AI INSTITUTE #AIISC
UNIVERSITY OF SOUTH CAROLINA

UNIVERSITY OF SOUTH CAROLINA

CSCE 581: Introduction to Trusted AI

Lectures 17 and 18: Invited Talk, AI with Text (Intro)

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

18TH AND 20TH MAR, 2025

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

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

Organization of Lectures 17, 18

- Introduction Section
  - Recap from Week 8 (Lectures 15 and 16)
  - Announcements and News
- Main Section
  - L17:Invited Talk on Trust and Agentic AI
  - L18: Introduction to AI with Text Data (NLP)
- Concluding Section
  - About next week – Lectures 19, 20
  - Ask me anything

Introduction Section

Recap from Week 8 (Lectures 15, 16)

- We looked at
  - L15: LLMs, Explanation
  - L16: Project Update #2

Announcements and News

- Reading list updated with new material on LLMs and prompt methods
  - https://github.com/biplav-s/course-tai-s25/blob/main/reading-list/Readme-LLMs.md
- From next class, we will work with text data for AI-based processing. Suggested reading from reading list - **Contextual Word Representations: Putting Words into Computers**, by Noah Smith, CACM June 2020
  - https://cacm.acm.org/research/contextual-word-representations/
- Courtesy of NSF's National AI Resources (NAIRR) award for teaching last week, I will be introducing Vocareum Notebooks (https://www.vocareum.com/) to students. We will work with SC traffic data. Details to follow.

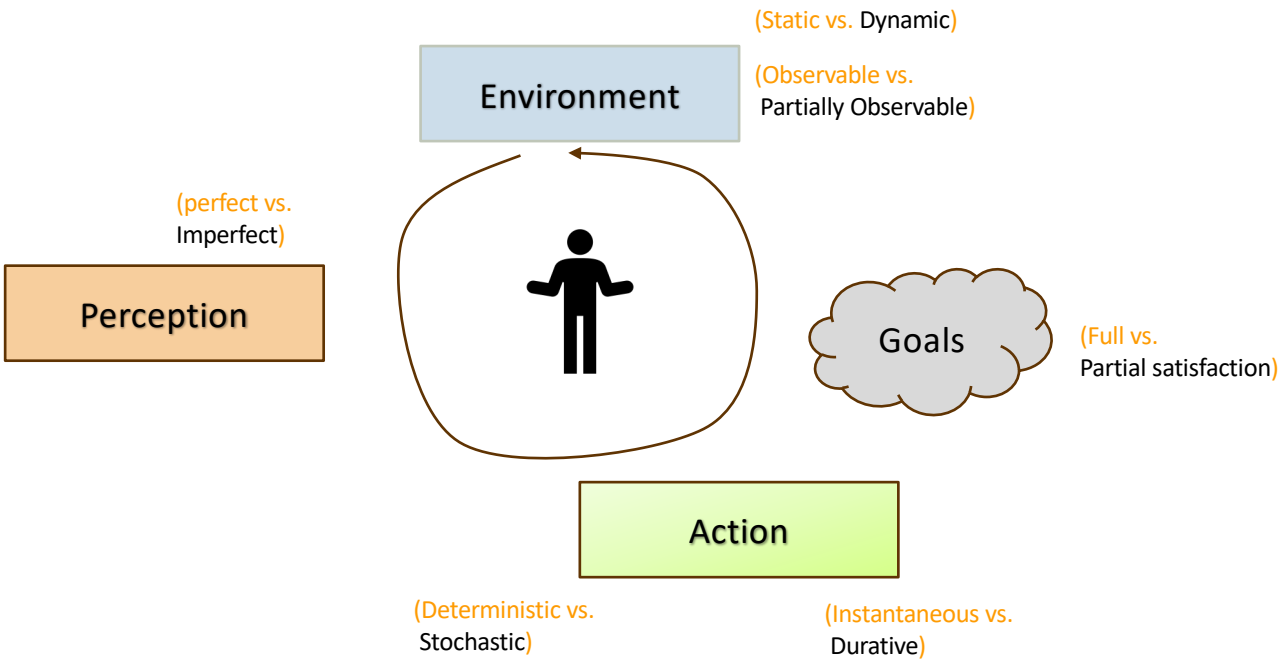
Announcement: Change to Student Assessment

A = [920-1000]
B+ = [870-919]
B = [820-869]
C+ = [770-819]
C = [720-769]
D+ = [670-719]
D = [600-669]
F = [0-599]

