CS 2731 Introduction to Natural Language Processing

Session 28: Final project presentations

December 13, 2023

University of Pittsburgh | School of Computing and Information

Course logistics

- Final project reports due **tomorrow, Thu Dec 14, 11:59pm**
- *Thanks for a great semester!*

Schedule

1. Shijia, Shuhao, Lokesh, Vincent
2. Tom, Max, Ben
3. Bhiman, Aziz, Atharva
4. Qikun, Ming, Jiyuan
5. Lingwei and Yuhang
6. Gina, Modhumonty, Norah
7. Ahana, Dhanush, Yixiao
8. Birju and Robbie
9. Qichang, Yuxuan, Haoyu
10. RJ and Jacob
11. Marcelo and Connor

Instructions

- Plan for **5 min max** presentations + a brief Q&A
- Cover at least these key points
  - Project motivation (briefly?)
  - Data
  - Methods, or annotation/collection approach for dataset projects
  - Results
- Put your slides in this presentation after your project name slide by **class session, 2:30pm on Wed Dec 13**

1. Shuhao, Lokesh, Vincent

Introduction

- **Motivation**

  Build a model to summarize a movie based on its subtitles

- **Data**
  - **CMU Movie Summary Corpus:**

    Dataset contains summaries for 42,306 movies.

  - **Subtitles:**
    - Scraping opensubtitles for movies that are present in the Movie Corpus - 1322 subtitles
    - Challenges:
      - Movies with same name but released in different years
        Ex: The Message (1976, 2006, 2009, 2009, 2011, 2012, 2015, 2020, 2022)
      - Same name but different languages
      - API rate limits

Training model

1. **Subtitle Preprocessing**: Clean and format subtitles.
2. **Reading Subtitles**: Load subtitles from various formats.
3. **Loading Data**: Import movie information and summaries.
4. **Subtitle Processing**: Preprocess and store movie subtitles.
5. **Text Preprocessing**: Convert text to lowercase, remove spaces.
6. **Training Data** Preparation: Combine subtitles and summaries.
7. **BERT Tokenization**: Tokenize text using BERT tokenizer.
8. **Data Tokenization**: Tokenize subtitles and summaries.
9. **BERT Model Setup**: Configure and load BERT model.
10. **Model Building**: Create a deep learning model with BERT and LSTM.
11. **Text Conversion**: Process text for model compatibility.
12. **Data Generator**: Generate training data batches.
13. **Model Training**: Train the model with subtitle and summary data.
14. **Model Saving**: Save the trained model.

Extractive Summarization

Since the subtitle contains too much data that may exceed the limits of summarization model. We need to modulate the raw data before using it.

1. **Data Pre-process**

   Selectively remove the uppercase, stopword, punctuation, number, etc.

2. **Ranking data**

   Based on weight parameters to select top x sentences

Model Result

The training result shows selected sentences:

(x=5, clearn_number=True, remove_stopword=False, clean_punctuations=True, lowercase=True)

```
0    welcome to our anti-gravity research project
1    i am like a blade destined to fight for his maj...
2    i want to prove it's not a myth but something...
3    chancellor wants you dead
4    from now on we will never separate again...
Name: extractive_summarized_text, dtype: object
```

2. Tom, Max, Ben

Introduction

- Project Goal: to use already available patient EHR data to generate discharge notes without the need for a doctor to do this manually
    - This will allow doctors to allocate their time to more useful tasks and shorten the patient discharge procedure
- Related work attempts to generate patient notes by using demographics, medications, labs, and past notes
    - To generate, they indicate the intended note type and a hint about the current note (its first 10 tokens)
        - *Learning to Write Notes in Electronic Health Records* by Liu, et al. 2018

Methodology

- Objective: generate unstructured discharge summaries from structured EHR records (charts, inputs, labs, procedures)
- End-to-end generation from preprocessed EHR using pretrained LLM
- Diagnosis classification from hidden representations
- Pretrained models: BioClinical Bert, GPT-2, Flan-T5
- GPT-2 and Flan-T5: finetuned on MIMIC-III
- Baseline model: zero-shot gpt-3.5-turbo

Data

- MIMIC-III dataset
- EHR data of hospital stays, including medication, lab results, physiological signals, procedures, etc
- Contains note events like progress notes, lab reports, discharge summaries, etc. Also has diagnoses in ICD-9
- Used by several related work

Preprocessing

- Features: chart events, lab events, input events, procedure events
- Features are selected based on frequency and threshold
- Multi-hot encoded binary vectors, 512 dimensional
- Initially featurized with temporal dimension for classification task
- Summarized through time for text generation task
- Model finetuning and text generation length limited to 512

Evaluation

- Evaluation metrics:
    - BLEU Score: overlap of n-grams between generated and reference text
    - ROUGE-2: uses bigrams to evaluate how well the generated text captures important phrases from the reference text
    - ROUGE-L: measures the longest common sequence between the generated and reference text
- Same metrics used by the paper which generated discharges notes given all patient information including existing discharge notes
- Allows us to use their results as an upper bound of what we could expect from our approach
- Zero-shot gpt-3.5-turbo baseline:
    - "Write a patient discharge summary based on the following information: [input]”

Results

	BLEU	ROUGE-2	ROUGE-L
GPT-2	0.1750	0.1205	0.2455
Flan-T5	0.2248	0.1457	0.3045
Liu, et al. 2018	n/a	0.3306	0.5942
gpt-3.5-turbo (zero-shot)	0.0004	0.0131	0.0969

Generated: Admission Date: [**2177-5-23**] Discharge Date: [**2177-5-28**] Date of Birth: [**2135-8-29**] Sex: M Service: SURGERY Allergies: Patient recorded as having No Known Allergies to Drugs Attending:[**First Name3 (LF) 371**] Chief Complaint: Motor vehicle crash Major Surgical or Invasive Procedure: None History of Present Illness: 21 yo male s/p motor vehicle crash, unrestrained, +LOC, unresponsive at scene. He was transported to [**Hospital1 18**] for further care. Past Medical History: Denies Family History: Noncontributory Pertinent Results: [**2177-5-23**] 10:15AM ASA-NEG ACETAMINOPHEN-NEG bnzodzpn-NEG barbitrt-NEG tricyclic-NEG [**2177-5-23**] 10:15AM ASA-NEG ACETAMINOPHEN-NEG bnzodzpn-NEG barbitrt-NEG tricyclic-NEG ...

Label: Admission Date: [**2182-1-26**] Discharge Date: [**2182-2-8**] Date of Birth: [**2153-12-30**] Sex: M Service: NEUROSURGERY Allergies: No Known Allergies / Adverse Drug Reactions Attending:[**First Name3 (LF) 1271**] Chief Complaint: 28 y/o M unresponsive after MVA Major Surgical or Invasive Procedure: None History of Present Illness: 28 M unrestrained passenger in MVA. Patient found under dashboard of car. Unresponsive, taken to OSH where noted GCS 6. Patient received induction medications for intubation and lorazepam. CT head, c-spine, chest/abd/pelvis done, patient transferred for higher level of care. On admission, patient not responsive, motor exam abnormal. Neurosurgery called for consult ...

Baseline (gpt-3.5) generation

The patient presented with multiple abnormal laboratory findings and medical interventions during their hospital stay. The patient's initial laboratory results showed low levels of Albumin, Hematocrit, WBC, Platelet Count, Ionized Calcium, Lactic Acid, and Glucose finger stick, as well as high levels of Anion gap, Prothrombin time, PTT, INR, Fibrinogen, Phosphorous, Arterial Blood Pressure (systolic and diastolic), Total Bilirubin, TCO2 (calc) Arterial, AST, Chloride (serum), and Arterial O2 pressure.

During their hospital stay, the patient received various medications and interventions including Morphine Sulfate, NaCl 0.9%, Vancomycin, Gastric Meds, NaCl 0.45%, Potassium Chloride, Calcium Gluconate, Packed Red Blood Cells, Platelets, PO Intake, Dextrose 5%, GT Flush, Solution, Sterile Water, Insulin - Regular, Dextrose 50%, Insulin - Glargine, Insulin - Humalog, D5 1/2NS, LR, Piggyback, K Phos, Fresh Frozen Plasma, Fentanyl, Phenylephrine, Metoprolol, Heparin Sodium (Prophylaxis), Pre-Admission Intake, Magnesium Sulfate, OR Crystalloid Intake, Cefepime, Albumin 25%, OR Packed RBC Intake, KCL (Bolus), Magnesium Sulfate (Bolus), Norepinephrine, Propofol, Metronidazole, Furosemide (Lasix), Nitroglycerin, Famotidine (Pepcid), and Pantoprazole (Protonix).

The patient also underwent various medical procedures including Invasive Ventilation, Intubation, Multi Lumen, PICC Line, 18 Gauge, Transthoracic Echo, CT scan, Extubation, X-ray, Ultrasound, Sputum Culture, Nasal Swab, Urine Culture, PA Catheter, Chest X-Ray, OR Received, Arterial Line, Bronchoscopy, Blood Cultured, and EKG.

Throughout their hospital stay, the patient experienced fluctuating levels of laboratory parameters including Hemoglobin, Creatinine, Arterial CO2 Pressure, Alkaline Phosphate, Glucose (serum), Fspn High, PH (Arterial), BUN, Calcium non-ionized, LDH, Inspired O2 Fraction, Magnesium, O2 Flow, ALT, Sodium (serum), Troponin-T, Ventilator Mode, Differential-Lymphs, Differential-Neuts, Glucose (whole blood), Potassium (serum), HCO3 (serum), Direct Bilirubin, pCO2, pH, pO2, Potassium (Whole Blood), Lactate Dehydrogenase (LD), Fibrinogen (Functional), NTproBNP, Hematocrit, Hemoglobin, Phosphate, and Potassium.

Upon evaluation and treatment, the patient's laboratory parameters started to stabilize and return to within normal ranges. The patient's condition improved, symptoms resolved, and they were deemed stable for discharge. The patient was provided with appropriate medication instructions, follow-up care plan, and advised to seek medical attention if any symptoms reoccur or worsen.

