

CSCI 5541: Natural Language Processing

Lecture 1: Class Overview

Dongyeop Kang, University of Minnesota



UNIVERSITY OF MINNESOTA
Driven to Discover®

Overview

- ❑ Instructors
- ❑ Course information
- ❑ Grading and Late Day Policy
 - Class Participation
 - Homework
 - Project
 - Reading Assignment
- ❑ Outline of Topics
- ❑ Computing resources



Instructors



Dongyeop Kang (DK)
Instructor



Shirley A. Hayati
Graduate TA



Shuyu Gan
Graduate TA



Drew Gjerstad
Undergraduate TA



Xiaxuan Zhang
Undergraduate TA

- ❑ DK (mostly Slack, in-person by appointment)
- ❑ Shirley: Wednesday 11:30-12, <https://meet.google.com/fnd-mmeh-mus>
- ❑ Shuyu: Thursday 4:30-5:00, Shepherd 443 or <https://umn.zoom.us/j/94875649152?pwd=6VQa8EH5FZNMPWpqTZZqu8ZoYF0kjY.1>
- ❑ Drew: Tuesday 1-2, Keller 1-213 Table #1 or <https://umn.zoom.us/j/99290100950>
- ❑ Xiaxuan: 1:30-2



Course Information

❑ Class meets:

- Tuesday and Thursday, 11:15AM to 12:30PM
- Akerman Hall 319
- Note that no recording will be available this time

❑ Class homepage

- <https://dykang.github.io/classes/csci5541/F25>

❑ HW/Project discussion on Slack

- <https://csci5541f25.slack.com/>

❑ HW/Project submission and grading on Canvas

- <https://canvas.umn.edu/courses/518535>



Outline

- ❑ The purpose of this course is to provide an overview of the *computational techniques* developed to enable computers to **understand** and **respond** appropriately to ideas expressed using natural languages.
- ❑ This course will cover a broad range of *computational models* in NLP,
 - from language representations, to development and evaluation of NLP applications, to practical coding/machine-learning skills to implement them.
 - text classification, distributional representation methods of language, large language models, and advanced techniques used in chatGPT



IMPORTANT NOTE for background

- ❑ This course covers NLP techniques based on recent *deep learning* models, assuming students' knowledge of basic machine learning
- ❑ This course focuses ***practical learning*** of NLP knowledge
 - Programming & reading assignments and a semester-long project
 - Assignments and project require programming with PyTorch
 - We use Python for programming and LaTeX for report writing
- ❑ If students are not familiar with either one of the fields, they **should consider dropping the course** as it would be very challenging and the course is not designed to cover all aspects.



Prerequisite

❑ Required:

- CSCI 2041 Advanced Programming Principles
- **CSCI 5521 Machine Learning Fundamentals**,

❑ Recommended but not mandatory:

- CSCI 5527 Deep Learning, or
- Any other relevant course that covers fundamental machine learning algorithms.

❑ Furthermore, this course assumes:

- Background in basic **probability**, **linear algebra**, and **calculus**.
- Good coding ability, corresponding to senior undergraduate CS major.
 - ✓ **Assignments will be in Python with PyTorch.**



What we don't cover in this class (1)

❑ Classical ML techniques and NLP algorithms

- E.g., SVM, HMM, CRF, Viterbi
- Take CSCI 5521 Machine Learning

❑ Advanced techniques for NLP models

- E.g., GNNs, Recursive Nets, Diffusion, latent variable models,
- Take CSCI 5980 NLP with Deep Learning

❑ Human aspects of AI and NLP systems

- E.g., interaction design, formative/field study, Qual analysis, theory grounding
- Take CSCI 5115 User Interface Design or other HCI classes



What we don't cover in this class (2)

❑ Linguistic aspects or computational linguistics:

- Take classes in Institute of Linguistics

<https://cla.umn.edu/linguistics/graduate/courses>

❑ Basic Python programming skills

- Basic tutorials on PyTorch and HuggingFace programming will be given by Tas



8980 vs 5980 vs 5541

- ❑ Some lectures will be shared across the three classes but they have different focuses;
 - 5980 (NLP with Deep Learning)
 - ✓ focuses on more "processing" parts of NLP, particularly with deep learning methods.
 - 8980 (Intro to NLP Research)
 - ✓ covers broad aspects of NLP research, including theory grounding, data annotation, error analysis, emerging topics, and applications to different fields.
 - 5541 (NLP, *this course*)
 - ✓ is an *introductory* class to cover some basic computational methods with applications and representations, and learn practical skills to apply them to your work
 - 89XX (Large Language models, *In Spring 2026*)
 - ✓ is an advanced course that cover a broad ranges of LLMs from technical knowledge, to latest learning methods, to applications, to societal impacts



Class Slack

- ❑ csci5541f25.slack.com/
- ❑ Slack will be the primary method of communication.
- ❑ A separate channel for UNITE/Remote students
- ❑ “prj-[A/B]-MENTOR-TEAMNAME”
 - With instructors
- ❑ “prj-[A/B]-TEAMNAME”
 - Without instructors

```
# announcement
# hw0
# hw1
# hw2
# hw3
# hw4
# project
# random
🔒 unite-and-remote-students
```

```
🔒 prj-b-zae-semanticons
```

Group Mentor Project
A/B name

```
▼ projects
🔒 prj-a-dk-golish-project
🔒 prj-a-dk-lm_bros
🔒 prj-a-dk-the-adversaries
🔒 prj-a-dk-transformative-attentors
🔒 prj-a-shirley-clickbait-analysis
🔒 prj-a-shirley-pilot
🔒 prj-a-shirley-verbavores
🔒 prj-a-zae-word-nerds
🔒 prj-b-dk-language-model-alignment
🔒 prj-b-dk-tbd
🔒 prj-b-shirley-title-fight
🔒 prj-b-shirley-word-wizards
🔒 prj-b-zae-flashcard-generator
🔒 prj-b-zae-nlp-vision
🔒 prj-b-zae-semanticons
🔒 prj-b-zae-vjk
```



















Slack communication etiquette







- ❑ Don't ask about homework via *emails* or *Direct Messages* to TAs;
 - All questions **MUST be posted in the homework channel** (e.g., hw#1, hw#2) or during TA's office hours.
- ❑ Technical / coding questions should follow the stack overflow format
 - <Error message> + context <what you tried so far>
- ❑ Responses outside office hours (9-6pm) will be delayed
- ❑ Be **respectful** and **supportive**
 - Anyone can respond to each other (Slack participation point). Feel free to answer others' questions if you have already addressed similar issues.