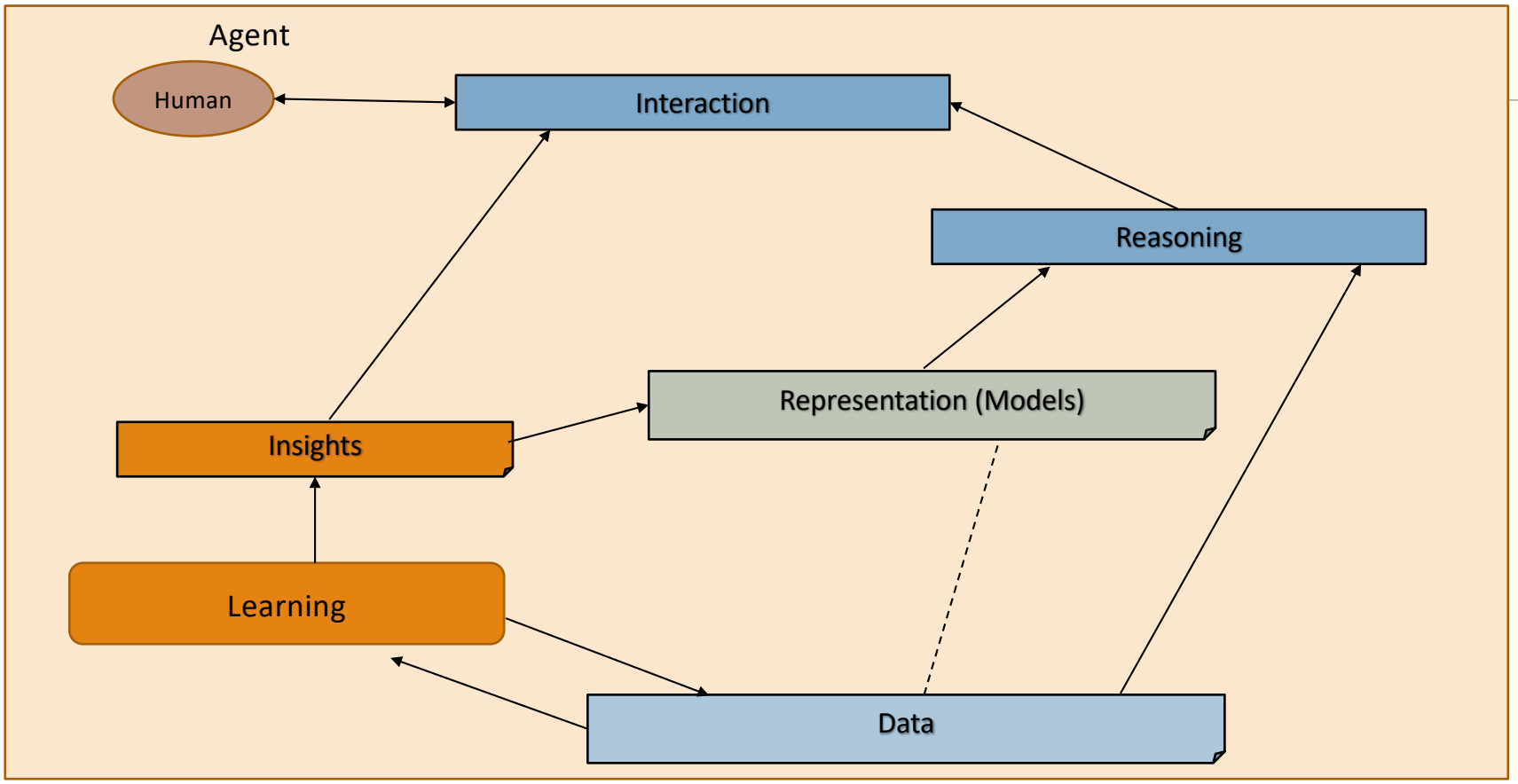
Tests	Undergrad	Grad
Course Project – report, in-class presentation	600	600
Quiz – 2 quizzes	200	200
Final Exam	200	100
Additional Final Exam – Paper summary, in-class presentation		100
Total	1000 points	1000 points

Change: 4 quizzes to 2; no best of 3

Intelligent Agent Model



Relationship Between Main AI Topics (Covered in Course)



High Level Semester Plan (Adapted, Approximate)

