Oct 12, 2023
  - Project checkpoint
  - In-class presentation
- Nov 30, 2023
  - Project report due
- Dec 5 / 7, 2023
  - In-class presentation

Skeleton: A Basic Chatbot

- Run in an infinite loop until the user wants to quit
- Handle any user response
  - User can quit by typing “Quit” or “quit” or just “q”
  - User can enter any other text and the program has to handle it. The program should write back what the user entered and say – “I do not know this information”.
- Handle known user query types // Depends on your project
  - “Tell me about N-queens”, “What is N ?”
  - “Solve for N=4?”
  - “Why is this a solution? ”
- Handle chitchat // Support at least 5, extensible from a file
  - “Hi” => **“Hello”**
  - …
- ***Store session details in a file***

Illustrative Project

1. **Title**: Solve and explain solving of n-queens puzzle
2. **Key idea**: Show students how a course project will look like
3. **Who will care when done**: students of the course, prospective AI students and teachers
4. **Data need**: n: the size of game; interaction
5. **Methods**: search
6. **Evaluation**: correctness of solution, quality of explanation, appropriateness of chat
7. **Users**: with and without AI background; with and without chess background
8. **Trust issue**: user may not believe in the solution, may find interaction offensive (why queens, not kings? …)

Project Discussion: Illustration

1. Create a private Github repository called "CSCE58x-Fall2023-<studentname>-Repo". Share with Instructor (biplav-s) and TA (kausik-l)
2. Create Google folder called "CSCE58x-Fall2023-<studentname>-SharedInfo". Share with Instructor (prof.biplav@gmail.com) and TA (lakkarajukausik90@gmail.com)
3. Create a Google doc in your Google repo called "Project Plan" and have the following by next class (Sep 5, 2023)

1. **Title**: Solve and explain solving of n-queens puzzle
2. **Key idea**: Show students how a course project will look like
3. **Who will care when done**: students of the course, prospective AI students and teachers
4. **Data need**: n: the size of game; interaction
5. **Methods**: search
6. **Evaluation**: correctness of solution, quality of explanation, appropriateness of chat
7. **Users**: with and without AI background; with and without chess background
8. **Trust issue**: user may not believe in the solution, may find interaction offensive (why queens, not kings? …)

Project Illustration: N-Queens

- Sprint 1: (Sep 12 – Oct 5)
  - Solving: Choose a decision problem, identify data, work on solution methods
    - Method 1: Random solution
    - Method 2: Search – BFS
    - Method 3: Search - …
  - Human interaction: Develop a basic chatbot (no AI) as outlined
  - Deliverable
    - Code structure in Github
      - ./data
      - ./code
      - ./docs
      - ./test
    - Presentation: Make sprint presentation on Oct 12, 2023

Reference: Project Rubric

- **Project results** – 60%
  - Working system ? – 30%
  - Evaluation with results superior to baseline? – 20%
  - Considered related work? – 10%
- **Project effort**s – 40%
  - Project report – 20%
  - Project presentation (updates, final) – 20%
- **Bonus**
  - Challenge level of problem – 10%
  - Instructor discretion – 10%
- **Penalty**
  - Lack of timeliness as per announced policy (right) - up to 30%

Milestones and **Penalties**

- Oct 12, 2023
  - Project checkpoint
  - In-class presentation
  - **Penalty: presentation not ready by Oct 10, 2023 [-10%]**
- Nov 30, 2023
  - Project report due
  - **Project report not ready by date [-10%]**
- Dec 5 / 7, 2023
  - In-class presentation
  - **Project presentations not ready by Dec 4, 2023 [-10%]**

<Project Title> - <Your Name>

Format for Interim Presentation on Oct 12, 2023

Project Context

1. Problem
2. Who will care/ users
3. Data needs:
4. Methods:
5. Evaluation:
6. Trust issue:

Achievement

- Status
- Test Case
  - E.g., <input, correct output>
- Sample Result
- Discuss others points:
  - Challenges faced
  - Any help needed

1 min context, 1 min achievement, 1 min Q/A

Lecture 16: Summary

- We talked about
  - (Sprint 1) Project Presentations
  - Trust Issues
    - Explainability
  - LIME tool

Concluding Section

About Next Lecture – Lecture 17

Lecture 17: Chatbot Building

- Building Chatbots
  - With RASA
  - SafeChat Framework