Canvas




<https://canvas.umn.edu/courses/518535>

Date	Details
Thu Sep 11, 2025	 HW1: Building MLP-based text classifier with pytorch
Thu Sep 18, 2025	 Team formation
Sun Sep 21, 2025	 HW2: Finetuning text classifier using HuggingFace
Thu Sep 25, 2025	 Project brainstorming
Sun Oct 5, 2025	 HW3: Authorship attribution using language models (LMs)
Thu Oct 9, 2025	 Proposal pitch
Tue Oct 14, 2025	 Proposal report
Sun Oct 19, 2025	 HW4: Generating and evaluating text generated from pretrained LMs
Sun Nov 2, 2025	 HW5: Prompting with large language models (LLMs)
Fri Nov 14, 2025	 Midterm office hour participation
Sun Nov 16, 2025	 HW6: Post-training LLMs for alignment
Thu Dec 4, 2025	 Poster presentation
Fri Dec 12, 2025	 Final report
	 In-person Class Participation
	 Office Hour Participation
	 Slack Participation

Homework
 HW1: Building MLP-based text classifier with pytorch Due Sep 11 at 11:59pm 10 pts
 HW2: Finetuning text classifier using HuggingFace Due Sep 21 at 11:59pm 10 pts
 HW3: Authorship attribution using language models (LMs) Due Oct 5 at 11:59pm 10 pts
 HW4: Generating and evaluating text generated from pretrained LMs Due Oct 19 at 11:59pm 10 pts
 HW5: Prompting with large language models (LLMs) Due Nov 2 at 11:59pm 10 pts
 HW6: Post-training LLMs for alignment Due Nov 16 at 11:59pm 10 pts

Project
 Team formation Due Sep 18 at 11:59pm 1 pts
 Project brainstorming Due Sep 25 at 11:59pm 1 pts
 Proposal pitch Due Oct 9 at 11:59pm 3 pts
 Proposal report Due Oct 14 at 11:59pm 5 pts

 Midterm office hour participation Due Nov 14 at 11:59pm 5 pts
 Poster presentation Due Dec 4 at 11:59pm 5 pts
 Final report Due Dec 12 at 11:59pm 10 pts

Participation
 Slack Participation 4 pts
 Office Hour Participation 4 pts
 In-person Class Participation 3 pts



Grading and Late Day Policy

- ❑ 60% Homework (hw1/2/3/4 for individual, hw5/6 for team)
- ❑ 30% Project (team)
- ❑ 10% Class Participation (individual)

- ❑ Each student will be granted **5 late days** to use for homework and project assignment over the duration of the semester. After all free late days are used up, penalty is 1 point for each additional late day.
- ❑ The late days and penalty will be applied to **all team members** for group homework and project.
- ❑ DRC and other medical urgencies can only be accommodated only with a *doctor's note or a letter from the DRC*



Class participation (10%)

Template: Weekly Participation on Slack (5): Weekly Participation in office hours (5): Weekly Participation in class (5): Scaled on 10 (with minimum 6/10):							
Weekly Participation Count on Slack	Weekly Participation on Slack (5 scaled)	Weekly Participation Count in office hours	Weekly Participation in office hours (5 scaled)	Weekly Participation in class (5 scaled)	Total (15)	Repeatedly Participating?	Total (scaled 10)
6	3	1		5	5	Very active	10

- ❑ Your class participation is **thoroughly** evaluated.
- ❑ The following metrics will be used to grade your participation:
 - Participation and discussion in class (no count for UNITE/Remote students)
 - **Discussion on Slack and during Office Hours for both instructor and TAs
 - Discussion and QA during the presentation of the project proposal and poster
- ❑ We explicitly **count the number of your offline and online participation**, and (min/max) normalize them at the end of the class.
- ❑ Your participation score will be zero if you haven't participated in class, Slack or other discussions.



Homework Details (60%)

- ❑ HW1, 2, 3 and 4 are individual assignments
- ❑ HW5 and 6 are team assignments (**the same team for projects**)
- ❑ Team assignments and projects
 - 3~4 people
 - Submit your team information by **Sept 18**
- ❑ All questions regarding homework **MUST be communicated with the lead TA** over Slack homework channels (e.g., #hw1) or during their office hours.



Homework Details (60%)

- ❑ The use of outside resources (books, research papers, websites), collaboration (students, professors) or AI tools (ChatGPT) are encouraged but **MUST be explicitly acknowledged with exact input/output from AI tools or collaboration**. Check out the Academic Integrity policy (<https://dykang.github.io/classes/csci5541/F25/index.html#note>).
- ❑ The deadline for all homework is by **midnight (11:59PM)** of the due date. Since our schedule is quite tight, there will be no deadline extension, but you can still use your late days. For the delayed **team homework and project**, late days for every team member will be used.



Use of AI Tools (e.g., chatGPT)

- ❑ **Encouraged** to use AI tools like ChatGPT, CoPilot, etc.
 - ❑ You can use it to:
 - Clarify concepts learned in class
 - Get instant answers to your questions before emailing TAs and instructor
 - Test your knowledge by asking it questions
 - ❑ **Warning:**
 - Please properly mention what kinds of help you get from the tools in your homework and project (missing acknowledgement will be treated as **cheating**)
 - Use the tools with responsibility:
 - ✓ Generative AI tools often **hallucinate** and only provide **general** answers
- Again, missing acknowledgement of AI usage will be treated as cheating**



Homework Details (60%)

- ❑ HW1: Building MLP-based text classifier with pytorch
 - 10 points, Individual, due: **Sep 11**
- ❑ HW2: Finetuning text classifier using HuggingFace
 - 10 points, Individual, due: **Sep 21**
- ❑ HW3: Authorship attribution using language models (LMs)
 - 10 points, Individual, due: **Oct 5**
- ❑ HW4: Generating and evaluating text from pretrained LMs
 - 10 points, Individual, due: **Oct 19**
- ❑ HW5: Prompting with large language models (LLMs)
 - 10 points, Team, due: **Nov 2**
- ❑ HW6: Post training LLMs for alignment
 - 10 points, Team, due: **Nov 16**





Homework Details (60%)

<https://dykang.github.io/classes/csci5541/F25/#homework>













Homework Details (60%)

All questions regarding homework **MUST be communicated with the lead TA via Slack homework channels** (e.g., #hw1, #hw2) or during their office hours. Homework 1, 2, 3, and 4 must be completed individually, while Homework 5 and 6 are team-based (maximum of 4 people). The same team must be used for both Homework 4/5 and the course project.

The use of external resources (books, research papers, websites, etc.) or collaboration (students, professors, ChatGPT, etc.) **must be clearly acknowledged in your report**. See the [notes](#) for academic integrity guidelines.

All homework is due by **midnight (11:59 PM)** on the due date. There are no extensions due to a tight schedule, but late days can still be used. For team assignments, late days will be deducted from each team member. Refer to the homework description  and Canvas link  for submission:

Homework Assignments & Deadlines

- **HW1:** Building MLP-based text classifier with PyTorch (10 points, Individual, due: **Sep 11 Thursday**) (, )
- **HW2:** Fine-tuning text classifier using HuggingFace (10 points, Individual, due: **Sep 21 Sunday**) (, )
- **HW3:** Authorship attribution using language models (LMs) (10 points, Individual, due: **Oct 5 Sunday**) (, )
- **HW4:** Generating and evaluating text from pretrained LMs (10 points, Individual, due: **Oct 19 Sunday**) (, )
- **HW5:** Prompting with large language models (LLMs) (10 points, Team, due: **Nov 2 Sunday**) (, )
- **HW6:** Post training LLMs for alignment (10 points, Team, due: **Nov 16 Sunday**) (, )