Conclusion

Improves over zero-shot gpt-3.5

Falls behind methods with more diverse, hand-crafted features

Limitations:

- Computation and data efficiency
- Input and output length
- Input flexibility

3. Bhiman, Aziz, Atharva

Fairness Analysis of Human/AI-Generated Summaries of Student Reflections

Bhiman Kumar Baghel

Abdulaziz Alotaibi

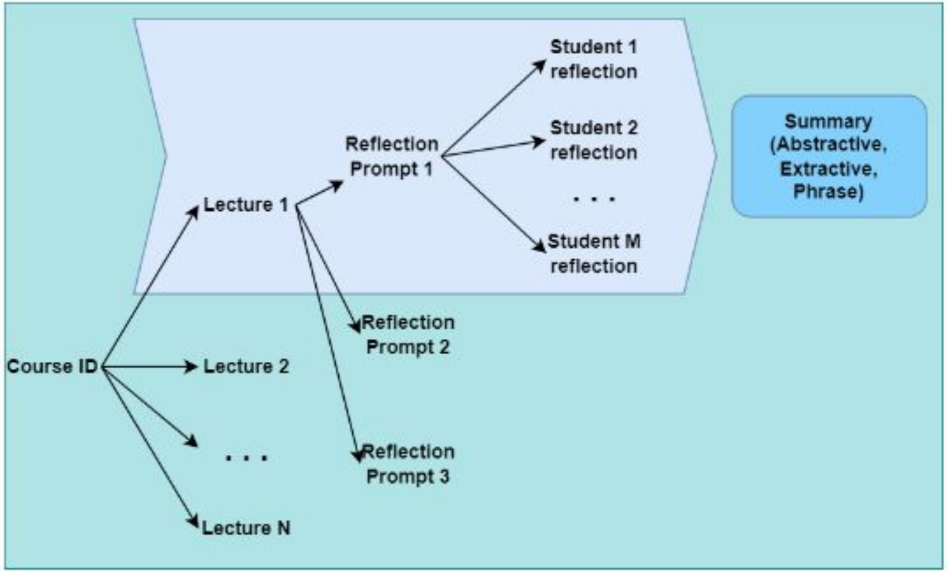
Atharva Vichare

Research Plan

1. Data preprocessing & Statistical Analysis.
   - Goal: Cleaning, Merging, Structuring.
2. Topic Modeling - LDA and BERTopic.
   - Goal: Analyze topic distribution among genders.
3. Predictive Modeling - Logistic Regression, SVM, Naive Bayes.
   - Goal: Analyze whether patterns exists among genders.

Data preprocessing & Statistical Analysis

- Data from different Datasets was merged. Created about 12k reflection entries.

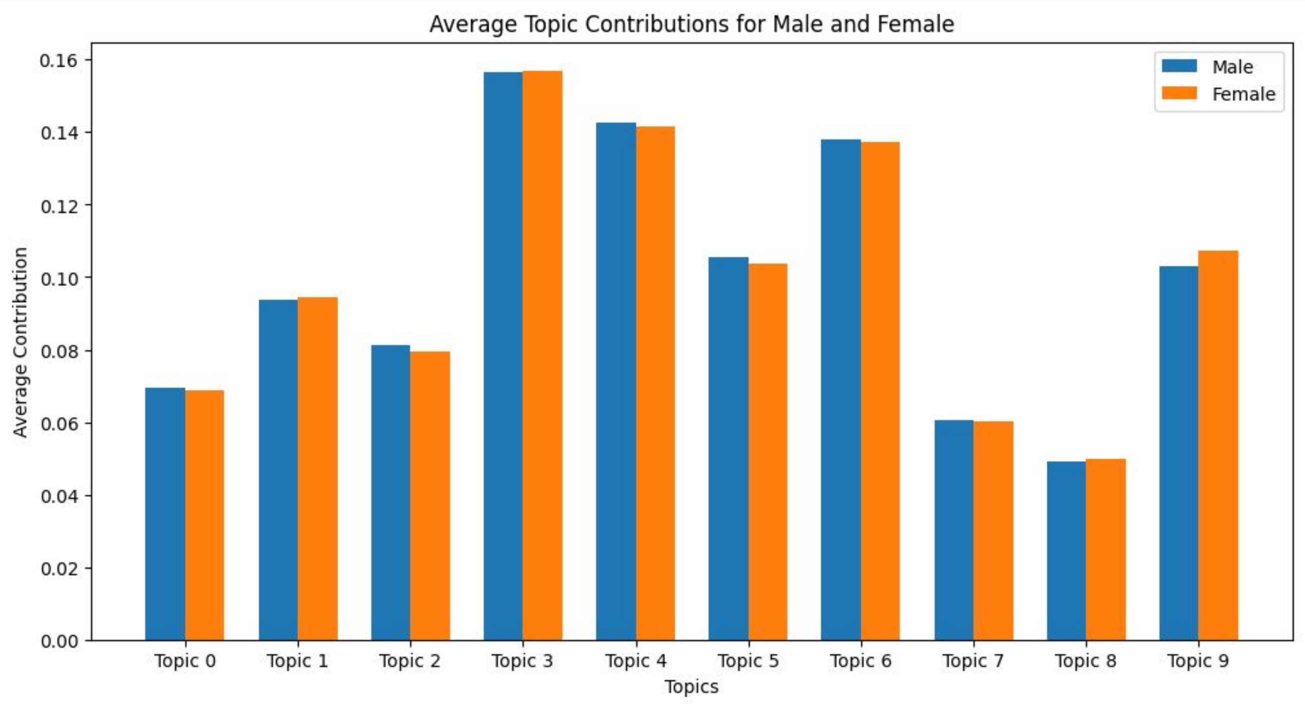


Data preprocessing & Statistical Analysis

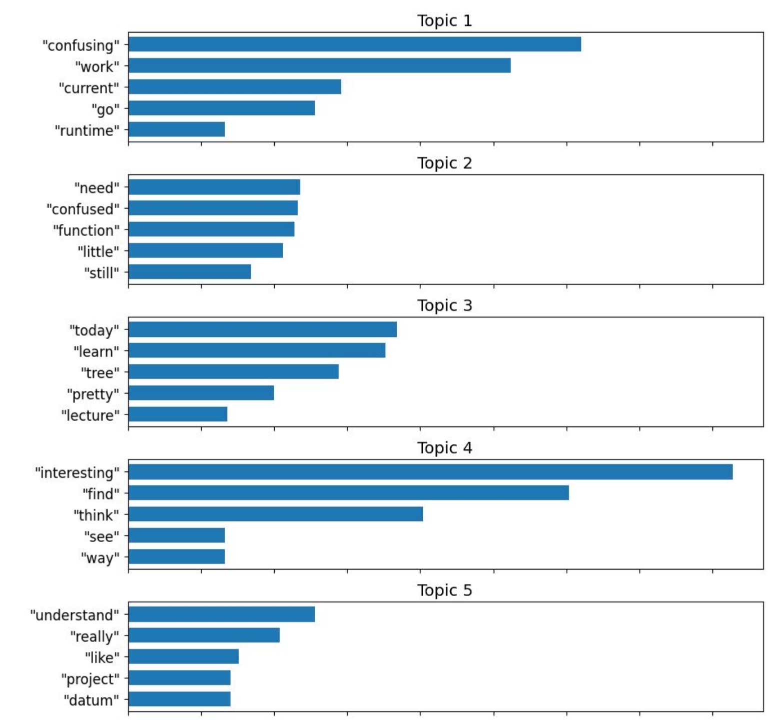
COURSE	Reflection Count.
CS	1863
ENGINEERING	2435
INFO_SCIENCE	776
PHYSICS	7296

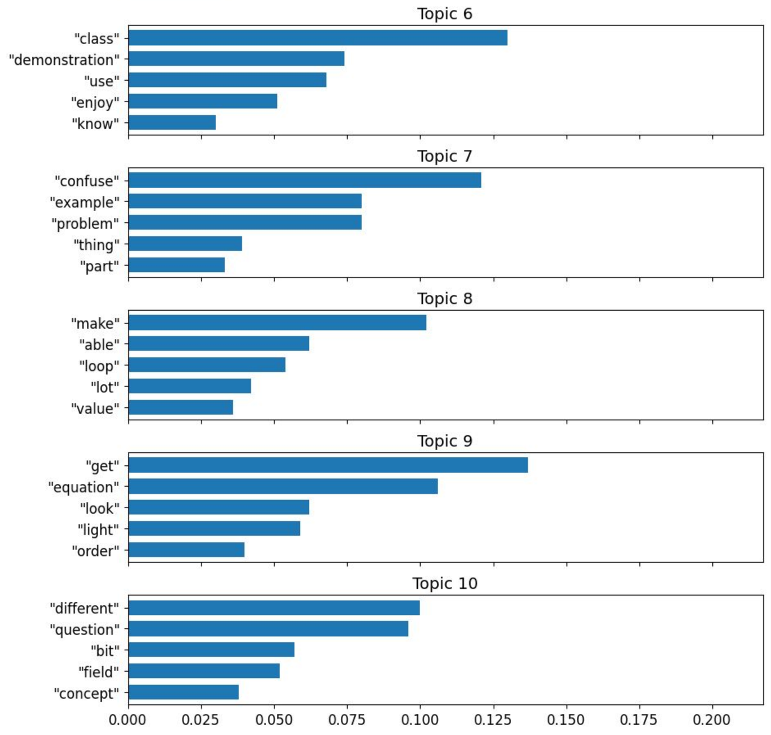
	GENDER		
COURSE	MALE	FEMALE	Not Disclosed
CS	58	45	5
ENGINEERING	148	60	5
INFOSCIENCE	42	24	10
PHYSICS	127	132	10