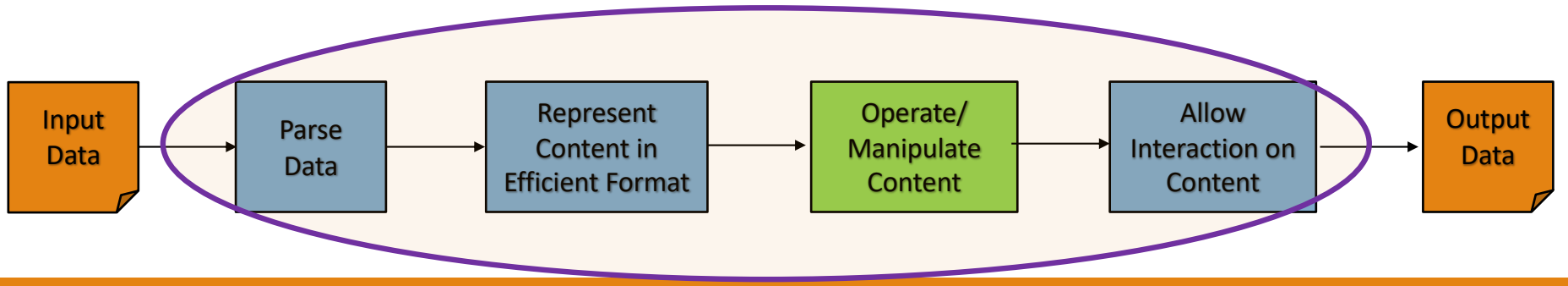
CSCE 581 –

- Week 1: Introduction
- Week 2: Background: AI - Common Methods
- Week 3: The Trust Problem
- Week 4: Machine Learning (Structured data) - Classification
- Week 5: Machine Learning (Structured data) - Classification – Trust Issues
- Week 6: Machine Learning (Structured data) – Classification – Mitigation Methods
- Week 7: Machine Learning (Structured data) – Classification – Explanation Methods
- Week 8: Machine Learning (Text data, **vision**) – Classification, **Large Language Models**
- Week 9: Machine Learning (Text data) - Classification – Trust Issues, LLMs
- Week 10: Machine Learning (Text data) – Classification – Mitigation Methods
- Week 11: Machine Learning (Text data) – Classification – Explanation Methods
- Week 12: Emerging Standards and Laws, **Real world applications**
- Week 13: Project presentations
- Week 14: Project presentations, Conclusion

Increased focus on LLMs and projects now

AI/ ML topics and with a focus on fairness, explanation, Data privacy, reliability

Main Segment



Guest Speaker

Human-Centered Trustworthy Agentic AI, by Dr. Kush Varshney

March 18, 2025, 4:30-5:30pm, Zoom Meeting - https://sc-edu.zoom.us/j/83709414377

Dr. Varshney is an IBM Fellow based at the Thomas J. Watson Research Center, Yorktown Heights, NY, where he directs Human-Centered Trustworthy AI research. He applies data science and predictive analytics to human capital management, healthcare, olfaction, computational creativity, public affairs, international development, and algorithmic fairness, which has led to the Extraordinary IBM Research Technical Accomplishment for contributions to workforce innovation and enterprise transformation, and IBM Corporate Technical Awards for Trustworthy AI and for AI-Powered Employee Journey. He and his team created several well-known open-source toolkits, including AI Fairness 360, AI Explainability 360, Uncertainty Quantification 360, and AI FactSheets 360. AI Fairness 360 has been recognized by the Harvard Kennedy School's Belfer Center as a tech spotlight runner-up and by the Falling Walls Science Symposium as a winning science and innovation management breakthrough.

He independently-published a book entitled 'Trustworthy Machine Learning' in 2022, available at http://www.trustworthymachinelearning.com. He is a fellow of the IEEE.
Personal website: https://krvarshney.github.io

Discussion

- Trustworthy AI
- Agentic AI
- Trust and Human-Centric AI, Talk

Class 18: Introduction to Text and AI

Examples

- Processing
  - Resumes -> e.g., selecting candidates
  - Answer papers (of examinations) -> e.g., grading
  - Medical reports -> e.g., helping patients
- Predicting
  - Business intelligence
  - Financial forecasts
- Generating
  - News summary
  - Summarizing literature for research

Common Textual Data Processing Steps for ML

- Input: strings / documents/ corpus
- Processing steps (task dependent / optional - *)
  - Parsing
  - Word pre-processing
    - Tokenization – getting tokens for processing
    - Normalization* - making into canonical form
    - Case folding* – handling cases
    - Lemmatization* – handling variants (shallow)
    - Stemming* – handling variants (deep)
  - Semantic parsing – representations for reasoning with meaning *
  - Embedding – creating vector representation*

CSCE 771 goes into details

Common NLP Tasks

- Extracting entities [Entity Extraction]
- Finding sentiment [Sentiment Analysis]
- Generating a summary [Text Summarization]
- Translating to a different language [Machine translation]
- Natural Language Interface to Databases [NLI]
- Natural Language Generation [NLG]

CSCE 771 goes into details

Word Representation: Paper Discussion

Contextual Word Representations: Putting Words into Computers”,

by Noah Smith, CACM June 2020

https://cacm.acm.org/research/contextual-word-representations/

Problem

- How to represent words ?
- How to measure similarity, e.g., between words, and texts?
- How to determine different contexts (senses) in which words are used?
- How to handle noise, typos?