HW1: Building text classifier with pytorch

```
from torch import nn

class MLP(nn.Module):
    def __init__(self, vocab_size, embed_dim, num_class):
        super().__init__()
        self.embedding = nn.EmbeddingBag(vocab_size, embed_dim, sparse=True)
        num_layer = 100
        # self.fc = nn.Linear(embed_dim, num_class)
        self.fc1 = nn.Linear(embed_dim, num_layer)
        self.fc2 = nn.Linear(num_layer, num_class)

        # initialize the weights
        self.init_weights()

    def init_weights(self):
        initrange = 0.5
        self.embedding.weight.data.uniform_(-initrange, initrange)
        self.fc1.weight.data.uniform_(-initrange, initrange)
        self.fc1.bias.data.zero_()
        self.fc2.weight.data.uniform_(-initrange, initrange)
        self.fc2.bias.data.zero_()

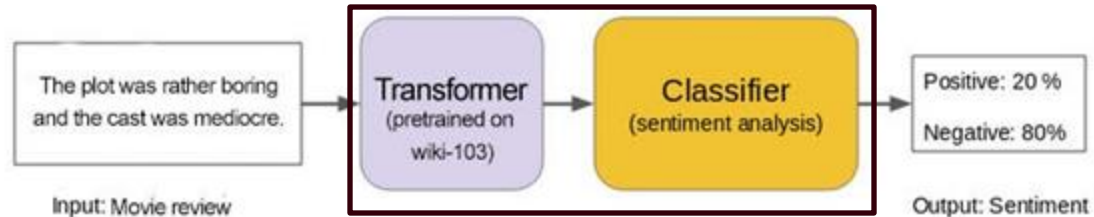
    def forward(self, text, offsets):
        embedded = self.embedding(text, offsets)
        return self.fc2(self.fc1(embedded))
```

end of epoch	1	time: 6.61s	valid accuracy	0.714
end of epoch	2	time: 6.69s	valid accuracy	0.672
end of epoch	3	time: 6.86s	valid accuracy	0.806
end of epoch	4	time: 6.69s	valid accuracy	0.829
end of epoch	5	time: 6.60s	valid accuracy	0.834
end of epoch	6	time: 6.68s	valid accuracy	0.809
end of epoch	7	time: 6.79s	valid accuracy	0.843
end of epoch	8	time: 6.81s	valid accuracy	0.850
...				
end of epoch	10	time: 6.61s	valid accuracy	0.844

Does my classifier learn something?



HW2: Finetuning text classifier using HuggingFace



Finetuning a pretrained text encoder

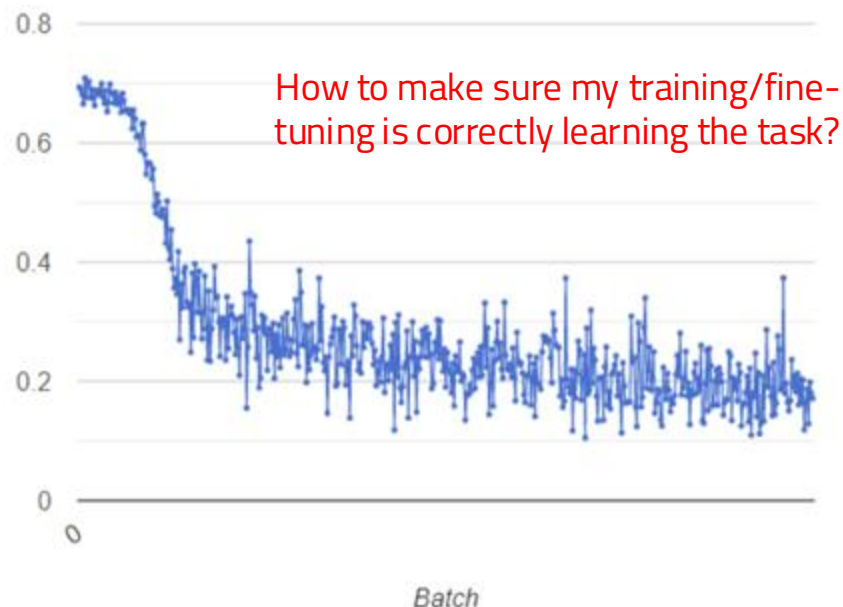


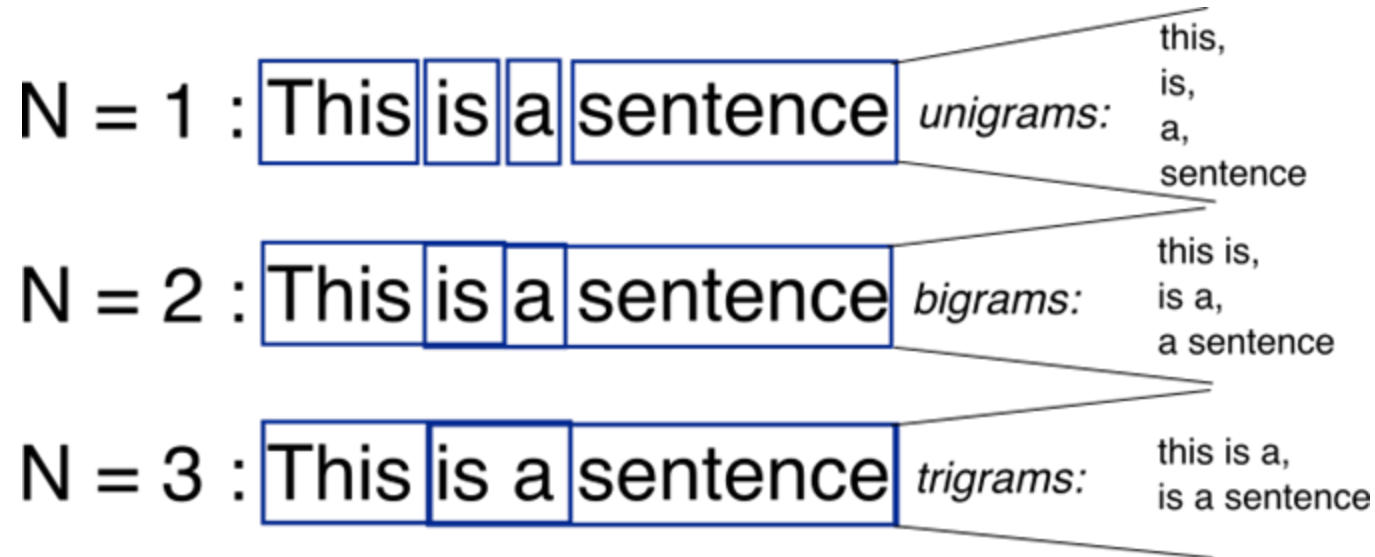
Table 1: Samples the model struggles with.

Sample	Ground-Truth Label	Predicted Label	Confidence
"This shit is fire"	Positive	Negative	0.870
"I love when my thoughts are definitely valued"	Negative	Positive	0.995
"Abso-fucking-lutely"	Positive	Negative	0.931
"Poggers"	Positive	Negative	0.895
"You are my favorite person to never see"	Negative	Positive	0.682

In which cases, does the fine-tuned classifier make mistakes?



HW3: Authorship attribution using language models



Tolstoy Samples

<s> i thought recalling the entrenchment and in the flank
had never thought nothing
perplexity : 499.5855375522907

<s> i burn my regiment and me five oclock then accept
for the innumerable
perplexity : 1072.2662726616034

How plausible is text as Tolstoy language?

<s> i thought recalling the entrenchment and in the flank
had never thought nothing

Written by Tolstoy? Or Written by Hamlet?



HW4: Generating and evaluating text from pretrained LMs

Los Angeles Times

World & Nation
Palestinian Authority joins International Criminal Court as 123rd member



Palestinian Authority President Mahmoud Abbas speaks during a meeting of the Foreign Ministers' Council in the West Bank city of Ramallah on Oct. 16, 2014. (Oleg Mironov / Associated Press)

Ground Truth Summarization

Membership gives the ICC jurisdiction over alleged crimes committed in Palestinian territories since last June . Israel and the United States opposed the move, which could open the door to war crimes investigations against Israelis .

