AI INSTITUTE #AIISC
UNIVERSITY OF SOUTH CAROLINA

UNIVERSITY OF SOUTH CAROLINA

CSCE 590-1: Trusted AI

Lecture 14: Unstructured Text – Representation and Common NLP Tasks

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

5TH OCT, 2021

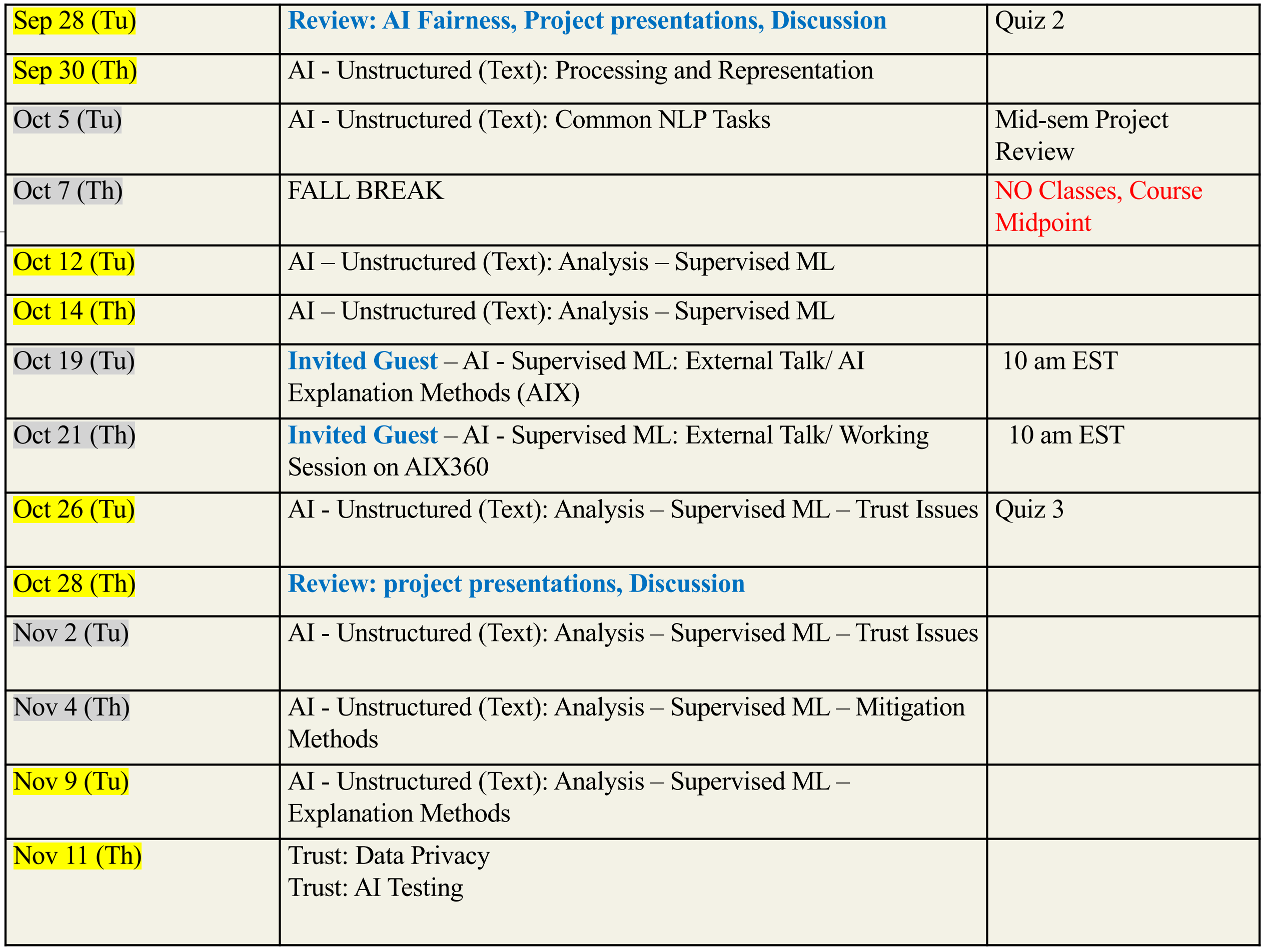
Carolinian Creed: "I will practice personal and academic integrity."