Topic Modeling - LDA

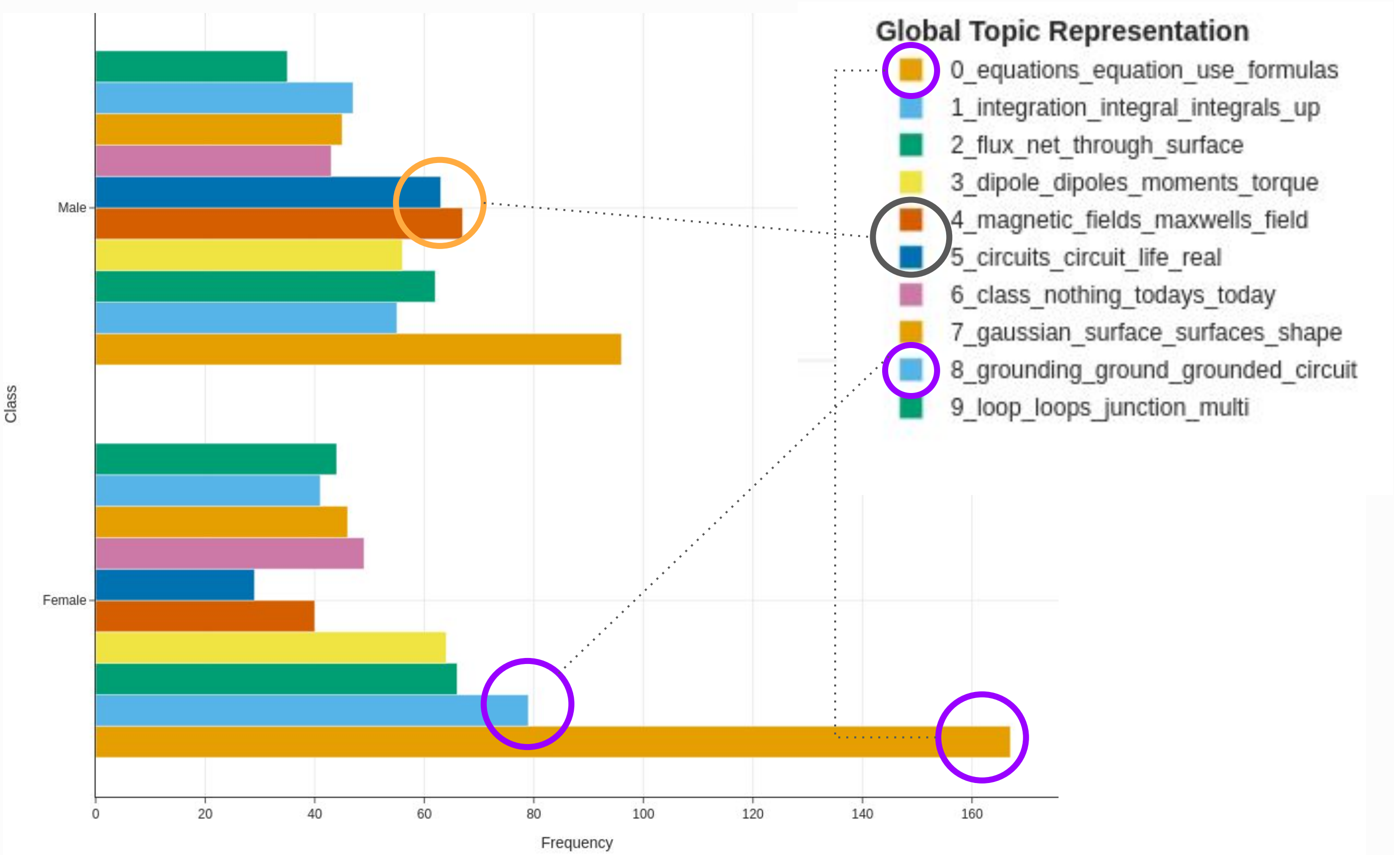


Topic Modeling - LDA Top five words per topic





Topic Modeling - BERTopic

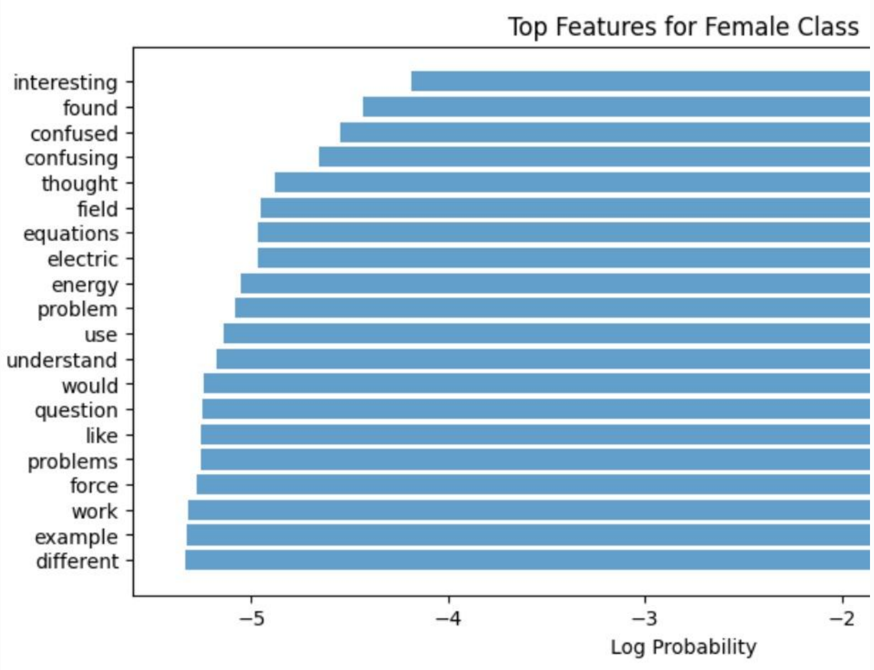


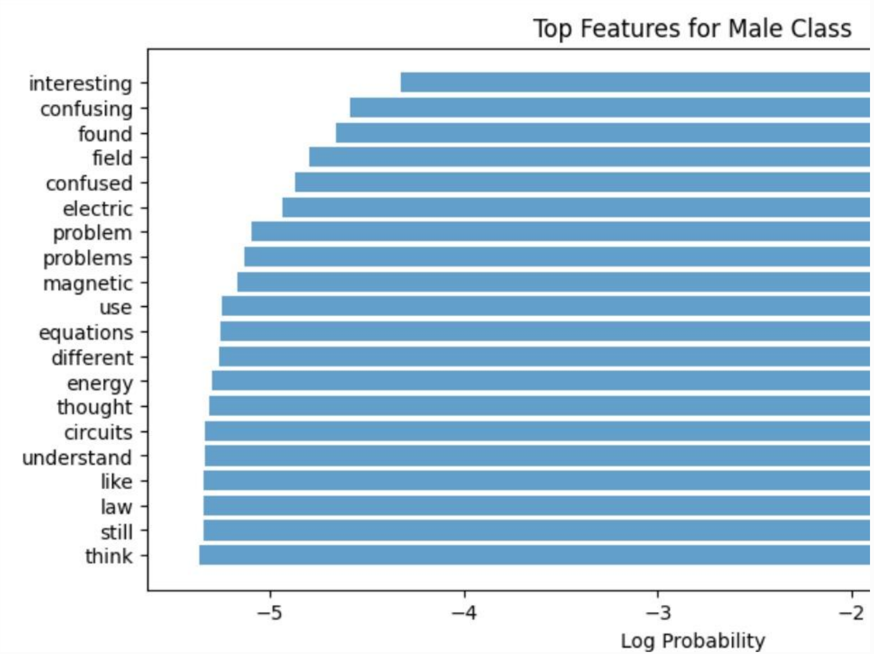
Predictive model

- Logistic Regression
- Support Vector Machines
- Naive Bayes - Unigram
- Naive Bayes - Bigram
- Naive Bayes -Tri-gram

Precision	Recall	F1
0.59	0.58	0.58
0.58	0.58	0.57
0.63	0.61	0.60
0.63	0.61	0.60
0.67	0.58	0.53

Predictive model - Top Feature Analysis



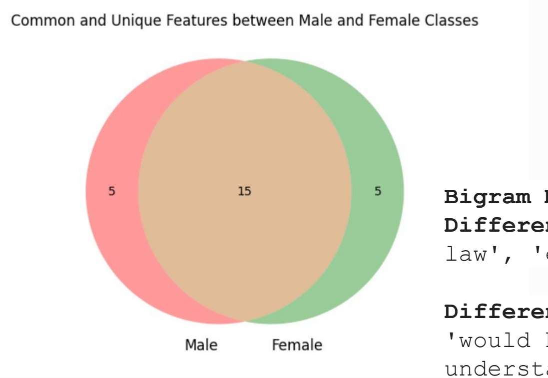


Predictive model - Top Feature Analysis

Unigram Model:
Common Features: ['found', 'confusing', 'understand', 'electric', 'equations', 'like', 'interesting', 'problems', 'confused', 'different', 'thought', 'use', 'problem', 'energy', 'field']

Different Features for Male Class: ['magnetic', 'circuits', 'think', 'law', 'still']

Different Features for Female Class: ['example', 'work', 'would', 'force', 'question']



Bigram Model:
Different Features for Male Class: ['bit confusing', 'cross product', 'little confused', 'gauss law', 'electric potential', 'found confusing', 'dont understand', 'magnetic fields', 'im still']

Different Features for Female Class: ['hat questions', 'thought interesting', 'little confusing', 'would helpful', 'bit confused', 'field lines', 'kinetic energy', 'would like', 'confusing understand']

Tri-gram Model:
Different Features for Male Class: ['problems bit confusing', 'find anything confusing', 'real world applications', 'hard time understanding', 'torque potential energy', 'top hat problem', 'im still confused', 'parallel axis theorem', 'electric field inside', 'interesting real life', 'nothing really confusing', 'concept quiz confused', 'use right hand', 'also found interesting']

Different Features for Female Class: ['multi loop circuits', 'concept quiz question', 'top hat question', 'found interesting use', 'using right hand', 'little confusing understand', 'drawing field lines', 'thought todays lecture', 'would like explanation', 'little bit confused', 'position velocity acceleration', 'found example problem', 'energy stored inductor', 'still bit confused']

Extend Analysis

1. Most interesting between genders!
2. Most confusing between genders!
3. Topics covered in Summaries!
4. Predictive model for AI vs Human generated Summaries!

Q&A

University of Pittsburgh

4. Qikun, Ming, Jiyuan

Motivation

- Lyrics are the soulful bridge that connects the rhythm to human emotions.
- Sentiment analysis could be a powerful tool that can unravel the profound layers of human emotions articulated in song lyrics.
- The existing lyric emotion classifiers have limited performance. We desire a better one.

Data

Processed_test_data

	A	B	C	D
1	Sentiment	Combined_Sentence	Filename	
2	Angry	sometim feel tire feel weak feel weak feel like want give get search	angry_1.txt	
3	Angry	see roll hat patrol tryin catch rid dirti tri catch rid dirti tri catch rid	angry_10.txt	
4	Angry	want empti head say true stain bed need free lean like rootless tre	angry_11.txt	
5	Angry	rush talib kweli feel rush yeah shit real game play might career yo	angry_12.txt	
6	Angry	scream vengeancejuda priest hey listen let get mind fill brain orde	angry_13.txt	
7	Angry	rockstar bitch call elvi mob call selfish success get jealou shorti kill	angry_14.txt	
8	Angry	god need friend god come end god lose mind god find want want	angry_15.txt	
9	Angry	know shade blue felt touch know way ghostlik sens state go seem	angry_16.txt	
10	Angry	get mad get piss grab pen write list peopl miss make shitlist one b	angry_17.txt	
11	Angry	sick sick way look thing drive crazi drive crazi sick break pay shit sn	angry_18.txt	
12	Angry	desper disappear would oh dumb fuck babi much young end bedr	angry_19.txt	
13	Angry	late side catch chang life late side catch chang life yeah late keep	angry_2.txt	
14	Angry	littl girl littl girl littl girl littl girl littl girl littl girl where go day day da	angry_20.txt	
15	Angry	feel babi yeah feel damn know glad could spend time togeth see	angry_21.txt	
16	Angry	drop hold foot let go fell seat noth break yet feel brilliant afterglov	angry_22.txt	
17	Angry	someth tell tell tell fuck mani tri chang man life alway uphil fight s	angry_23.txt	
18	Angry	impress tri hard like fli high burn wing lose faith everyth lick aroun	angry_24.txt	
19	Angry	unforeseen futur nestl somewher time unsuspect victim warn sign	angry_25.txt	
20	Angry	lose friend lose drink drive lose friend get back mend least say tri h	angry_26.txt	
21	Angry	touch feel rush clutch much enough make wonder store you lust t	angry_27.txt	
22	Angry	workin graveshift make shit wish could buy spaceship fli past sky w	angry_28.txt	
23	Angry	beauti spot borrow sweet knife rust tomorrow confess wait hear b	angry_29.txt	
24	Angry	cast eye distanc tri focu find spirit resist instead pride fall forg oppo	angry_3.txt	
25	Angry	st anger round neck st anger round neck never get respect st ange	angry_30.txt	
26	Angry	tea go cold wonder get bed morn rain cloud window see even cou	angry_31.txt	
27	Angry	god sit back limousin god come wrapper cellophan god pout cove	angry_32.txt	