Greedy Search	Beam Search	Top-K Sampling
The Palestinian Authority becomes the 123rd member of the International Criminal Court. The move gives the court jurisdiction over alleged crimes in Palestinian territories. Israel and the United States opposed the Palestinians' efforts to join the body. But Palestinian Foreign Minister Riad al-Malki says it is a move toward greater justice.	Palestinian Authority becomes 123rd member of the International Criminal Court. The move gives the court jurisdiction over alleged crimes in Palestinian territories. Israel and the United States opposed the Palestinians' efforts to join the body. But Palestinian Foreign Minister Riad al-Malki said it was a move toward greater justice.	The Palestinian Authority becomes the 123rd member of the International Criminal Court. The move gives the court jurisdiction over alleged crimes in Palestinian territories. Israel and the United States opposed the Palestinians' efforts to join the body. But Palestinian Foreign Minister Riad al-Malki said it was a move toward greater justice.

Different versions of generated text/summaries using various decoding algorithms

How to evaluate the generated text against the human's summary?

Algorithm	Meteor AVG
Greedy	37.12
Beam	38.33
Top-K	38.64
Top-P	38.14

Other aspects humans evaluate text summaries?

Average Coherence (out of 5)	Average Factuality (out of 5)
4.52	4.20
4.70	4.28
4.57	4.25
4.33	4.20



HW5: Prompting with large LMs



Q: Minneapolis and Saint Paul are known as the Twin Cities in Minnesota. What's the capital?

A: Saint Paul.

Q: New York is a large city in New York state. What's the capital?

A: Albany.

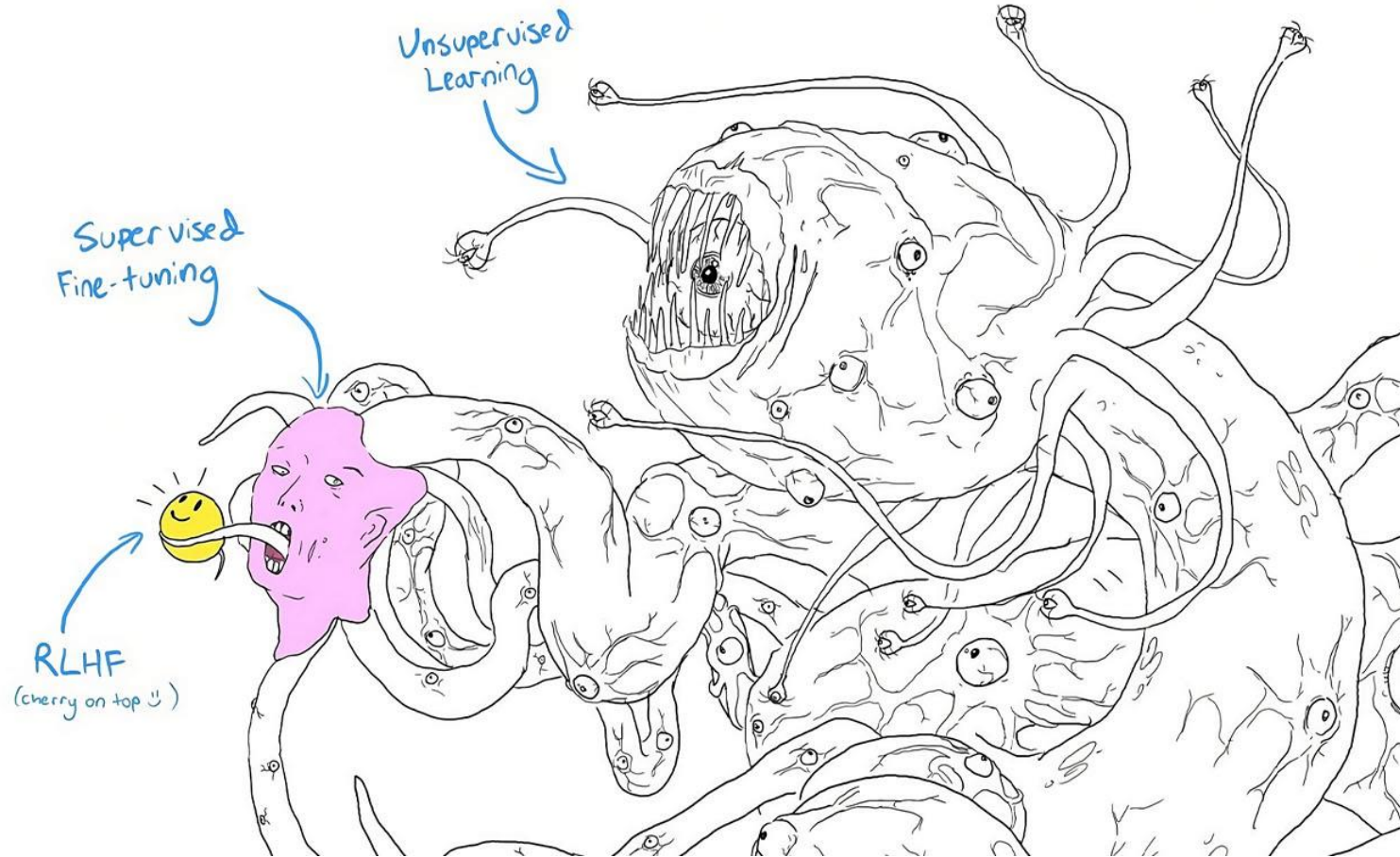
Q: Kansas City is another large city. Kansas City is the capital of which state?

A: Kansas ❌

Task Description	Instructions	Number of examples	Examples	Input Task	Expected Answer	Predicted Answer by GPT3
Convert movie titles into emoji.	N/A	0	N/A	Convert movies with emojis: The Hunger Games	👤🔪💀	🔥🍁🔪👤
Convert movie titles into emoji.	N/A	1	This is an example of converting a movie title into an emoji. Harry Potter and the Sorcerer's Stone: 🧙🔮💎	Convert movies with emojis based on the example: Harry Potter and the Goblet of Fire	🧙🔥🏆	🧙🔥🏆🐉🌟
Convert movie titles into emoji.	N/A	0	N/A	Convert movie titles into emoji: Full River Red	🌳🔴	🌳🌳🌳🌳❤️
Convert movie titles into emoji.	N/A	1	Full River Red 🌳🔴	Convert movie titles into emoji: From Vegas to Macau	🇲🇦👤🇲🇦	👤🇲🇦🇲🇦



HW6: Post-training of LLMs for alignment



Project Goal

- ❑ A course project would be one of the following types:
 - New research results judged suitable for acceptance to a NLP or ML workshop,
 - Critical analysis of existing NLP models or datasets,
 - Collection of your own dataset on new problems or adversarial datasets that can fool the existing systems ,
 - An in-depth literature survey on emerging topics,
 - Interactive demonstration (e.g., Chrome Extension, Flask) or visualization of existing systems,
 - New open-source repository or dataset with a high impact on the community
- ❑ Please carefully read the project guideline later once posted



Project Evaluation Rubric

- ❑ HWs are generously graded but the **projects are not!** Therefore, students should consider the potential contribution of the projects rather than **trying to play it safe**. Playing it safe won't give them full marks.
- ❑ Three important rubrics:
 - **Novelty**: Compared to the state-of-the-art methods/systems/datasets, how novel is your approach?
 - **Significance**: How strong is your result? Is your finding still holding if different setups or prompting tricks?
 - **Clarity**: How clear and easy-to-follow is your report? Do you have well organized presentation of your results and problem definition?
 - <https://jimtmoney.github.io/Courses/S25/rubrick.html>




Project Deliveries and Dues (30%)

- ❑ Team formation (1 point, due: Sep 18)
- ❑ Project brainstorming (1 point, due: Sep 25)
- ❑ Proposal pitch (3 points, due: Oct 7 and 9)
- ❑ Proposal report (5 points, due: Oct 14)
- ❑ Midterm office hour participation (5 points, due: Nov 14)
- ❑ Poster presentation (5 points, due: Dec 2 and 4)
- ❑ Final report (10 points, due: Dec 12)










Project Details (30%)













Please **carefully read the project description**  (To be updated) , as it contains essential details about expectations, deadlines, grading rubric, and FAQs. It is your responsibility to ensure you do not miss any of the information provided.