S1 - This is an apple
S2 - These are apples

S3 - This is an apples
S4 - There are apply

Option 1 - Characters

- How to represent words?
  - Characters / Unicode / …
- How to measure similarity between words, and texts?
  - Edit distance***: actions to convert one string to another***
  - Hamming distance: ***difference considering substitution***
- How to determine different contexts (senses) in which words are used?
  - Neighborhood of words: Bi-, tri-, N-gram representations

Distance between: Kitten, Sitting

Edit Distance

Algorithm	Operations Allowed			
	Insertions	Deletions	Substitutions	Transposition
Levenshtein Distance	✓	✓	✓	
Longest Common Subsequence (LCS)	✓	✓		
Hamming Distance			✓	
Damerau–Levenshtein Distance	✓	✓	✓	✓
Jaro distance				✓

Levenshtein distance:
1.kitten → **s**itten (substitute "s" for "k")
2.sitt**e**n → sitt**i**n (substitute "i" for "e")
3.sittin → sittin**g** (insert "g" at the end)

LCS distance (insertions and deletions only):
1.kitten → itten (delete "k" at 0)
2.itten → **s**itten (insert "s" at 0)
3.sitt**e**n → sittn (delete "e" at 4)
4.sittn → sitt**i**n (insert "i" at 4)
5.sittin → sittin**g** (insert "g" at 6)

Source: https://en.wikipedia.org/wiki/Edit_distance

Option 2 - Vectors

- How to represent words? Vectors
  - But, what scheme in vectors
    - One-hot encoding
    - Arbitrary, principled, …
- How to measure similarity between words, and texts?
  - Cosine similarity
- How to determine different contexts in which words are used?
  - Neighborhood of words: Bi-, tri-, N-gram representations
  - Contextual word vectors

Cosine Similarity

$$\text{cosine similarity} = S_C(A, B) := \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \cdot \sqrt{\sum_{i=1}^{n} B_i^2}},$$

Property: two proportional vectors have a cosine similarity of 1, two orthogonal vectors have a similarity of 0, and two opposite vectors have a similarity of -1.

Usually used for [0,1]

Sci-kit method python: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html

Source: https://en.wikipedia.org/wiki/Cosine_similarity

TF-IDF based Word Representation -1

- Given N documents

- **Term frequency (TF):** for term (word) t in document d = tf(t, d)

Variants to reduce bias due to document length

Variants of term frequency (tf) weight

weighting scheme	tf weight
binary	$0, 1$
raw count	$f_{t,d}$
term frequency	$f_{t,d} \Big/ \sum_{t' \in d} f_{t',d}$
log normalization	$\log(1 + f_{t,d})$
double normalization 0.5	$0.5 + 0.5 \cdot \frac{f_{t,d}}{\max_{\{t' \in d\}} f_{t',d}}$
double normalization K	$K + (1 - K)\frac{f_{t,d}}{\max_{\{t' \in d\}} f_{t',d}}$

Sources:
(a) sci-kit documentation
(b) Wikipedia: https://en.wikipedia.org/wiki/Tf%E2%80%93idf

TF-IDF based Word Representation -2

- Given N documents
- Term frequency (TF): for
  term (word) t in document d
  = tf(t, d)
- **Inverse document frequency IDF**(t)

  = log [N / **DF**(t)] + 1

DF(t) = **document frequency**, the number of documents in the document set that contain the term t.

- **TF-IDF**(t, d) = TF(t, d) * IDF(t),

Variants of inverse document frequency (idf) weight

weighting scheme	idf weight ($n_t = \|\{d \in D : t \in d\}\|$)
unary	1
inverse document frequency	$\log \frac{N}{n_t} = -\log \frac{n_t}{N}$
inverse document frequency smooth	$\log\left(\frac{N}{1+n_t}\right)+1$
inverse document frequency max	$\log\left(\frac{\max_{\{t' \in d\}} n_{t'}}{1+n_t}\right)$
probabilistic inverse document frequency	$\log \frac{N-n_t}{n_t}$

Sources:
(a) sci-kit documentation
(b) Wikipedia: https://en.wikipedia.org/wiki/Tf%E2%80%93idf