Organization of Lectures 14

- Introduction Segment
  - Recap of Lecture 13
  - Review of Quiz 2 and Submissions
  - Attendance policy - reiterate
- Main Segment
  - Word Representation – code samples
  - Contextual Word Representation / Embedding
  - Common NLP Tasks
- Concluding Segment
  - About next lecture – Lecture 15
  - Ask me anything

Introductory Segment

Schedule Snapshot



Sep 28 (Tu)	**Review: AI Fairness, Project presentations, Discussion**	Quiz 2
Sep 30 (Th)	AI - Unstructured (Text): Processing and Representation	
Oct 5 (Tu)	AI - Unstructured (Text): Common NLP Tasks	Mid-sem Project Review
Oct 7 (Th)	FALL BREAK	NO Classes, Course Midpoint
Oct 12 (Tu)	AI – Unstructured (Text): Analysis – Supervised ML	
Oct 14 (Th)	AI – Unstructured (Text): Analysis – Supervised ML	
Oct 19 (Tu)	**Invited Guest** – AI - Supervised ML: External Talk/ AI Explanation Methods (AIX)	10 am EST
Oct 21 (Th)	**Invited Guest** – AI - Supervised ML: External Talk/ Working Session on AIX360	10 am EST
Oct 26 (Tu)	AI - Unstructured (Text): Analysis – Supervised ML – Trust Issues	Quiz 3
Oct 28 (Th)	**Review: project presentations, Discussion**	
Nov 2 (Tu)	AI - Unstructured (Text): Analysis – Supervised ML – Trust Issues	
Nov 4 (Th)	AI - Unstructured (Text): Analysis – Supervised ML – Mitigation Methods	
Nov 9 (Tu)	AI - Unstructured (Text): Analysis – Supervised ML – Explanation Methods	
Nov 11 (Th)	Trust: Data Privacy Trust: AI Testing	

Recap of Lecture 13

- Reviewed words in English and forms
- Understood basics of language processing
  - Parsing with Regular expressions
- Explored (discrete) representation

Quiz 2 Recap and Submissions

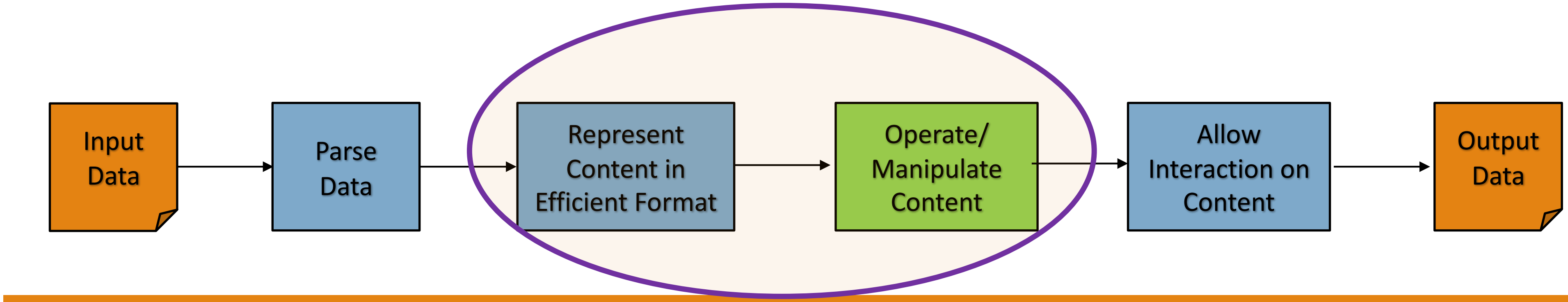
- Individual part: Trust issues in projects
- Group part:
  - Water treatment data in Weka
  - Water Atlas data, multiple locations

Attendance Policy – Communicated: 23 Sep 2021

[Attendance penalty / reward rule effective from communication date]

- For those not attending in person and not informing (3 or more): penalty of 10% aggregate score
  - Translates to 1 grade below or more
- For those not attending in person but informing on non-medical grounds (3 or more):
  - Penalty of 5% aggregate score
- For those attending all in person (1 miss max allowed): 10% extra aggregate score
  - Translates to 1 grade above