Each team (maximum 4 members) must submit their report, a link to the code (or a zipped archive), and presentation slides/poster on Canvas before the final deadline. Use the official ACL style templates via [Overleaf](#) or [GitHub](#).

Below are the required deliverables and deadlines (some fall on weekdays):

- **Team formation** (1 point, **Due: Sep 18**) 
- **Project brainstorming** (1 point, **Due: Sep 25**) 
- **Proposal pitch** (3 points, **Due: Oct 7 and 9**)  – Slide decks: Group A | Group B
- **Proposal report** (5 points, **Due: Oct 14**) 
- **Midterm office hour participation** (5 points, **Due: Nov 14**) 
- **Poster presentation** (5 points, **Due: Dec 2 and 4**) 
- **Final report** (10 points, **Due: Dec 12**)  | [Evaluation rubric](#)

Below are selected reports and posters from previous semesters. Some were extended into publications at top-tier venues:

- [CSCI 5541 S23] *Simulating Everyone's Voice: Exploring ChatGPT's Ability to Simulate Human Annotators*  
- [CSCI 5541 S23] *Vision & Language-guided Generalized Object Grasping*  
- [CSCI 5541 S23] *Generalizability of FLAN-T5 Model Using Composite Task Prompting*  
- [CSCI 5541 S23] *Comparing the Effectiveness of Fine-tuning vs. One-Shot Learning on the Kidz Bopification Task*  
- [CSCI 5980 F22] *Generating Controllable Long-dialogue with Coherence*  → **Published in AAAI 2024** 
- [CSCI 8980 S22] *Understanding Narrative Transportation in Fantasy Fanfiction*  → **Published in Workshop on Narrative Understanding (WNU) @ACL 2023** 



VLanGO Gh: Vision and Language guided Generalized Object Grasping

CSCI 554.1 Spring 2023
Nikhilanj Pelluri



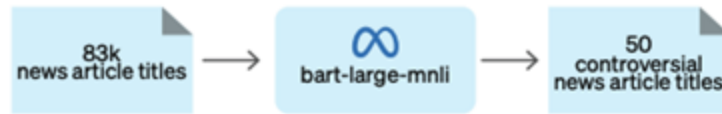
Simulating Everyone's Voice: Exploring ChatGPTs Ability to Simulate Human Annotators

CSCI 5541 Spring 2023

Abdirizak Yussuf, Claire Chen, Dinesh Challa, Venkata Sai Krishna

Step 1

Scraping and filtering data.



Step 2

Human annotation.

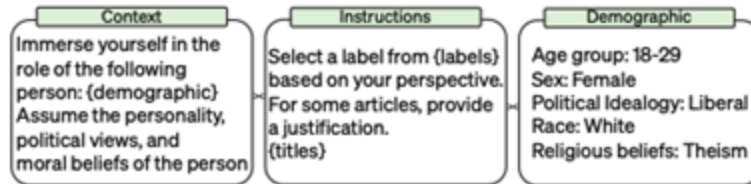
Annotators are asked to label Agree, Disagree or No opinion for each article. For 10 articles, they also provide a justification.



Step 3

ChatGPT annotation.

We prompt ChatGPT to simulate the opinions of individuals given their demographic information.



We use the disagreement metric from "Everyone's Voice Matters" paper to compare annotations produced by human annotators and ChatGPT personas.



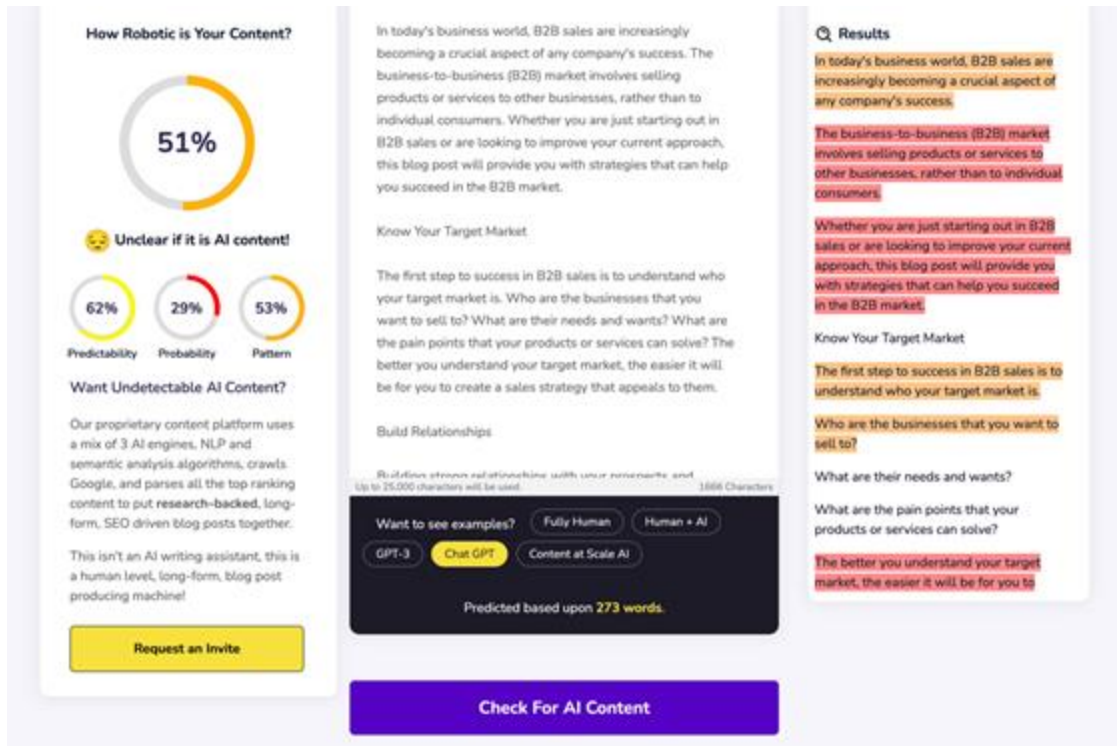
Topic	Human Annotators	ChatGPT Personas
Abortion	0.22	0.32
Immigration	0.15	0.40
Social Issues	0.11	0.40
Political Issues	0.017	0.50
Racial Justice	0.19	0.40
Religion	0.18	0.36
All Topics Combined	0.15	0.42

- **Human annotators: 0.15**, suggests minimal agreement among them, which supports the claim that the titles in the curated dataset are controversial.
- **ChatGPT personas: 0.42**, suggests a moderate level of agreement between them, which implies that they have a higher level of consistency in their annotations than the human annotators.