Processed_train_data

	A	B	C
1	Sentiment	Combined_Sentence	Filename
2	Angry	anoth day wast time get alter state mind go overboard conscienc mee	angry_1.txt
3	Angry	decay ever faster wear immin disast open eye rumin hopeless end tim	angry_10.txt
4	Angry	final felt calm breez step watch final scene pride speak whatev speak s	angry_100.txt
5	Angry	peac confront meet war machin seiz civil liberti honest ballot among b	angry_11.txt
6	Angry	want know happi wish noth best old version pervert like would go the	angry_12.txt
7	Angry	blood rose blood rose back street blood rose blood rose back street bl	angry_13.txt
8	Angry	one day ya want wake everyth fuck everybodi suck realli know want ju	angry_14.txt
9	Angry	road complet wast work insid first rate turn town town fill nobodi care	angry_15.txt
10	Angry	shake head like wrong alreadi go even light around alon dark break he	angry_16.txt
11	Angry	black deciev lie new star bear bring light hand unti tri hard follow noo	angry_17.txt
12	Angry	see eye feel paranoid know know becom feel deep insid pain lie know	angry_18.txt
13	Angry	get failur commun men reach get last week way want well get n like n	angry_19.txt
14	Angry	crawl wreckag one time horrif memori twist mind dark rug cold hard t	angry_2.txt
15	Angry	ever hat discrimin protest demonstr picket sign wick rhyme look time	angry_20.txt
16	Angry	told felt lucki humbl breast well say sure world go chang well give dar	angry_21.txt
17	Angry	slaveshebrew bear serv pharaoh heedto everi word live fear faithof un	angry_22.txt
18	Angry	lot peopl ask stupid fuckin question lot peopl think say record talk reco	angry_23.txt
19	Angry	place know pretti ever go show make run away stay even hurt even tr	angry_24.txt
20	Angry	ay let kick right quick man nawt gangsta shit man real shit anybodi th	angry_25.txt
21	Angry	track run stupid shit good run fuck play heart say fuck play heart drea	angry_26.txt
22	Angry	address letter dear father know complet unknown guess better bothe	angry_27.txt
23	Angry	may push around win may throw rise say defi get face stop you bring	angry_28.txt
24	Angry	walk away angri brace noth gain old enough know outcom blood alw	angry_29.txt
25	Angry	want american idiot want nation new medium hear sound hysteria su	angry_3.txt
26	Angry	could see tomorrow plan one live sorrow ask friend time take stride ba	angry_30.txt
27	Angry	peek hole struggl control child love show fail see anguish eye fail see a	angry_31.txt

Methods

- TF-IDF
- Bert
- Roberta

Results

Figure 1: Results of TFIDF

Figure 2: Results of Bert

	precision	recall	f1-score	support
Angry	0.68	0.80	0.74	71
Happy	0.49	0.63	0.55	106
Relaxed	0.37	0.30	0.33	101
Sad	0.45	0.34	0.39	99
accuracy			0.50	377
macro avg	0.50	0.52	0.50	377
weighted avg	0.48	0.50	0.48	377

Accuracy: 0.50

```
250        0.432000
[6/6 02:48]
Epoch: 10.0, Training Loss: N/A, Validation Loss: 1.5400995016098022, Accuracy: 0.4482758620689655, F1: 0.43377852490742014, Precision: 0.4368142858833333, Recall: 0.4
{'eval_loss': 1.5400995016098022,
 'eval_f1': 0.43377852490742014,
 'eval_accuracy': 0.4482758620689655,
 'eval_precision': 0.4368142858833333,
```

Results

Figure 3: Results of Roberta

[75/75 1:11:47, Epoch 3/3]

Epoch	Training Loss	Validation Loss	Accuracy	F1	Precision	Recall
1	No log	1.403949	0.244032	0.156696	0.178750	0.244032
2	No log	1.384790	0.299735	0.258944	0.286270	0.299735
3	No log	1.349324	0.379310	0.368155	0.394899	0.379310

```
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344: Undefinec
  _warn_prf(average, modifier, msg_start, len(result))
```

[6/6 04:29]

```
Performance metrics:
Accuracy: 0.3793103448275862
F1 Score: 0.3681549546242349
Precision: 0.3948987916125811
Recall: 0.3793103448275862
```

5. Lingwei and Yuhang

Motivation

- **Speech acts**, which are the actions that speaker intend with utterances(actions like asking questions or making requests), playing a crucial role in understanding the intentions of a speech.
- **Emojis** may help automated systems determine speech acts.
- Build a **new dataset** and evaluate it using a simple classifier with interpretable features and see if emojis are informative.

Data

- Obtained **unprocessed** data from GitHub
- Consist of Twitter **comments containing emojis**, captured by providers using the Twitter API
- We processed about 1000 of these data and **categorized** them into five classes according to our predefined **speech act types**

Types of speech acts

Act	Explanation	Example(s)
Statement	facts, any kind of statement or specifically asserting something	twenty years 🎂 🎉 it s been two decades since we first logged on as the mozilla project and got started bringing together the
Question	for any question asked or for any kind of request	who is your favourite 🔥; could i possibly please get a birthday tweet for the 😀
Suggestion	giving any kind of suggestion and recommendation	get some video of this if you can and post it on the twitter 👍
Comment	for any kind of expression of feeling or thought	oh very happy day😀
Miscellaneous	for any commitment to future action or for any declarative	i m in the far east for the next few weeks but when i m back i ll sort it 😀; You are fired!!!😡😡😡😡

Annotation

- We create a “coding manual“ to specify the definitions for types.
  - Typical sentence orders include **declarative**, **interrogative**, and **imperative** order. Key pattern to confirm the types: “I will….”

```
1. checking the starting word of the sentence:
    if find auxiliary/model verb or "wh-": # can, will, when, why
        # check the total sentence order
        if Auxiliary/Modal Verb + Subject + Main Verb
            # most likely this sentence is of type "question"
            # check the tone, may be rhetorical question.
            if rhetorical question:
                # for example: Emotional Appeal or No Expected Answer
                label as "comment"
            else:
                label as "question"
```

How reliable the scores are

- Two members to create the dataset
  - May result in inconsistency issue.
- Cohen's kappa's "inter-annotator agreement":
  - Po: relative observed agreement
  - Pe: probability of chance agreement for each category

$$\kappa = \frac{P_o - P_e}{1 - P_e}$$

```
line 41 Wang's label is 0 and Li's label is 3
line 50 Wang's label is 2 and Li's label is 4
line 59 Wang's label is 0 and Li's label is 2
line 84 Wang's label is 3 and Li's label is 0
line 87 Wang's label is 3 and Li's label is 2
```

Method and result

- Classifiers of logistic regression with tf-idf features
- 5-fold cross validation
-

	Accuracy	Recall	F1 score
Text with emoji	0.72	0.72	0.602
Emoji translates to text	0.705	0.705	0.583
Text without emoji	0.69	0.69	0.567

Baseline: 71.5%. 715 sentences with type “comment”

6. Gina, Modhumonty, Norah

Gender Bias by Region: Motivation

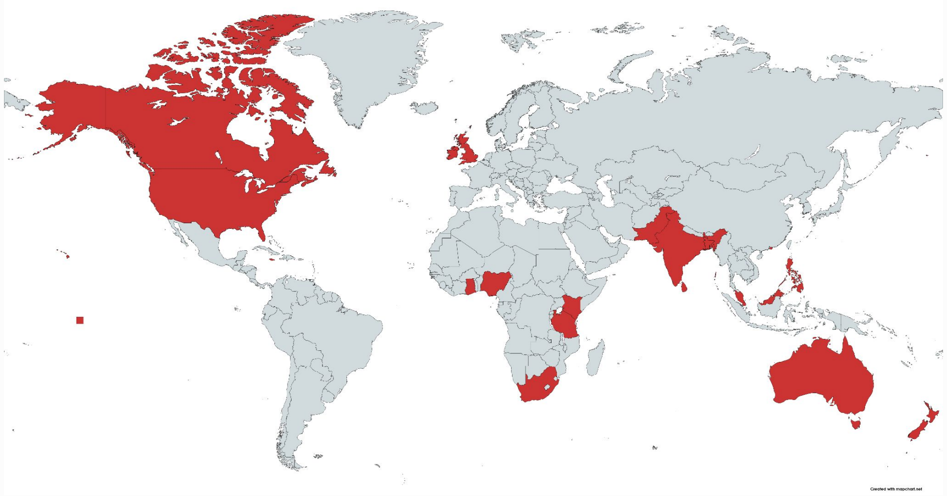
Inspired by previous work analyzing gender bias in Canadian news, we wanted to **compare gender bias** across multiple regions by news topic.