Main Segment


Input Data
Parse Data
Represent Content in Efficient Format
Operate/ Manipulate Content
Allow Interaction on Content
Output Data

Word Representation

- Words as independent, discrete symbols
- Why
  - To process language efficiently (numeric representation)
  - Find similarity between words efficiently (vector representation)

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

Code Review: Discrete Word Representation

- Word parsing with regex in Python
- Discrete word representation

Code: https://github.com/biplav-s/course-tai/blob/main/sample-code/l13-l16-supervised-text/l13-basic-text.ipynb

Contextual Word Representation

Key Ideas

- **Context**: given a word, its context is defined by its nearby words or sequences of words
- **Distributional semantics:** Words used in similar ways are likely to have related meanings
  - Words used in the same (similar) context have related meanings
  - No claim about meaning except relative similarity v/s dis-similarity of words

- Two main strategies to know about context
  - Obtain from words in a manually-created taxonomy, e.g., Wordnet
  - Learn context and representation from data

Details:
Contextual Word Representations: Putting Words into Computers", by Noah Smith, CACM June 2020,
https://cacm.acm.org/magazines/2020/6/245162-contextual-word-representations/fulltext

Give Meaning to Each Dimension in Vector

- Manually assign meaning to dimensions. Examples
  - Dimension by morphological forms
  - Dimension by semantic type: e.g., days of week

- Issues:
  - Need to decide dimensions ahead of time

Can We Do Better?

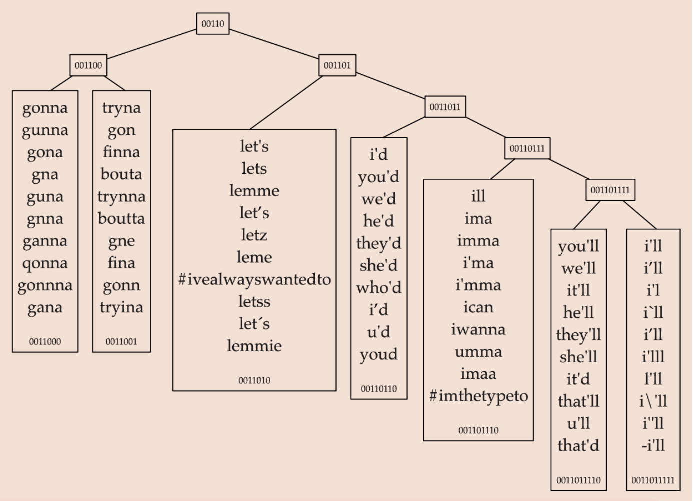
- Representations that make computation robust
  - Overcome typographical mistakes

- Learn representation that helps with down-stream tasks
  - Finding similar words
  - Finding words in similar situations

- Learn new properties from external datasources (like Wikipedia)

Contextual Representation by Clustering

- Cluster words by context
- Compare with words in a manually-created taxonomy, e.g., Wordnet



The 10 most frequent words in clusters in the section of the hierarchy with prefix bit string 00110.

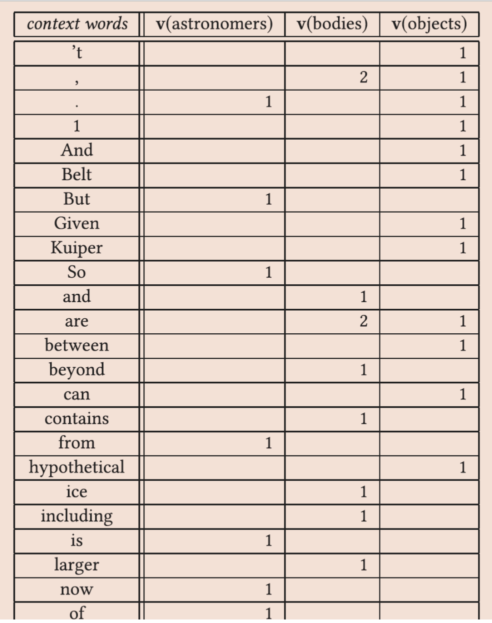
Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K.,
Schneider, N., and Smith, N.A. Improved part-ofspeech
tagging for online conversational text with
word clusters. In Proceedings of 2013 NAACL.