Who is speaking? Distinguishing Artificial Intelligence Generated and Human Written Text

CSCI 5541 Spring 2023
Moyan Zhou, Mingsheng Sun, Yutong Sun



RQ1: Do people **agree with each other** when distinguishing AI-generated and Human-written text?

Fleiss' Kappa
0.05 (p-value = 0.017)

RQ3: How does the existing tools work?

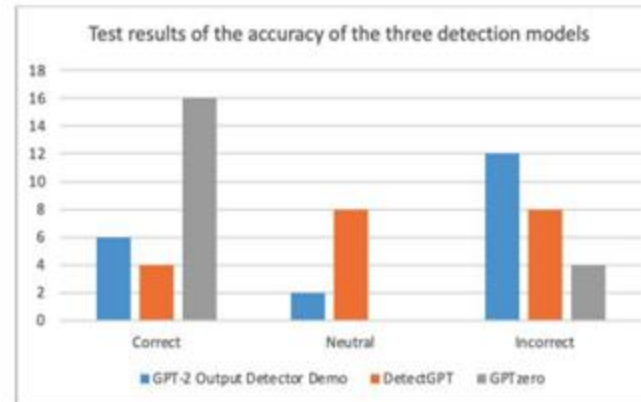


Figure 5. Test results of the accuracy of the three detection models

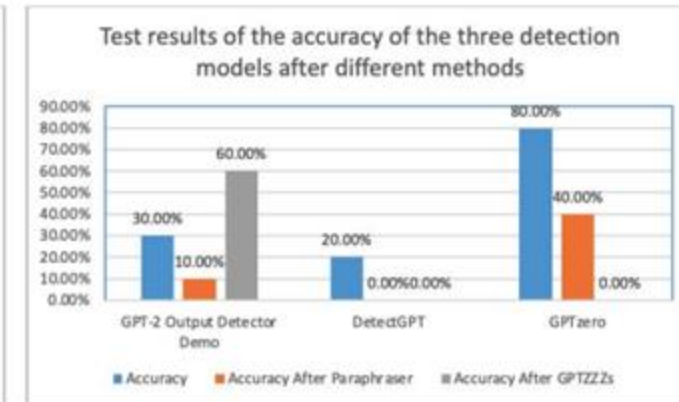


Figure 6. Test results of the accuracy after different methods



Comparing the Effectiveness of Fine-tuning vs. One-Shot Learning on the Kidz Bopification Task

CSCI 5541 Spring 2023

Jeonghoon Kim, Matthew Olson, Marco Berriodi

“I buy a new car for the b**** (For real)
I tear down the mall with the b**** (For real)”
from Bank Account by 21 Savage

“Lean with me, pop with me
Get high with me if you rock with me
Smoke with me, drink with me
F***** up liver with some bad kidneys”
from Lean Wit Me by Juice WRLD



You are a helpful assistant who is tasked with cleaning up songs. You will remove explicit material from a song's lyrics.

So we'll piss off the neighbors
In the place that feels the tears
The place to lose your fears
Yeah, reckless behavior
A place that is so pure, so dirty and raw
In the bed all day, bed all day, bed all day
Fucking and fighting on
It's our paradise and it's our war zone
It's our paradise and it's our war zone



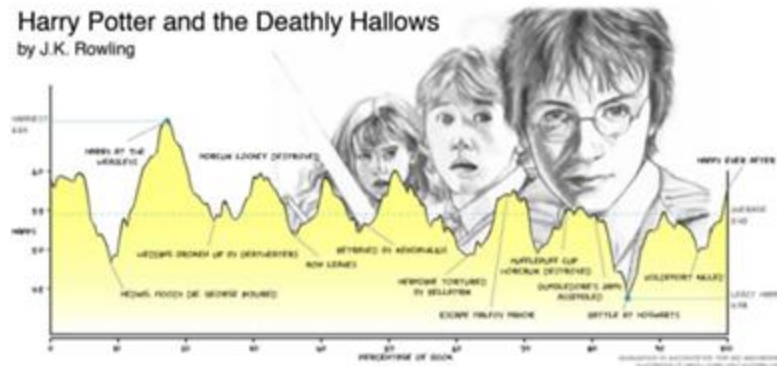
Here's a cleaned-up version of the lyrics:

We'll tick off the neighbors
In the place that feels the tears
The place to lose your fears
Yeah, wild behavior
A place that is so real, so gritty and raw
In the bed all day, bed all day, bed all day
Arguing and laughing on
It's our haven and it's our battleground
It's our haven and it's our battleground.



Understanding Narrative Transportation in Fantasy Fanfiction

CSCI 8980 Spring 2022
Kelsey Neis, Yu Fang



Select and highlight phrases that increase your sense of presence or connection to the story **blue** or decrease it.

He'd been dreaming of it since the defeat of Voldemort. The Veil, that is. The one that Sirius had fallen behind. The last time he'd dreamed about the Department of Mysteries, Sirius had died. The world had also finally woken up to the truth about Voldemort, but the price had been too high for him to be grateful. And once the truth was out, the attacks had gotten worse. He wondered, briefly, why he was there. Ginny was at home, she was pregnant, she needed him. Yet he could not seem to stay away. Voldemort was gone, true. But then, so were so many other people, good people, who should not have died. He counted the steps until he was standing right in front of where Sirius had fallen through. Cedric. Sirius. Dumbledore. Hedwig. Moody. Dobby. Tonks father. Remus. Colin Creevy. Tonks. Snape. Fred. Hell, even Crabbe didn't deserve to die then. There were more, many more, but none of them close to him.

Figure 1: Sample emotion arc for Harry Potter. (Reagan et al., 2016)

That was why he found himself, during his latest bout with insomnia, browsing a lesser known hero forum.	2.520
He'd found it several years earlier and quickly figured out that a lot of underground heroes used it to communicate with each other, since it offered encrypted chats and accounts were only known by random numbers, rather than usernames.	1.890
He'd spent about two weeks back then figuring out which accounts corresponded to which heroes, but he had never posted himself.	3.950
As Izuku drowsily scrolled through old posts, a crazy idea occurred to him.	4.580
If he couldn't be a hero himself, why couldn't he help the real heroes be better?	5.580
In the morning, he'd blame it on sleep deprivation and then promptly die of mortification, but that didn't change the fact that, at two o'clock in the morning, Izuku Midoria sent ten underground heroes in depth analyses of their quirks and fighting styles.	7.070

Published in Workshop on Narrative Understanding (WNU) @ACL 2023 <https://arxiv.org/abs/2306.04043>



Transidiomation: Optimizing translation of idioms embedded in text

CSCI 5541 Spring 2024
Dhondup Dolma, Jaeeun Lee, Yongtian Ou, Jiyeon Pyo Team
NLPitch

English sentences with PIE	GPT 3.5	Google Translate	SeamlessM4T
It was raining cats and dogs last weekend.	지난 주말에는 개구리가 비 내리는 것 같았어요. It was like frogs were raining last weekend. (hallucination)	지난 주말에는 고양이와 개에게 비가 내렸습니다. Cats and dogs were rained on last weekend. (incorrect)	지난 주말에는 고양이와 개가 비를 내고 있었다. Not translatable
Getting that done was a piece of cake.	그것을 해내는 것은 쉬운 일이었다. It was easy to get that done.	그 일을 끝내는 건 아주 쉬운 일이었어 It was easy to get that done.	그 일을 처리하는 것은 쉬운 일이었습니다. It was easy to get that done.
Teaching is my bread and butter.	가르치는 것이 내 소득원이다. Teaching is my source of income.	가르치는 것이 나의 빵과 버터이다. Teaching is my bread and butter. (literal translation)	가르치는 게 내 과 버터야. Not translatable

Table 1: Examples of various NMT models translating English sentences with PIE to Korean. Text that have red Xs are incorrect translations.