- Topics are generated by an LDA topic model
- Gender bias is analyzed per topic per region
    - Gender binary (M/F) was analyzed

Photo by Fabien Barral on Unsplash

Gender Bias by Region: Data

- News On the Web (NOW) Corpus[1]
  - English texts from 20 countries spanning the globe
  - Used subset of **Pakistan** and **Malaysia** news sources
  - Used **6 months** of news data
    - 28,462 MY articles
    - 30,952 PK articles
  - Data took approximately 5 minutes to preprocess and 5 minutes to train the models



Countries included in NOW News Corpus: AU, BD, CA, GB, GH, HK, IE, IN, JM, KE, LK, **MY**, NG, NZ, PH, **PK**, SG, TZ, US, ZA

https://www.english-corpora.org/now/

Gender Bias by Region: Preprocessing and Models

- Extensive **preprocessing** applied to documents before use
  - Tokenization, Normalization, Lowercasing, Stop Word Removal
  - Lemmatization by NLTK
  - Relative Pruning
    - Words present in 80% or more documents removed: 0
    - Words present in less than 5% of documents removed: 168,828
- Compared LDA (both Gensim and Scikit-Learn), BerTOPIC, and Top2Vec
  - BerTOPIC performed the best when unlimited, but poorly when we limited the number of topics to 10
  - Ultimately chose to use the **scikit-learn LDA model**

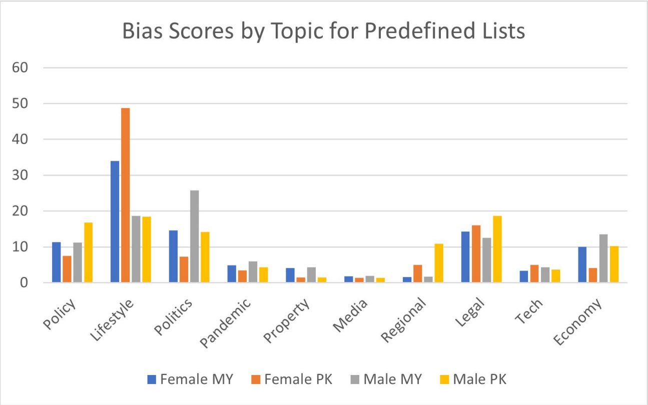
Gender Bias by Region: Gender Bias Analysis

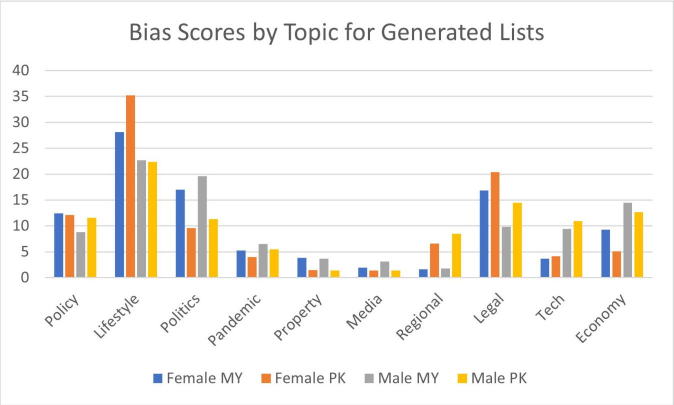
- For each text, analyzed which gender appears the most out of all gendered words
  - First method tried was a **predefined** word list of M/F pronouns
  - Second method was to **generate** a word list using top 10 words by GloVe embedding similarity to a seed word (e.g. “man” and “woman”)
    - List pruned of opposite-gendered words and “person”

Predefined Female	she, her, hers
Predefined Male	he, him, his
Generated Female	woman, girl, mother, child, herself, victim, wife, she, teenager, couple
Generated Male	man, boy, one, turned, another, whose, once, life, thought, victim

Gender Bias by Region: Results

- Findings are consistent with expected results:
  - **Female-centric** words are primarily found in **lifestyle** topics
  - **Male-centric** words are somewhat higher in **politics** and **economy** topics
  - Generated lists enhance some biases (tech) and reduce others (lifestyle)
  - Some topics, like pandemic, property, and media, contain minimal gendered words





7. Ahana, Dhanush, Yixiao

Introduction

What we did

We have redirected our research focus. Rather than enhancing model accuracy in predicting hate speech (HS) targets, our main goal is now to develop models specifically trained on explicit HS and assess their effectiveness in recognizing implicit HS targets.

Why we did it

Limited research has been conducted on comparing model performance in identifying targets of explicit versus implicit HS. The focus of current research is primarily on detecting targets of explicit HS. Implicit HS, often masked in nuanced expressions might pose challenges for automated detection systems.

Research Goal

In this direction, we study how well models trained on explicit HS datasets perform on implicit HS datasets for target group identification.

Assumption

For our study, we are operating under two key assumptions regarding the HS dataset and the HS target identification models we have fine-tuned.

- Firstly, if a HS dataset isn't explicitly labeled as containing explicit or implicit hate speech, we will classify it as an explicit HS dataset.

- Secondly, we will consider that the models we've fine-tuned are trained solely on explicit HS.

These assumptions are considered reasonable for our research's depth, given the scarce availability of implicit HS datasets and the limited research on explicitly integrating implicit HS in HS modeling tasks.

Training Data

We fine-tuned the models using a comprehensively target-labeled HS dataset (Yoder et al., CoNLL 2022), which includes 10 distinct target categories.

women	8431
black people	5288
muslims and arabic/middle eastern people	4140
lgbtq+ people	3215
asian people	2401
latinx people	2097
jews	1530
white people	680
men	553
christians	545

As evident, the categories exhibit a high level of imbalance, a common phenomenon in target-labeled HS datasets. Consequently, we anticipate that the model's performance on categories with smaller sample sizes will be comparatively weaker.

Test Data

We tested our models on both explicit and implicit HS datasets to gain insights. The test data for the explicit HS datasets to gain insights. The test data for the explicit HS was sourced from the same dataset as the training set.

To compile an implicit HS dataset with the same target labels as our training dataset, we found a comprehensively target-labeled implicit HS dataset(ElSherief et al., EMNLP 2021), and then recategorized and relabeled it to align the categories with those of our training data.

women	5515
black people	3553
muslims and arabic/middle eastern people	2718
lgbtq+ people	2032
asian people	1601
latinx people	1422
jews	1045
white people	479
men	367
christians	356

white people	993
muslims and arabic/middle eastern people	621
black people	514
jews	503
men	70
lgbtq+ people	64
latinx people	56
women	53
asian people	45
christians	29

Note that the amounts of test data we have for implicit and explicit HS are not at all proportional; they differ significantly in magnitude and exhibit high imbalance across categories. This disparity could affect our interpretation and analysis of the model's performance on explicit and implicit HS.

Results

Dataset	#Target	Model	Acc	F1
Gab (Kennedy et al., 2022)	10	RoBERTa	0.71	0.55
MLMA Hate Speech	6	Bidirectional LSTM	0.93	0.94
Identity hate speech corpora	10	RoBERTa	0.6	0.68
Identity hate speech corpora	10	distilbert	0.78	0.72
Identity hate speech corpora	10	HateBert	0.24	

RoBERTa based

Test Result on Explicit HS

	precision	recall	f1-score	support
asian people	0.65	0.63	0.64	1601
black people	0.71	0.67	0.69	3553
christians	0.61	0.48	0.54	356
jews	0.62	0.60	0.61	1045
latinx people	0.69	0.48	0.57	1422
lgbtq+ people	0.62	0.75	0.68	2032
men	0.00	0.00	0.00	367
muslims and arabic/middle eastern people	0.69	0.77	0.73	2718
white people	0.38	0.39	0.39	479
women	0.69	0.75	0.72	5515
accuracy			0.67	19088
macro avg	0.57	0.55	0.56	19088
weighted avg	0.66	0.67	0.66	19088

Test Result on Implicit HS

	precision	recall	f1-s
asian people	0.42	0.60	
black people	0.75	0.79	
christians	0.34	0.72	
jews	0.88	0.80	
latinx people	0.39	0.79	
lgbtq+ people	0.70	0.67	
men	0.00	0.00	
muslims and arabic/middle eastern people	0.86	0.86	
white people	0.86	0.75	
women	0.26	0.79	
accuracy			
macro avg	0.55	0.68	
weighted avg	0.79	0.77	

Analysis

Surprisingly, the RoBERTa-based model demonstrates better performance in identifying targets in implicit HS on average.

However, it's noteworthy that in categories of implicit HS where the model's performance was relatively poor (with an f1-score around or below 0.5), the model actually performs better when dealing with explicit HS.

These categories are:

- Asian people
- Christians
- Latinx
- Men
- Women

These five categories are the ones with the least amount of test data in the implicit HS dataset.

Conversely, in categories where the model showed good performance in implicit HS (f1-score above 0.6) - its performance was weaker in explicit HS.

These categories are:

- Black people
- Jews
- LGBTQ+ people
- Muslims and Arabic/Middle Eastern people White people

These categories are the top five in terms of the amount of test data in the implicit HS dataset.

In theory, increasing the amount of test data should make the test results more representative. We observed that as the test data volume increased, accuracy also increased. This suggests that for this specific model, the implicit nature of HS does not negatively impact target identification accuracy.

To explain this somewhat counterintuitive phenomenon, consider the following reasons:

- Firstly, the amount of test data for implicit HS is still significantly smaller compared to that for explicit HS. Therefore, the results might not be sufficiently representative. If we continue to increase the volume of implicit HS test data, the accuracy might decrease.

- Secondly, LLMs have the capability to understand language nuances. Given that this is a classification task with a relatively small number of categories, the model's ability to discern these nuances might be adequate in this context.

8. Birju and Robbie

Content Warning: homophobia

Project Motivation

Americans increasingly get their political news from podcasts.

Many listeners create fan communities around these podcasts on Reddit.

We wanted to measure how much influence political podcasters have over those that listen to them.

Data

- Podcast transcripts and community comments from 23 political podcasts
- Podcast Dataset (“training” data)
  - Main Source: YouTube Transcripts API
  - Alternate Source: podcastindex.org API
- Comment Dataset (“test” data)
  - Cornell corpus of Reddit data
- Data organized and stored in a relational SQLite database
  - Transcripts are in files which database links to
  - Comments are stored directly in db tables

Methods - Word Significance using Term-Context Matrix

Question:

Does the language that podcasters use to describe minority groups on their show relate to the way those groups are talked about in their fan communities?

Method:

Construct term-context matrices for each podcast and subreddit. Find word associations for a list of identity words.