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

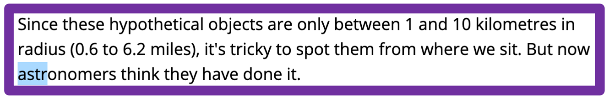
Contextual Representation by Dimensionality Reduction

- Creating word vectors in which each dimension corresponds to the frequency the word type occurred in some context.

Document: https://bit.ly/2B9uaKr
“frequency of each word occurring within two positions on either side of the word whose vector we are constructing”

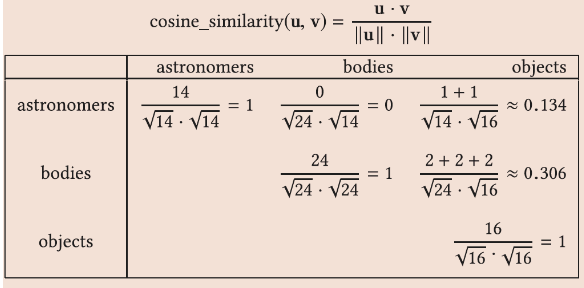


context words	**v**(astronomers)	**v**(bodies)	**v**(objects)
’t			1
,		2	1
.	1		1
1			1
And			1
Belt			1
But	1		
Given			1
Kuiper			1
So	1		
and		1	
are		2	1
between			1
beyond		1	
can			1
contains		1	
from	1		
hypothetical			1
ice		1	
including		1	
is	1		
larger		1	
now	1		
of	1		



Since these hypothetical objects are only between 1 and 10 kilometres in radius (0.6 to 6.2 miles), it's tricky to spot them from where we sit. But now astronomers think they have done it.

- Strategy 1: select contexts
  - Examples
    - Words in the neighborhood
    - Words of specific types
  - Build vectors
  - Use vector operations to derive meaning



$$\text{cosine\_similarity}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \cdot \|\mathbf{v}\|}$$

	astronomers	bodies	objects
astronomers	$\frac{14}{\sqrt{14} \cdot \sqrt{14}} = 1$	$\frac{0}{\sqrt{24} \cdot \sqrt{14}} = 0$	$\frac{1+1}{\sqrt{14} \cdot \sqrt{16}} \approx 0.134$
bodies		$\frac{24}{\sqrt{24} \cdot \sqrt{24}} = 1$	$\frac{2+2+2}{\sqrt{24} \cdot \sqrt{16}} \approx 0.306$
objects			$\frac{16}{\sqrt{16} \cdot \sqrt{16}} = 1$

Bodies and **objects** are **most similar** (0.306) than

- **Bodies** and **astronomers** (0)
- **Objects** and **astronomers** (0.134)

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

TF-IDF Method

- Given N documents
  - For document d, with terms t

- Computer two quantities
  - Term-Frequency (TF) : TF(t, d)
  - Inverse Document Frequency (IDF): **IDF**(t)

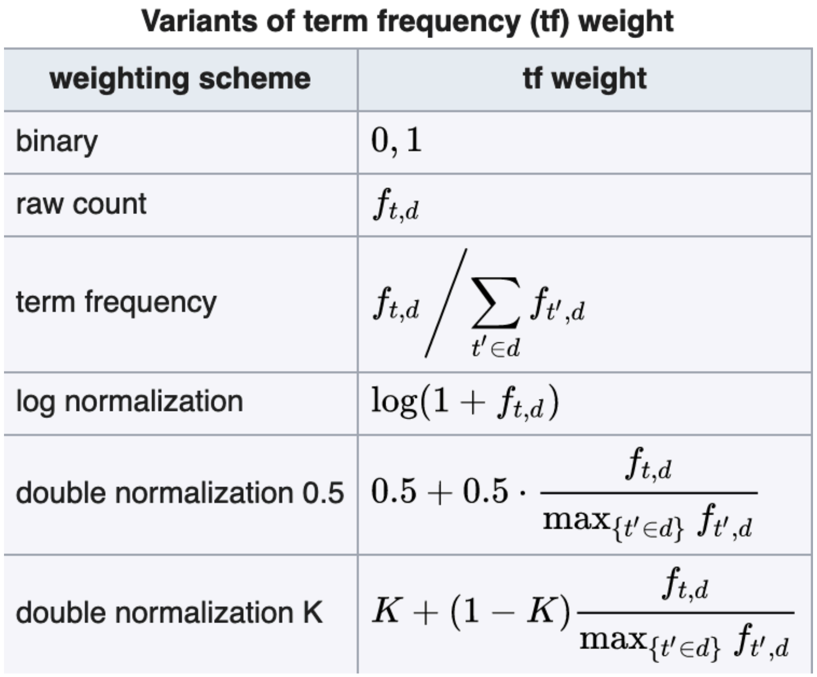
- Representation: **TF-IDF**(t, d) = TF(t, d) * IDF(t)