“Translate the sentence into target language by following these steps:

Step1. Identify the idiom.

Step2. Find an idiom with the same meaning in the target language. If there is no equivalent idiom, give the figurative meaning of it.

Step3. Include response from Step2 to translate the sentence.”

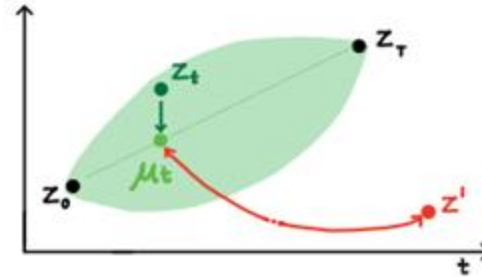


Generating Controllable Long-dialogue with Coherence

CSCI 5980 Fall 2022

Zhecheng Sheng, Chen Jiang and Tianhao Zhang

Time control in language model using Brownian bridge (Wang et al., ICLR 2022)



x_0 : [USER] Hello, I'd like to buy tickets for tomorrow.

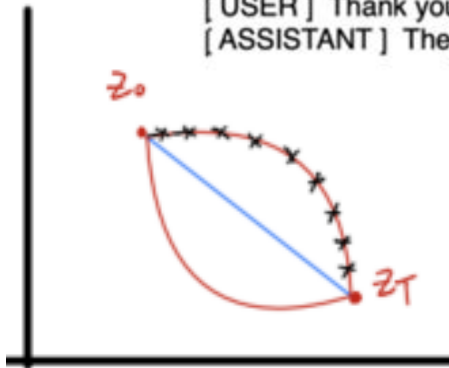
x_t : [ASSISTANT] What movie theater do you prefer?

x_T : [USER] Could you confirm my tickets just in case?

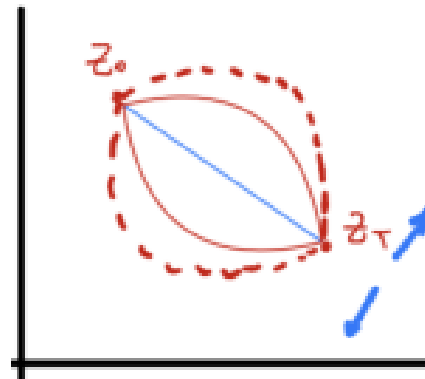
x' : [USER] Hi, I'm looking to purchase tickets for my family.

$$\mathcal{L} = -\log \frac{\exp(d(z_t, \mu_t))}{\exp(d(z_t, \mu_t)) + \exp(d(z', \mu_t))}$$

[USER] I am thinking about seeing a movie tonight, please.
[ASSISTANT] What movie do you have in mind?
[USER] The Elizabeth Theatres.
[ASSISTANT] They have 2 tickets available.
[USER] Thank you.
[ASSISTANT] They have been waiting in the line for you.



[USER] I would love to get to the movies tonight.
[ASSISTANT] OK. And where will you be seeing the movie?
[USER] Creek's End, Oregon.
[ASSISTANT] Creek's End, Oregon. Got it. Is there a particular movie you have in mind?
[USER] No wait, the visuals are so darned.
[ASSISTANT] No problem.
[USER] No problem.
[ASSISTANT] No problem.
[USER] No problem.
[ASSISTANT] No problem.
[ASSISTANT] No problem.




Published in AAAI 2024, <https://arxiv.org/abs/2312.16893>



Topics

Schedule

We will cover basic NLP representations $g(x)$, to build text classifiers $P_{\theta}(y|g(x))$, language models $P_{\theta}(g(x))$, and large language models $P_{\theta}(g(x))$. Based on knowledge you gain during the class, your team will develop your own NLP systems during the semester-long project. Pay attention to due dates and homework release. Lecture slides and homework/project description will be available in .

Date	Lectures and Dues	Readings
Sep 2	Class Overview HW1 out 	
Sep 4	Intro to NLP	
Sep 5	Recitation on computing basics (TBD) <ul style="list-style-type: none">• Colab+JupyterNotebook Tutorial  	















Topics

□ Text classification

- Programming Tutorials
- Finetuning Tutorial

□ Basic representations

- Distributional semantics
- Contextualized word embeddings
- Self-Attentions and Transformers






Sep 9	Text Classification  HW2 out 	<ul style="list-style-type: none">• Determining the sentiment of opinions• Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank• From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series• Style is NOT a single variable: Case Studies for Cross-Style Language Understanding• Beyond Accuracy: Behavioral Testing of NLP Models with CheckList
Sep 11	Text Classification: Practices Tutorial on Scikit-Learn and PyTorch (TBD)  • Scikit-Learn  • Pytorch  Tutorial on Finetuning & vLLM (TBD)  • Huggingface  • vLLM  HW1 due	<ul style="list-style-type: none">• Text classifier with NLTK and Scikit-Learn• Blog post on Pre-training vs Fine-tuning in LLM: Examples• Tutorial on Text classification using HuggingFace's Transformers• vLLM
Sep 16	Word Embeddings  Project description out 	<ul style="list-style-type: none">• From Frequency to Meaning: Vector Space Models of Semantics• Efficient Estimation of Word Representations in Vector Space• Linguistic Regularities in Continuous Space Word Representations• GloVe: Global Vectors for Word Representation• Retrofitting Word Vectors to Semantic Lexicons• Gensim's word2vec tutorial
Oct 2	Contextualized Word Embeddings  HW3 due	<ul style="list-style-type: none">• Deep contextualized word representations• BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding• A Primer in BERTology: What we know about how BERT works
Oct 14	Transformers  Proposal Report due	<ul style="list-style-type: none">• Attention is All you Need• Tutorial on Illustrated Transformer• Language Models are Unsupervised Multitask Learners• Language Models are Few-Shot Learners• Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer



Topics (cont'd)

□ Language Models

- Ngram LM, Neural LM, RNN, LSTM, and seq2seq
- Search and decoding
- Search in Training
- Evaluation
- Applications







Sep 18	Language Models (1): Models  ColabPro Reimbursement  HW3 out  HW2 due Project Team Formation due	<ul style="list-style-type: none">• Chapter 3 of Jurafsky and Martin• A Neural Probabilistic Language Model• Recurrent neural network based language model• Long Short-Term Memory• Multivariable chain rule, simple version• Long Short-Term Memory• Sequence to Sequence Learning with Neural Networks
Sep 23	Language Models (2): Search 	<ul style="list-style-type: none">• The Curious Case of Neural Text Degeneration• Mutual Information and Diverse Decoding Improve Neural Machine Translation• Sequence Level Training with Recurrent Neural Networks• An Actor-Critic Algorithm for Sequence Prediction
Sep 25	Language Models (3): Evaluation  Project Brainstorming due	<ul style="list-style-type: none">• Neural Machine Translation by Jointly Learning to Align and Translate• Perplexity of fixed-length models• BLEU: a Method for Automatic Evaluation of Machine Translation• ROUGE: A Package for Automatic Evaluation of Summaries



Topics (cont'd)

□ Large language models:

- Pretraining and scaling laws
- Prompting
- Instructing and augmenting LLMs
- LLMs as agents
- Alignment
- Efficiency
- Reasoning