Sample from Results - Term-Context Matrix

Target Word: ‘gay’

The Ben Shapiro Show:

Episode Associations: ['black', 'religious', 'poor', 'homeless', 'bad', 'rich', 'young', 'dead', 'crime', 'great']

Subreddit Associations: ['hypocrite', 'straw', 'joke', 'doctor', 'good', 'little', 'memoir', 'lasting', 'thief', 'f****t']

Sample from Results - Term-Context Matrix

Target Word: ‘gay’

Pod Save America:

Episode Associations: ['journey', 'rich', 'cheerleader', 'spokesperson', 'young', 'few', 'felony', 'precedent', 'good', 'sympathy']

Subreddit Associations: ['civil', 'voting', 'having', 'reproductive', 'joke', 'woman', 'great', 'abuses', 'good', 'human']

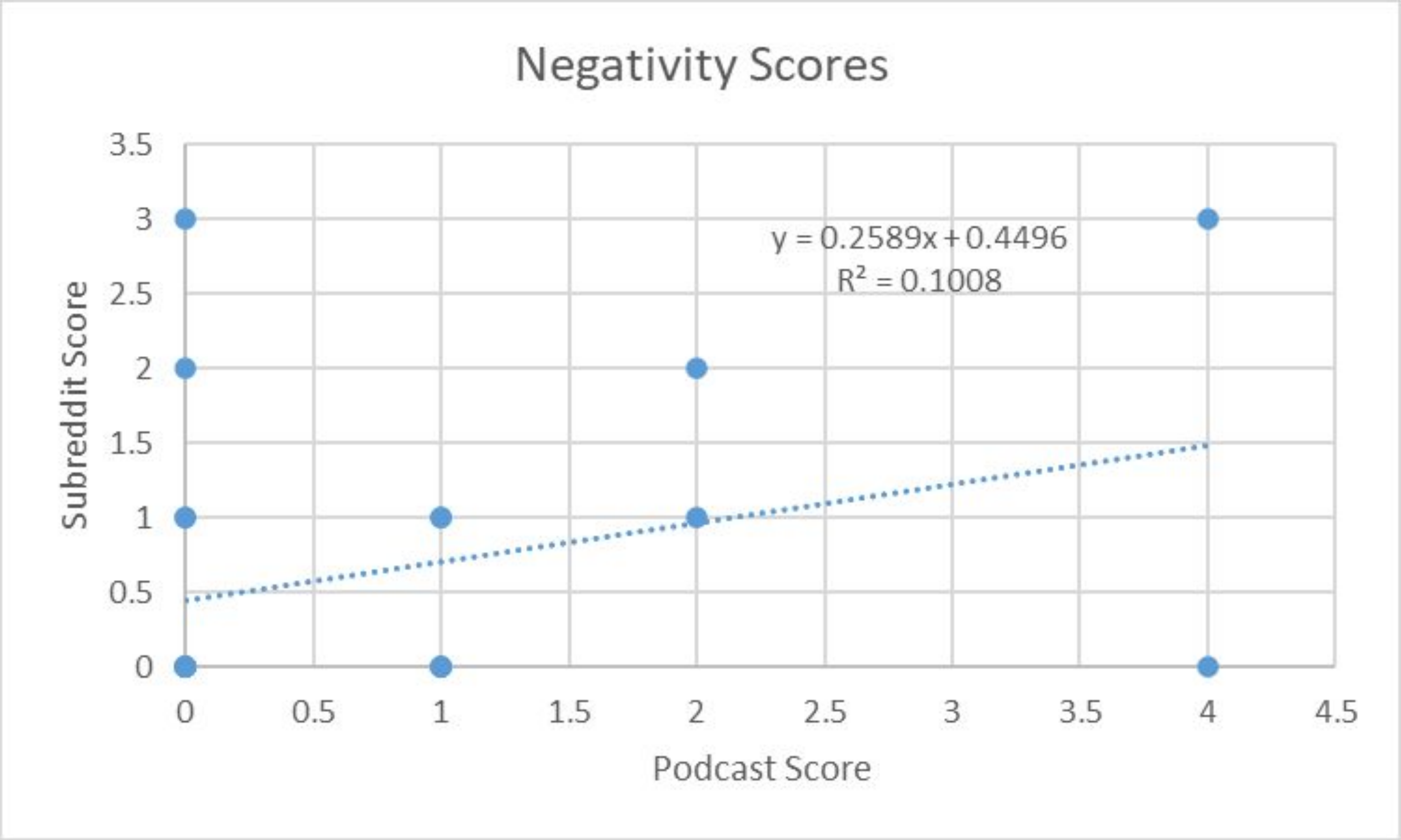
Results - Term-Context Matrix

We calculated a negativity score for each podcast and each subreddit.

Negativity Score = # of negative adjectives in list of word associations

We did see some relationship between the language use for some podcasts. However, across all podcasts, we saw no correlation between the language use.

Target Word: Gay



Methods - N-Gram Language Models

- 69 language models are built with training data from podcast transcripts
  - Adopted code from Homework 3 to build models
  - Unigram, Bigram and Trigram models for each of the 23 podcasts
- Tested on reddit comments from the community associated with that podcast
  - Also 5 randomly chosen alternative reddit communities
- We found that these n-gram models trained on podcast hosts could **not** be used to predict community language use

Sample from Results - Unigram Language Models

Training Podcast	Testing Community	Perplexity
The Ben Shapiro Show	The Ben Shapiro Show	47.569
The Ben Shapiro Show	The Alex Jones Show	25.907
The Ben Shapiro Show	Human Rights Watch	23.716
The Ben Shapiro Show	The Jimmy Dore Show	24.507
The Ben Shapiro Show	Political Gabfest	25.418
The Ben Shapiro Show	The Joe Rogan Experience	24.59

Sample from Results - Unigram Language Models

Training Podcast	Testing Community	Perplexity
Pod Save America	Pod Save America	18.975
Pod Save America	Human Rights Watch	23.979
Pod Save America	The Daily	24.836
Pod Save America	H3 Podcast	23.89
Pod Save America	Political Gabfest	25.603
Pod Save America	Contrapoints	24.353

Sample from Results - Trigram Language Models

Training Podcast	Testing Community	Perplexity
The Ben Shapiro Show	The Ben Shapiro Show	16.573
The Ben Shapiro Show	hbomberguy	24.012
The Ben Shapiro Show	Louder with Crowder	24.095
The Ben Shapiro Show	Political Gabfest	25.552
The Ben Shapiro Show	Pod Save America	21.364
The Ben Shapiro Show	Jordan B Peterson	23.813

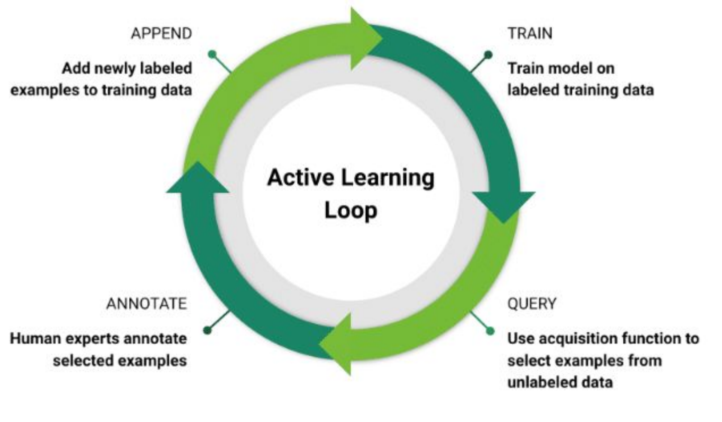
Reflection & Concluding Thoughts

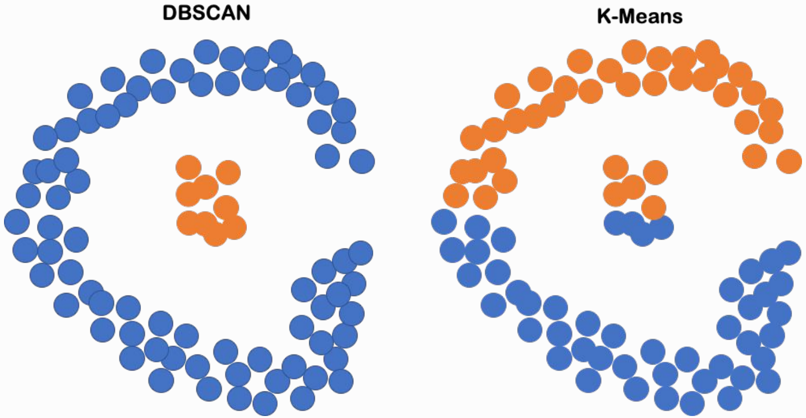
- Creating this new large dataset produced some interesting findings in terms of word significance

- However, not much can be concluded from the language models we built due to inconsistent results

9. Qichang, Yuxuan, Haoyu

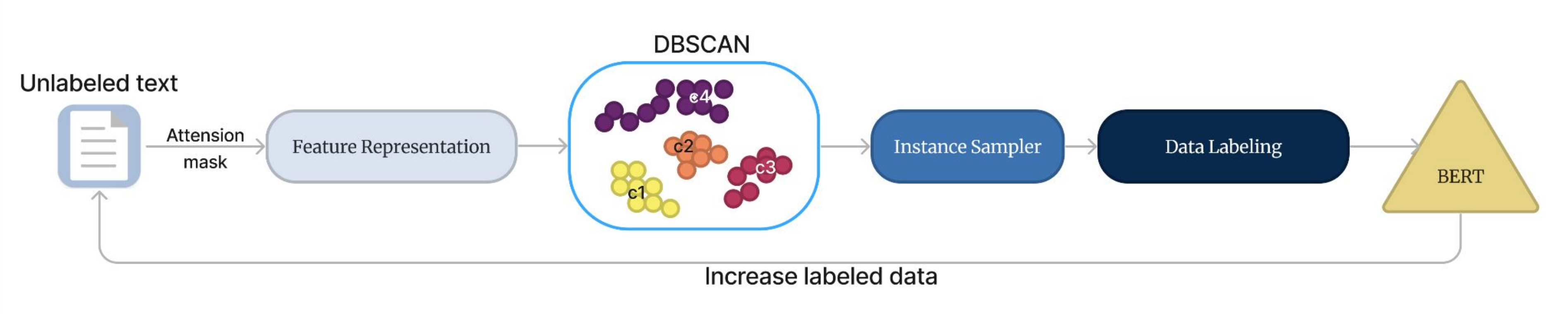
Motivation



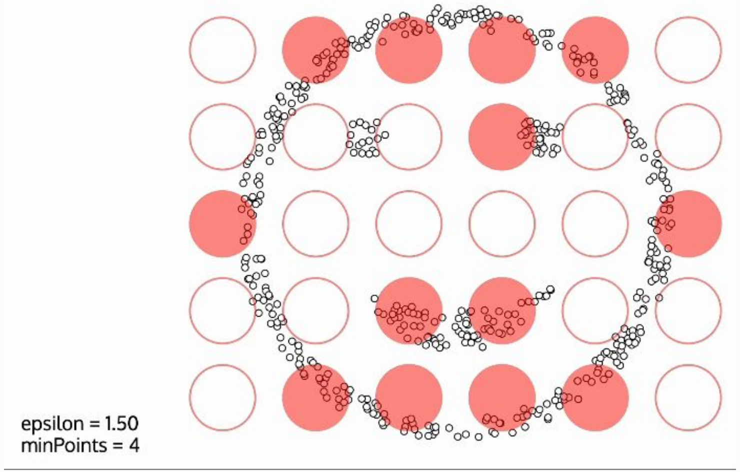


- Current sentence classification methods can achieve impressive performance such as Transformer (Vaswani et al., 2017) and BERT (Devlin et al., 2018), fine-tuning these models require great amount of data.
- K-means is vulnerable to outliers.
- Set specific cluster number requires rich experience.