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

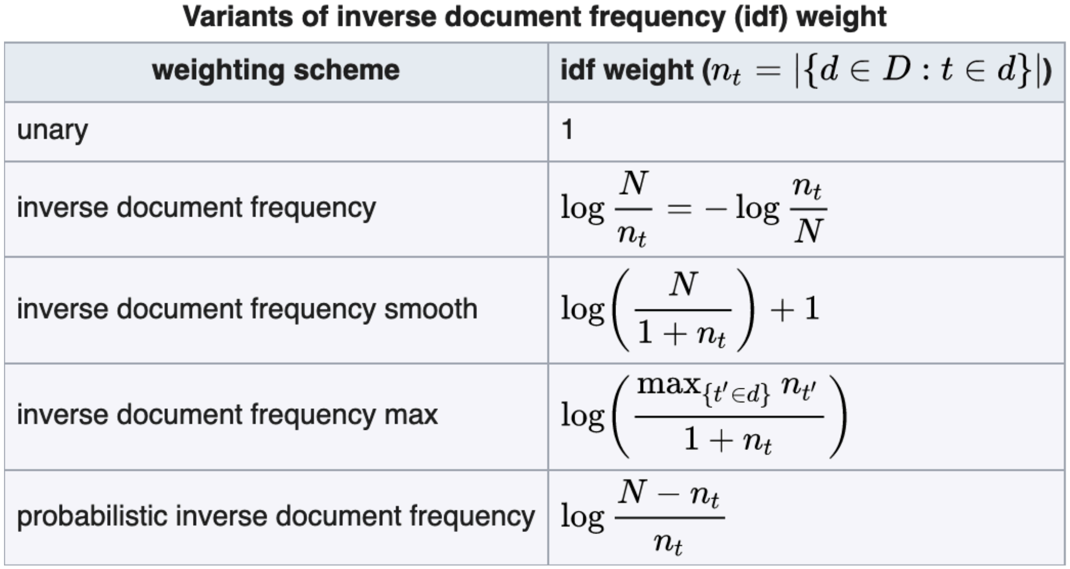
- Given N documents
- Term frequency (TF): for term (word) t in document d
  = tf(t, d)

- **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

Related Concept: Language Model

Problem:
Given a sentence fragment, predict what word(s) come next

Language Model:
estimate probability of substrings of a sentence

$$P(w_i|w_1, w_2, \ldots, w_{i-1}) = \frac{P(w_1, w_2, \ldots, w_{i-1}, w_i)}{P(w_1, w_2, \ldots, w_{i-1})}$$

Bigram approximation

$$P(w_i|w_1, w_2, \ldots, w_{i-1}) \approx \frac{P(w_{i-1}, w_i)}{P(w_{i-1})}$$

Applications:

- Spelling correction
- speech recognition
- machine translation,
- …

From Jurafsky & Martin

TF-IDF Example Calculation

Sample code: https://github.com/biplav-s/course-tai/blob/main/sample-code/l13-l16-supervised-text/l14-contextual-representation.ipynb

Contextual Representation by Dimensionality Reduction - 1

Summary: 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

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

Contextual Word Embeddings

	Name	Description	URL, References
1.	Elmo (embeddings from language models)	Contextual, deep, character-based	https://allennlp.org/elmo; Deep contextualized word representations, Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer. NAACL 2018.
2	Word2Vec	Word-based, prediction focus	*Mikolov, Tomas; et al. (2013). "Efficient Estimation of Word Representations in Vector Space". arXiv:1301.3781 [cs.CL]. Mikolov, Tomas (2013). "Distributed representations of words and phrases and their compositionality". Advances in Neural Information Processing Systems. arXiv:1310.4546.*
3	Glove	Word-based, count	https://nlp.stanford.edu/projects/glove/, Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. [pdf] [bib]
4	Fasttext	Variation of word2vec, works with N-gram, words not in vocabulary	
5	Transformers		https://arxiv.org/abs/1706.03762, http://www.columbia.edu/~jsl2239/transformers.html , Huggingface tutorial: https://huggingface.co/course/chapter0