Oct 16	Pretraining and Scaling  HW4 due	<ul style="list-style-type: none"> • Scaling Laws for Neural Language Models • On the Opportunities and Risks of Foundation Models • On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?
Oct 21	Prompting  HW5 out 	<ul style="list-style-type: none"> • Chain-of-Thought Prompting Elicits Reasoning in Large Language Models • Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing • Prefix-Tuning: Optimizing Continuous Prompts for Generation
Oct 23	Instructing and Augmenting LLMs 	<ul style="list-style-type: none"> • Training language models to follow instructions with human feedback • Augmented Language Models: a Survey • Toolformer: Language Models Can Teach Themselves to Use Tools • Internet-augmented language models through few-shot prompting for open-domain question answering
Oct 28	LLMs as Agents 	<ul style="list-style-type: none"> • ReAct: Synergizing Reasoning and Acting in Language Models • MetaGPT: Meta Programming for A Multi-Agent Collaborative Framework • Generative Agents: Interactive Simulacra of Human Behavior • WebArena: A Realistic Web Environment for Building Autonomous Agents
Oct 30	Efficiency  HW5 due	<ul style="list-style-type: none"> • Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding • LLM.int8(): 8-bit Matrix Multiplication for Transformers at Scale • Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer • The Power of Scale for Parameter-Efficient Prompt Tuning
Nov 4	Alignment  HW6 out 	<ul style="list-style-type: none"> • Learning to summarize from human feedback • Deep Reinforcement Learning from Human Preferences • Proximal Policy Optimization Algorithms • Direct preference optimization: Your language model is secretly a reward model • Open Problems and Fundamental Limitations of Reinforcement Learning from Human Feedback • Dynamic Multi-Reward Weighting for Multi-Style Controllable Generation • Benchmarking Cognitive Biases in Large Language Models as Evaluators
Nov 6	Reasoning 	<ul style="list-style-type: none"> • Understanding Reasoning LLMs • DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning • WebGPT: Browser-assisted question-answering with human feedback • Training Verifiers to Solve Math Word Problems • Easy-to-Hard Generalization: Scalable Alignment Beyond Human Supervision









Topics (cont'd)

□ Human-centric approaches

- Interpretability
- Evaluation
- Data
- Human-AI interaction

□ Human-centric NLP

□ Concluding Remarks

Nov 11	Interpretability 	<ul style="list-style-type: none">• Interpreting Language Models with Contrastive Explanations• BERT Rediscovered the Classical NLP Pipeline• Zoom In: An Introduction to Circuits• INTERPRETABILITY IN THE WILD: A CIRCUIT FOR INDIRECT OBJECT IDENTIFICATION IN GPT-2 SMALL• Transformer Feed-Forward Layers Build Predictions by Promoting Concepts in the Vocabulary Space• Towards Monosemanticity: Decomposing Language Models With Dictionary Learning
Nov 13	Evaluation  HW6 due Project midterm office hour due	<ul style="list-style-type: none">• HellaSwag: Can a Machine Really Finish Your Sentence?• MEASURING MASSIVE MULTITASK LANGUAGE UNDERSTANDING• Training Verifiers to Solve Math Word Problems• Evaluating Large Language Models Trained on Code
Nov 18	Data 	<ul style="list-style-type: none">• Annotation Artifacts in Natural Language Inference Data• Dataset Cartography: Mapping and Diagnosing Datasets with Training Dynamics• Everyone's Voice Matters: Quantifying Annotation Disagreement Using Demographic Information• ChatGPT Outperforms Crowd-Workers for Text-Annotation Tasks
Nov 20	Human-AI Collaboration 	
Nov 25	Human-centric NLP  Concluding Remark 	



There is a chance to skip one of the lectures due to international travels
Some lectures will be taught by guest speakers



Topics (cont'd)

□ In-class project sessions:

- Project guideline
- Project Proposal Pitch (Group A and B)
- Final Project Poster (Group B and A)

Oct 7	Project Proposal Pitch (1)	<u>Slides Deck for Group A</u> <ul style="list-style-type: none">•
Oct 9	Project Proposal Pitch (2)	<u>Slides Deck for Group B</u> <ul style="list-style-type: none">•
Sep 30	Project Guideline  HW4 out 	
Nov 27	No class (Thanksgiving)	
Dec 2	Final Project Poster (1)	<u>Project Pages for Group B</u> <ul style="list-style-type: none">•
Dec 4	Final Project Poster (2) Final Project Report due (Dec 12)	<u>Project Pages for Group A</u> <ul style="list-style-type: none">•



Textbook

- ❑ The lecture slides and reading list will be posted on the class site at least one day before the class.
- ❑ Textbook is **not required** but the following books are primarily referred
 - Jurafsky and Martin, Speech and Language Processing, 3rd edition [[online](#)]
 - These course materials are inspired by the slides from Stanford, UC Berkeley, and CMU.



Academic Integrity

- ❑ Everything you turn in must be your own work, and you must note **the** names of **anyone** you collaborated with on each problem and cite **any resources** (e.g., chatGPT prompts you used) that you used to learn about the problem.
- ❑ If you have any doubts about whether a particular action may be construed as cheating, ask the instructors for clarification before you do it. Cheating in this course will result in a grade of **F** for course and the [University policies](#) will be followed.



Computing Resources

- ❑ Google Colab Pro account by CS&E department
 - Current roster has been granted
- ❑ MSI account
 - Current roster has been granted
- ❑ LLM API Endpoints (Updated soon)



HW1: Building MLP-based text classifier with pytorch

Due: Sep 11 (Thur) CSCI 5541 (F25) HW1: Building text classifier with pytorch

page 1 of 2

The lead TA for this assignment is Shuyu Gan (gan00067@umn.edu). Please communicate with the lead TA via Slack or office hours. All questions MUST be discussed in the homework channel (#HW1). Questions through emails, Direct Messages, and other channels will not be answered.

The goal of this assignment is to make sure you get used to machine learning (ML) programming with [PyTorch](#) and implement a simple neural network based text classifier. By following the steps below, you can train your own classifier from scratch. This assignment will be the basis of your next assignment and class project that require more advanced Python/PyTorch programming, analysis, and deep learning knowledge.

First, carefully read tutorial slides and notebooks presented by the TAs on [Scikit-learn and Pytorch](#), and try to run the same scripts on your local machine with [Jupyter Notebooks](#) in [Google Colab](#). In the future, we will be running jobs that will require Google Colab Pro. For now, this assignment is small enough such that the free tier of Google Colab should be sufficient. For future assignments, we will be switching to Colab Pro and providing documentation for you to set this up.

In the tutorials, we developed a multi-layer perceptron (MLP) based binary classifier for predicting whether a tweet is about a real disaster or not. Now, let's build a simple text classifier using PyTorch. In this homework, you will simply stack one more layer to your MLP and develop a **two-layer MLP text classifier** using Pytorch on a **new dataset**.

Step 1: Choose a dataset from TorchText.datasets

You can choose any dataset from PyTorch's [torchtext](#). If you are using a laptop or local machine, choose a small dataset, such as [IMDb](#) (TRAIN/TEST: 25000/25000 samples) or [SST2](#) (TRAIN/TEST: 67349/1821 samples). The TRAIN split is used to train your model, and TEST split is used to evaluate the trained model's performance. Your TEST set must not be used in any way during training. Below is an example script for loading the original [IMDb](#) dataset.