TF-IDF Example Calculation

See sample code on GitHub:
https://github.com/biplav-s/course-nl-f22/blob/main/sample-code/l5-wordrepresent/Word%20Representations%20-%20Vectors.ipynb

Contextual Representation by Dimensionality Reduction - 1

- Strategy 2: learn contexts from documents. Vector size is given as input
- Train a neural network to learn vector representation
  - value placed in each dimension of each word type's vector is a parameter that will be optimized
  - Selection of parameter values is done using iterative algorithms / gradient descent
  - **Hope** is that ***different senses*** in which a word is used will be captured through the learning procedure as long as the dataset is large enough to represent all senses. Paper quotes: 30 meanings of **get**
- **Optionally**: Sometime task specific inputs are given during pre-processing, processing or post-processing

Disadvantage: individual dimensions are no longer interpretable

Contextual Representation by Dimensionality Reduction -2

- Strategy 2: learn contexts from documents. Vector size is given as input

Sometime task specific inputs are given during pre-processing, processing or post-processing

- Pre-processing
  - Vector initialization by pre-training. Called **finetuning**
- Processing
  - **Knowledge-infusion** (emerging area)
- Post-processing
  - Adjust output vectors so that word types that are related in reference taxonomy (like WordNet) are closer to each other in vector space. Called **retrofitting**.

Credit:
Contextual Word Representations: Putting Words into Computers", by Noah Smith, CACM June 2020

Where Are We

- Learning representation
  - Approach 1: count-based
    - Creating word vectors in which each dimension corresponds to the frequency the word type occurred in some context.
    - Example: TF-IDF
  - Approach 2: learning-based
    - learn contexts from documents. Vector size is given as input
    - Examples: Word2Vec, Glove, RNN/LSTM (arc), Transformers

Project Discussion

Course Project

- **Framework**
  1. (Problem) Think of a problem whose solution may benefit people (e.g., health, water, air, traffic, safety)
  2. (User) Consider how the primary user (e.g., patient, traveler) may be solving the problem today
  3. (AI Method) Think of what the solution will do to help the primary user
     1. Solution => ML task (e.g. classification), recommendation, text summarization, …
     2. Use a foundation model (e.g., LLM-based) solution as the baseline
  4. (Data) Explore the data for a solution to work
  5. (Reliability: Testing) Think of the evaluation metric we should employ to establish that the solution will works? (e.g., 20% reduction in patient deaths)
  6. (Holding Human Values) Discuss if there are fairness/bias, privacy issues?
  7. (Human-AI) Finally, elaborate how you will explain the primary user that your solution is trustable to be used by them

Project Discussion: What to Focus on ?

- Problem: you should care about it
- Data: should be available
- Method: you need to be comfortable with it. Have at least two – one serves as baseline
- Trust issue
  - Due to Users
    - Diverse demographics
    - Diverse abilities
    - Multiple human languages
  - Or other impacts
- What one does to mitigate trust issue

Rubric for Evaluation of Course Project

Project

- Project plan along framework introduced (7 points)
- Challenging nature of project
- Actual achievement
- Report
- Sharing of code

Presentation

- Motivation
- Coverage of related work
- Results and significance
- Handling of questions

Concluding Section

Week 9 (L17 and 18): Concluding Comments

- We looked at
  - Trust, Human Focus and Agentic AI – invited talk
  - Text processing, representation

About Next Week – Lectures 19, 20

Lectures 19, 20

- AI - Unstructured (Text): Representation, Common NLP Tasks
- Natural Languages/ Language Models and their Impact on AI

13	Feb 25 (Tu)	AI - Supervised ML: Explanation Tools
14	Feb 27 (Th)	AI Trust - Mitigation method (Trust rating) – Kausik Lakkaraju
15	Mar 4 (Tu)	Large Language Models (LLMs), Machine Learning – Trust Issues (Explainability)
16	Mar 6 (Th)	Student presentations - project
	Mar 11 (Tu)	
	Mar 12 (Th)	
17	Mar 18 (Tu)	**Invited Guest** – Kush Varshney
18	Mar 20 (Th)	AI - Unstructured (Text): Processing and Representation
19	Mar 25 (Tu)	AI - Unstructured (Text): Representation, Common NLP Tasks, Large Language Models (LLMs)
20	Mar 27 (Th)	Natural Languages/ Language Models and their Impact on AI
21	Apr 1 (Tu)	AI - Unstructured (Text): Analysis – Supervised ML – Trust Issues
22	Apr 3 (Th)	AI - Unstructured (Text): Analysis – Supervised ML – Mitigation Methods
23	Apr 8 (Tu)	AI - Unstructured (Text): Analysis – Rating and Debiasing Methods