Commentaries/ Tutorials:

- https://jalammar.github.io/illustrated-bert/ , https://cai.tools.sap/blog/glove-and-fasttext-two-popular-word-vector-models-in-nlp/
- Neural Machine Translation and Sequence-to-sequence Models: A Tutorial, Graham Neubig, https://arxiv.org/abs/1703.01619

Code Example

Word2Vec Sample code: https://github.com/biplav-s/course-tai/blob/main/sample-code/l13-l16-supervised-text/l14-contextual-representation.ipynb

Common NLP Tasks

- **Text similarity**
- Event Extraction
- **Sentiment detection**
- Question Answering
- Summarization
- Machine translation
- Natural Language Interface to Databases
- Natural Language Generation

Text Similarity

- Two variants: matching the concepts as expressed representations of strings (Alqahtani et al.2021) or as real world entities / entity matching (Li et al. 2021)

- String matching
  - Exact
  - Approximate
    - Distance metrics

- Code examples: https://github.com/biplav-s/course-tai/blob/main/sample-code/l13-l16-supervised-text/l14-string-similarity.ipynb

Sentiment Detection

Sentiment Detection

Types of Sentiment Tasks

- Sentiment-oriented Word Embedding
- Sentence-level Models
  - Input: Set of sentences, each made up of a set of words
  - Output: A set of labels (positive, negative, neutral)
- Document-level Models
  - Input: Set of documents, each made up of a set of sentences, each made up of a set of words
  - Output: A set of labels (positive, negative, neutral)
- Fine-grained sentiment labels
  - (e.g., sentiment strength)

A Simple Rule-Based Sentiment Engine

- Process input to get tokens
  - Perform: Stemming, tokenization, part-of-speech tagging and semantic parsing.
- Use lexicons to find polarity of words
- Use a method to aggregate over polarity of words

Basic Sentiment Analysis

Sample code for TextBlob, Vader:
https://github.com/biplav-s/course-d2d-ai/blob/main/sample-code/l23-textrepresent/Basic%20Sentiment.ipynb