Method



- Adapting DBSCAN to Cluster-based Active Learning
- Iteratively add unlabeled data and re-train BERT



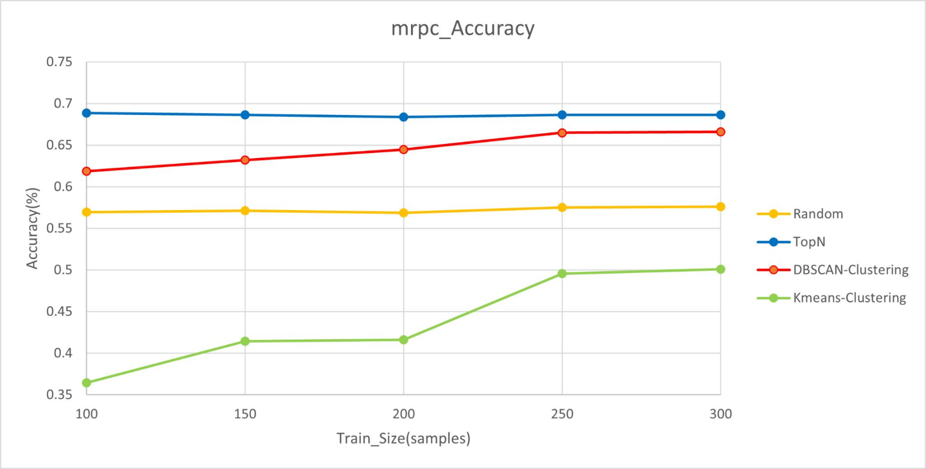
Experiment Setup

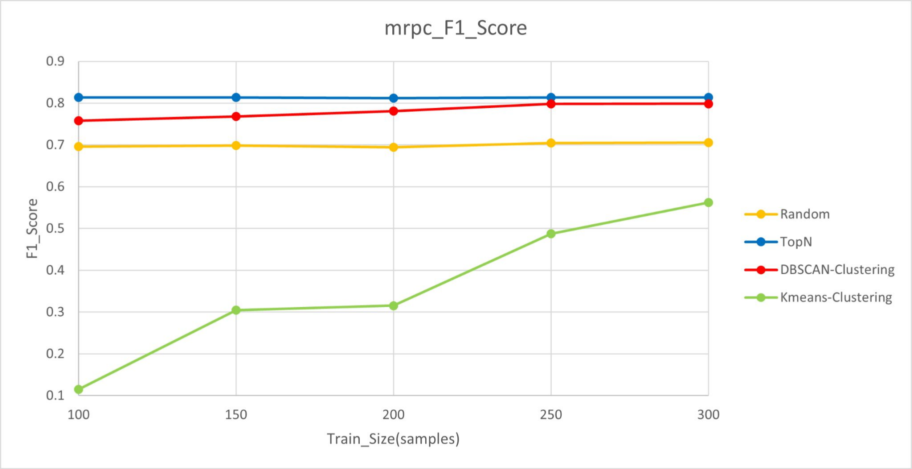
- Baselines:
  - No AL: using pure dataset to finetune BERT
  - Random: Randomly sampling from unlabeled data
  - TopN: Select the most N informative unlabeled data
  - Kmeans-Clustering: adaptive k-means clustering
- Using mean value to represent the result of Random AL
- Number of clusters: 10(Baselines)
- Initial sample: 50
- Batch to label: 50
- Active learning round: 5
- Model: bert-base-uncased

Dataset

- MRPC:
  - Microsoft Research Paraphrase Corpus (MRPC) is a corpus consists of 5,801 sentence pairs collected from newswire articles. Each pair is labelled if it is a paraphrase or not by human annotators.
- QQP:
  - Quora Question Pairs (QQP) dataset consists of over 400,000 question pairs, and each question pair is annotated with a binary value indicating whether the two questions are paraphrase of each other.
  - Created a subset of 3,691 unlabeled data, 1,391 test data.

Results





- DBSCAN-Clustering surpass Kmeans-Clustering around **20%**
- A clear rising trend for cluster-based Active Learning

Discussion

Methods	Train Size(MRPC)	Accuracy(MRPC)	F1(MRPC)
No AL	3688	0.6504	0.7437
Random	300	0.5759	0.7058
TopN	300	0.6863	0.8134
K-means Clustering	300	0.5009	0.5623
DBSCAN Clustering	300	0.6661	0.7986

- Adapted DBSCAN to cluster-based active learning.
- Active learning can efficiently reduce the amount of training data.
- Limitation
  - Experiments on more data
- Future Work: Handling Varying Densities

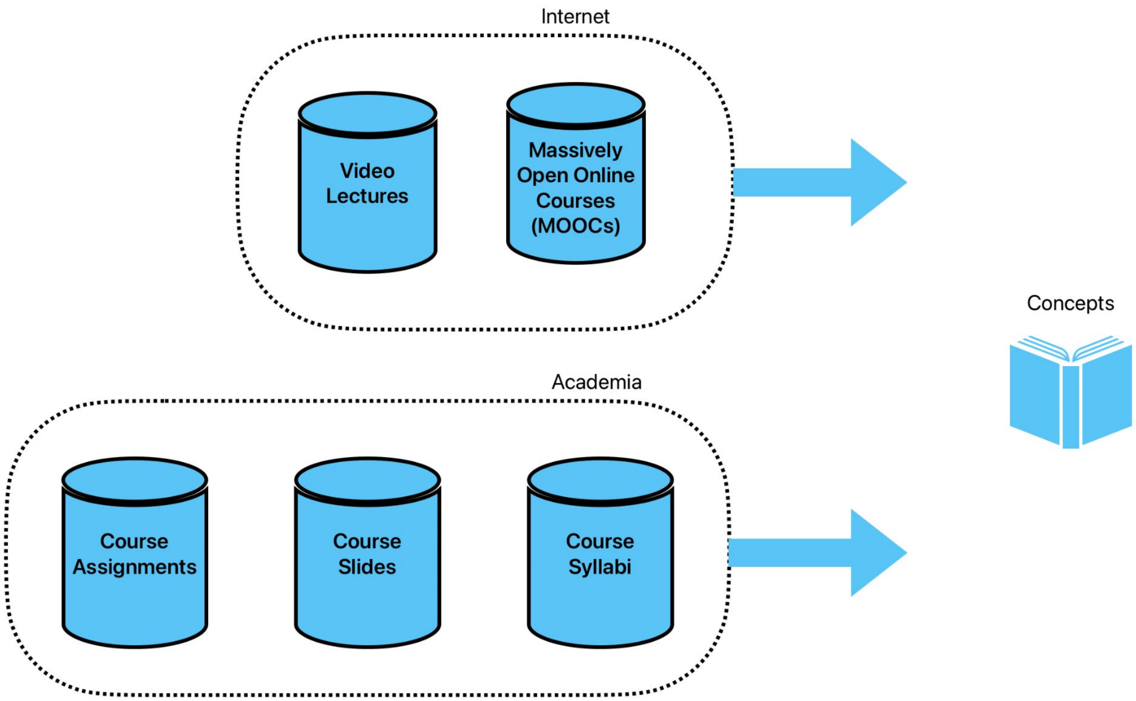
10. RJ🤓 and Jacob😎

CONCEPT EXTRACTION FROM COURSE MATERIAL

JACOB HOFFMAN AND RAJA KRISHNASWAMY

MOTIVATION

- Course material data collections available for automatically extracting concepts



MOTIVATION

- Concept extraction upon course material may:
    - Expedite the learning process for students
    - Help students better understand the main points of the material
    - Liberate instructors from the tedious process of human labeling

DATASET - OVERVIEW

1. Create a manually BIO labeled, small-sized dataset using a subset of existing course material (slides and syllabi). For this we chose the slides from the CS courses Raja took (CS-0441, CS-0449, CS-1541, CS-1550, CS-1567, CS-1622).
2. Create a dictionary of concepts based on a reputable source (Wikipedia…)
3. Split into a training set and a test set (80/20 of manual)
4. Extend the training dataset into a full-sized dataset by labeling more documents using a matching algorithm with the dictionary of concepts

DATASET - EXAMPLE BIO-LABELED ENTRY

text	the	operating	system	uses	interrupts	to	implement	system	calls
label	O	B	I	O	B	O	O	B	I

- B = Beginning, I = Inside, O = Outside
- Concepts =
  - Operating System
  - Interrupts
  - System Calls

DATASET LABELING - HOW

- A word/series of words was defined as a concept based on whether it was a defined term and whether it was used repeatedly (more than 2 times) later on, or focused on extensively.

- 10,659 lines out of the total 46,195 lines were manually labelled, or 23.1% of the total dataset.

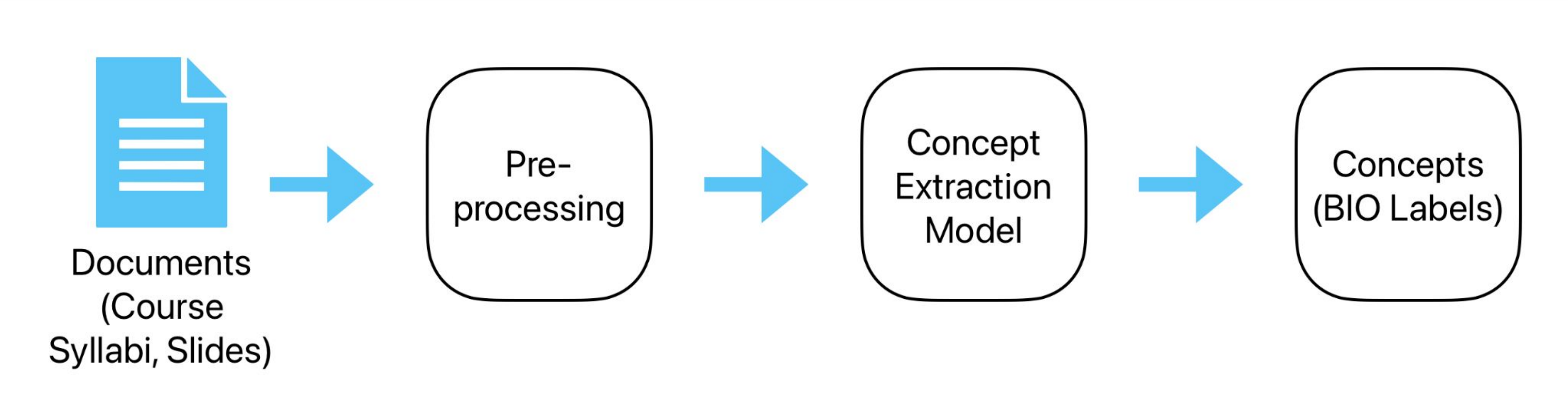
DATASET LABELING – WHAT

TO DO	IN PROGRESS	DONE
CS-1622	CS-0449 (9% done; 1,040 lines)	CS-1567 (2,241 lines)
CS-1550	CS-1541 (2% done; 208 lines)	CS-0441 (7,170 lines)

DATASET LABELING - DICTIONARY

- A dictionary with 1,137 concepts was compiled from multiple Wikipedia articles related to the “Computer Science” field.
  - *Compiler Construction/Glossary*
  - *Index of computing articles*
  - *List of computer term etymologies*
  - *Outline of computer science*
  - *Glossary of computer science*

APPROACH - OVERVIEW



APPROACH - PRE-PROCESSING

- Converted Slide Deck PDFs to data structure that could be labeled
  - Extracted the text from the pdfs using pdf2text python library, then used nltk.word_tokenize to tokenize into words to label (stored in a JSON file).
- Implemented dictionary empowerment:
  - Iterated through the words, and if concepts match with the successors of the word, choose the one with the most tokens, label the concept, and then iterate again starting with the word after.