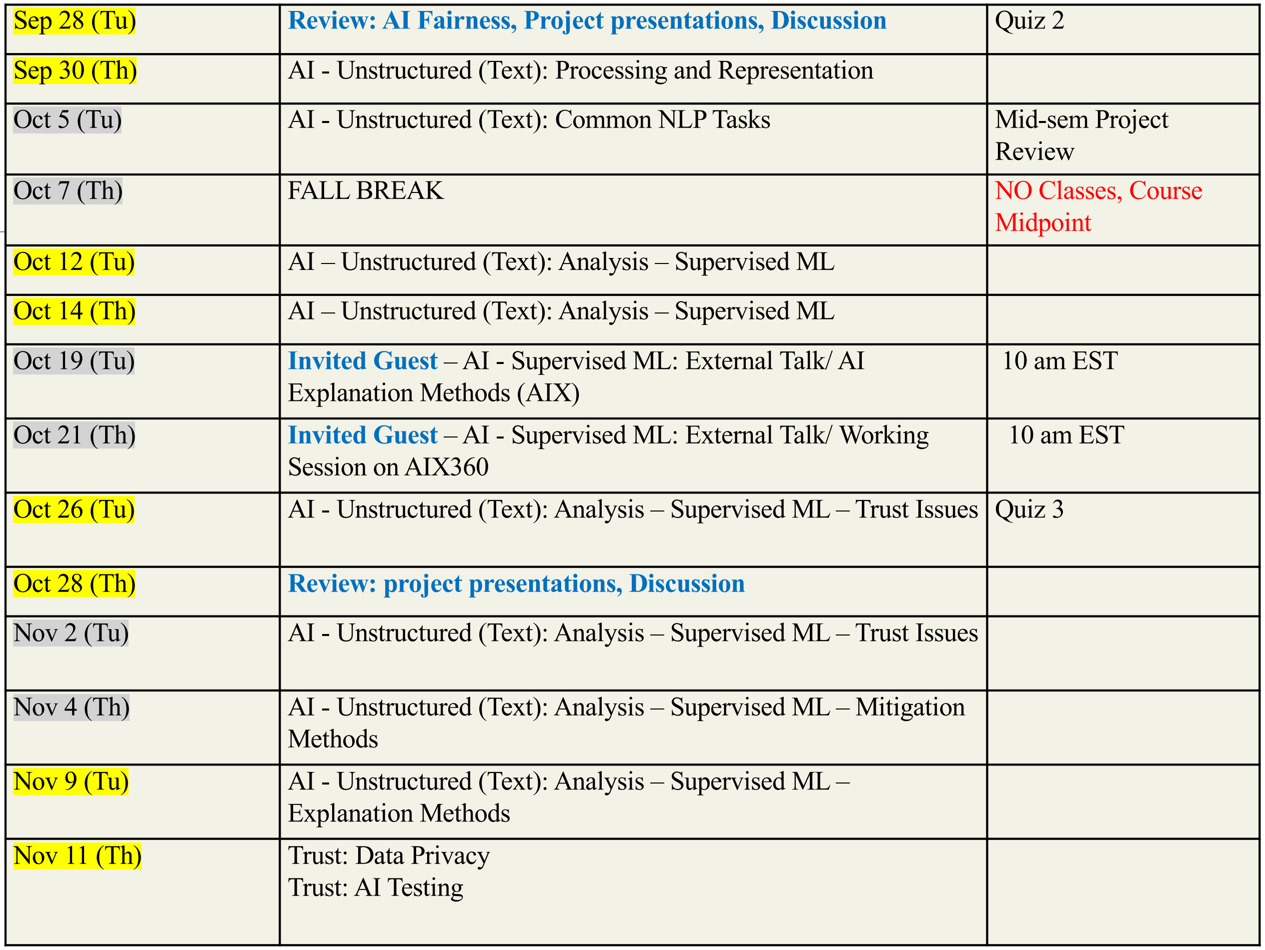
Concluding Segment

Lecture 14: Concluding Comments

- We looked at contextual word representation methods
- We started with basic NLP tasks: string similarity and sentiment detection

About Next Lecture – Lecture 15

Schedule Snapshot



Sep 28 (Tu)	**Review: AI Fairness, Project presentations, Discussion**	Quiz 2
Sep 30 (Th)	AI - Unstructured (Text): Processing and Representation	
Oct 5 (Tu)	AI - Unstructured (Text): Common NLP Tasks	Mid-sem Project Review
Oct 7 (Th)	FALL BREAK	NO Classes, Course Midpoint
Oct 12 (Tu)	AI – Unstructured (Text): Analysis – Supervised ML	
Oct 14 (Th)	AI – Unstructured (Text): Analysis – Supervised ML	
Oct 19 (Tu)	**Invited Guest** – AI - Supervised ML: External Talk/ AI Explanation Methods (AIX)	10 am EST
Oct 21 (Th)	**Invited Guest** – AI - Supervised ML: External Talk/ Working Session on AIX360	10 am EST
Oct 26 (Tu)	AI - Unstructured (Text): Analysis – Supervised ML – Trust Issues	Quiz 3
Oct 28 (Th)	**Review: project presentations, Discussion**	
Nov 2 (Tu)	AI - Unstructured (Text): Analysis – Supervised ML – Trust Issues	
Nov 4 (Th)	AI - Unstructured (Text): Analysis – Supervised ML – Mitigation Methods	
Nov 9 (Tu)	AI - Unstructured (Text): Analysis – Supervised ML – Explanation Methods	
Nov 11 (Th)	Trust: Data Privacy Trust: AI Testing	

Lecture 15: Unstructured Text - Classification

- Pre-trained language models
- Common NLP Tasks
  - Text similarity
  - Event Extraction
  - Sentiment detection
  - Question Answering
  - Summarization
  - Machine translation
  - Natural Language Interface to Databases
  - Natural Language Generation
- Classification with Unstructured Text