APPROACH - MODEL

- Fine-tuned a BERT NER model (bert-base-uncased) on the dataset.
  - Utilized the BertForTokenClassification (with PyTorch) included in HuggingFace's *Transformers* library.
  - *NielsRogge - Transformers Tutorials - Custom Named Entity Recognition with BERT*

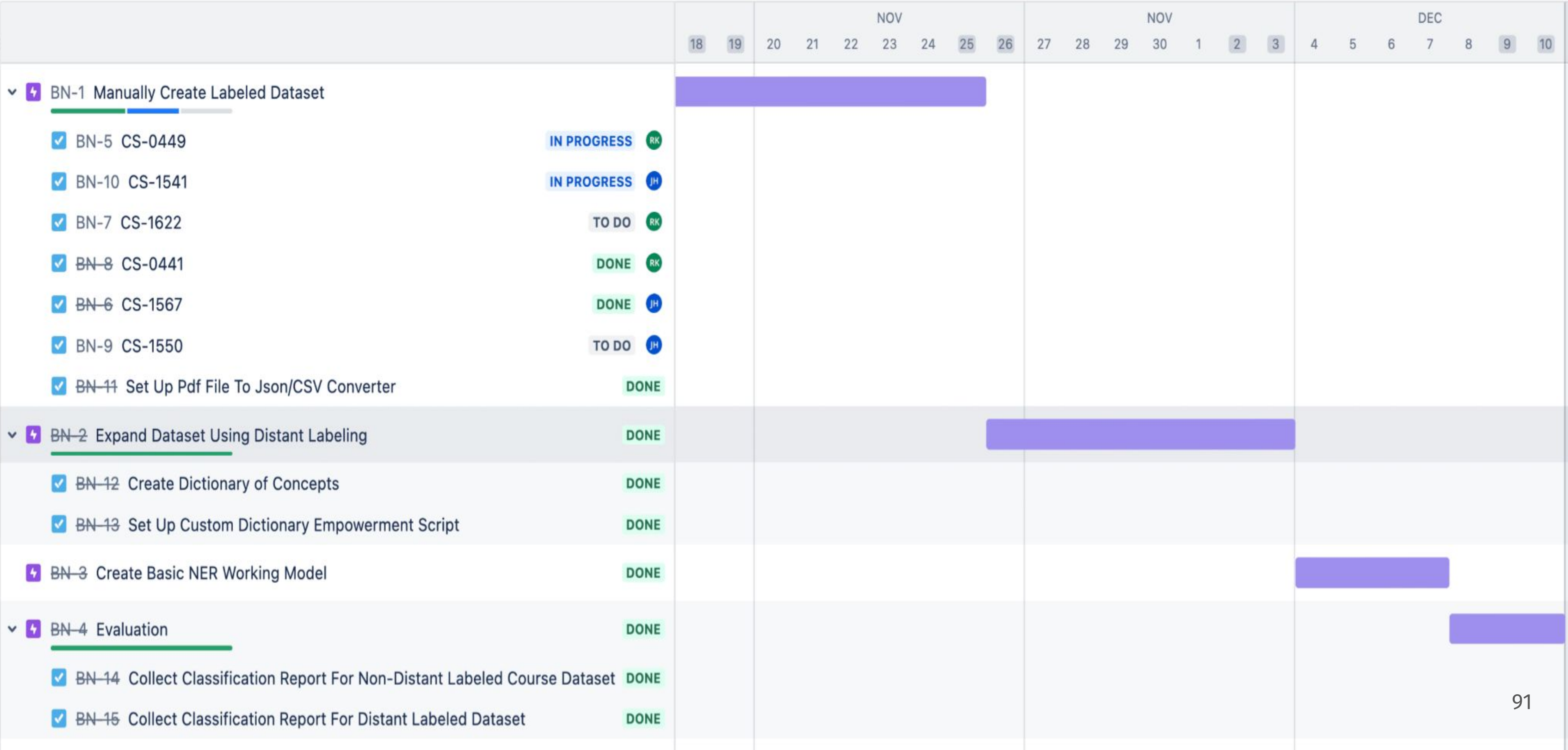
EVALUATION - DISTANT LABELING WORKS!

DISTANT LABELING?	ACCURACY	PRECISION	RECALL	F1-SCORE	SUPPORT
NO	0.92	0.48	0.42	0.45	1151
YES	0.93	0.50	0.62	0.55	1162

SAMPLE OUTPUT

```
the client node in a wide area network , which connects to a route ##r node .
['O', 'B', 'B', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'B', 'O']
```

TIMELINE – JIRA



QUESTIONS?

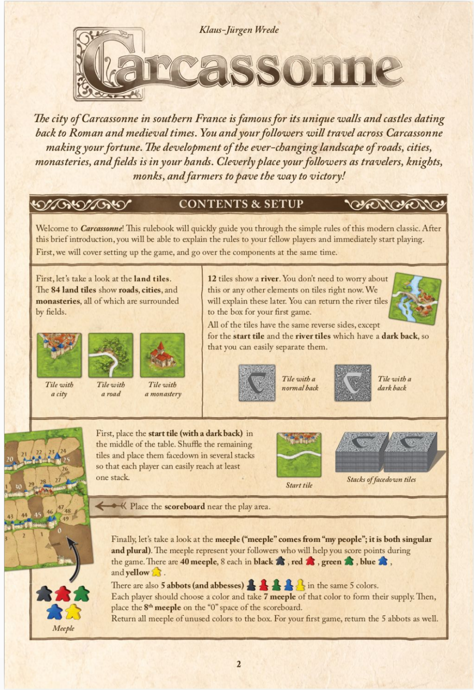
11. Marcelo and Connor

Motivation

Game Manual

Pre-Trained Language Model
+ RL Agent

Environment



Klaus-Jürgen Wrede

Carcassonne

The city of Carcassonne in southern France is famous for its unique walls and castles dating back to Roman and medieval times. You and your followers will travel across Carcassonne making your fortune. The development of the ever-changing landscape of roads, cities, monasteries, and fields is in your hands. Cleverly place your followers as travelers, knights, monks, and farmers to pave the way to victory!

CONTENTS & SETUP

Welcome to *Carcassonne*! This rulebook will quickly guide you through the simple rules of this modern classic. After this brief introduction, you will be able to explain the rules to your fellow players and immediately start playing.

First, we will cover setting up the game, and go over the components at the same time.

First, let's take a look at the **land tiles**. The **84 land tiles** show **roads**, **cities**, and **monasteries**, all of which are surrounded by fields.

Tile with a city *Tile with a road* *Tile with a monastery*

12 tiles show a **river**. You don't need to worry about this or any other elements on tiles right now. We will explain these later. You can return the river tiles to the box for your first game.

All of the tiles have the same reverse sides, except for the **start tile** and the **river tiles** which have a **dark back**, so that you can easily separate them.

Tile with a normal back *Tile with a dark back*

First, place the **start tile (with a dark back)** in the middle of the table. Shuffle the remaining tiles and place them facedown in several stacks so that each player can easily reach at least one stack.

Start tile *Stacks of facedown tiles*

Place the **scoreboard** near the play area.

Finally, let's take a look at the **meeple ("meeple" comes from "my people"; it is both singular and plural)**. The meeple represent your followers who will help you score points during the game. There are **40 meeple**, 8 each in **black**, **red**, **green**, **blue**, and **yellow**.

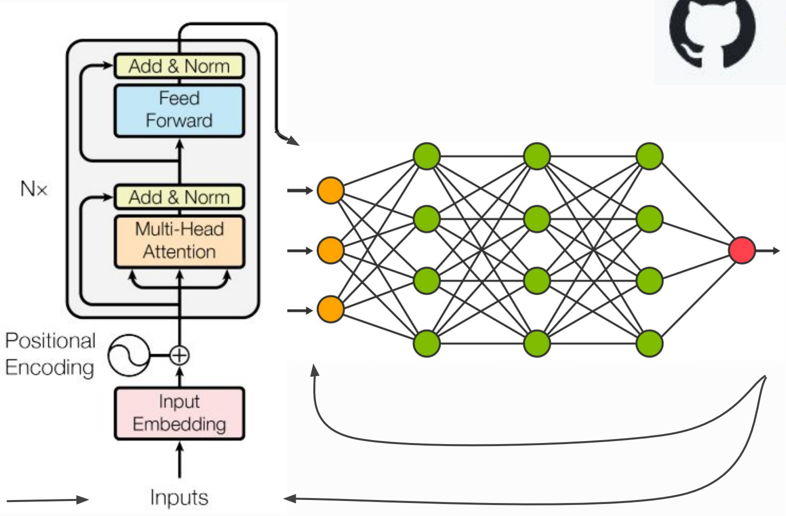
There are also **5 abbots (and abbesses)** in the same 5 colors.

Each player should choose a color and take **7 meeple** of that color to form their supply. Then, place the **8th meeple** on the "0" space of the scoreboard.

Return all meeple of unused colors to the box. For your first game, return the 5 abbots as well.

Meeple

2



wingedsheep / **carcassonne**

⬅️ ["Carcassonne Rulebook v3", zmangames.com/en/products/carcassonne]
⬆️ [Transformer block from "Attention is All you Need", Vaswani et al. 2017.
Feedforward network adapted from vitalflux.com/sklearn-neural-network-regression-example-mlpregressor]
↖️ [github.com/wingedsheep/carcassonne]

Data

- Carcassonne Game Manual
- BERT
  - Fine tuned on the manual
- Reinforcement Learning Rewards

Methods

- Prompt the language model for a move
  - State (in text form)
- If it gives a valid move, use it
  - "Exploratory move"
- Otherwise, choose the move the RL agent sees as the best move
  - "Exploitation move"
- After the game, each move will get a reward depending on how the game went

Results (or lack thereof)

- No significant data points yet
- Project components that are done
  - Language model fine-tuned on the manual
    - Not performing well, probably needs more data to train on
  - RL agent playing the game
  - RL agent learning from the game
- Project components that need developed
  - Integration between LM and RL agent
  - Testing on both LM-informed and LM-uninformed models

Questions?