# NLP TECHNIQUES IN ENGINEERING EDUCATION

Neha Kardam
Under supervision of Prof. Mari Ostendorf
EE517: Introduction to Statistical Learning
Spring 2021

# TASK DESCRIPTION

Problem #1: Multi label classification of student responses to the following question:

*"What one action can students in your lab groups take to improve your educational experience at UW?"*

Problem #2: Identify the topics, Sentiments and semantic similarity of student responses to the following question:

*"What one action can faculty take to improve your educational experience at UW?”*

# DATA DESCRIPTION

## P #1: Multi label classification (712 rows × 3 columns)

| Students Response to Question 1 | C.1 | C.2 |
|---|---|---|
| Participate and engage as much as possible by collaborating during the agreed meeting time and being prepared for the lab. | PI | IA |
| They can be better at communicating and more time efficient | ISS | POSI |

## P #2: Topics, Sentiment and Semantic similarity (1624rows × 60 columns)

| Class | Quarter | Year | B1_Age | B2_Gender | B3.1_USStatus | B4.1_Race | Students Response to Question 2 |
|---|---|---|---|---|---|---|---|
| EE233_Spring2020 | Spring | 2020 | 20 | 2 | 1 | 6 | Restructure quizzes and stuff. In 235 we had a weekly quiz in lieu of midterms and a final, and that helped keep people engaged and paying attention. |

# OVERVIEW OF APPROACH

| P #1: Multi Label Classification |
|---|
| A. Single Output Layer Model (Baseline)<br>B. Multiple Output Layers Model |

| P #2A: Topic Modeling | P #2B: Sentiment Analysis | P #2C: Semantic Similarity |
|---|---|---|
| 1. Latent Dirichlet Allocation (LDA)<br>2. Non-Negative Matrix Factorization (NMF) | Pertained Sentiment analysis tools are used<br>1. NLTK<br>2. Textblob<br>3. Flair | 1. Sentence embedding using infersent<br>2. K-means for clustering<br>3. Agglomerative clustering |

# RESULTS

## P #1: Multi Label Classification

| Single Output Layer Model | Multiple Output Layers Model |
|---|---|
| Test Accuracy: 0.475 | Test Accuracy: 0.693 |

## P #2A: Topic Modeling

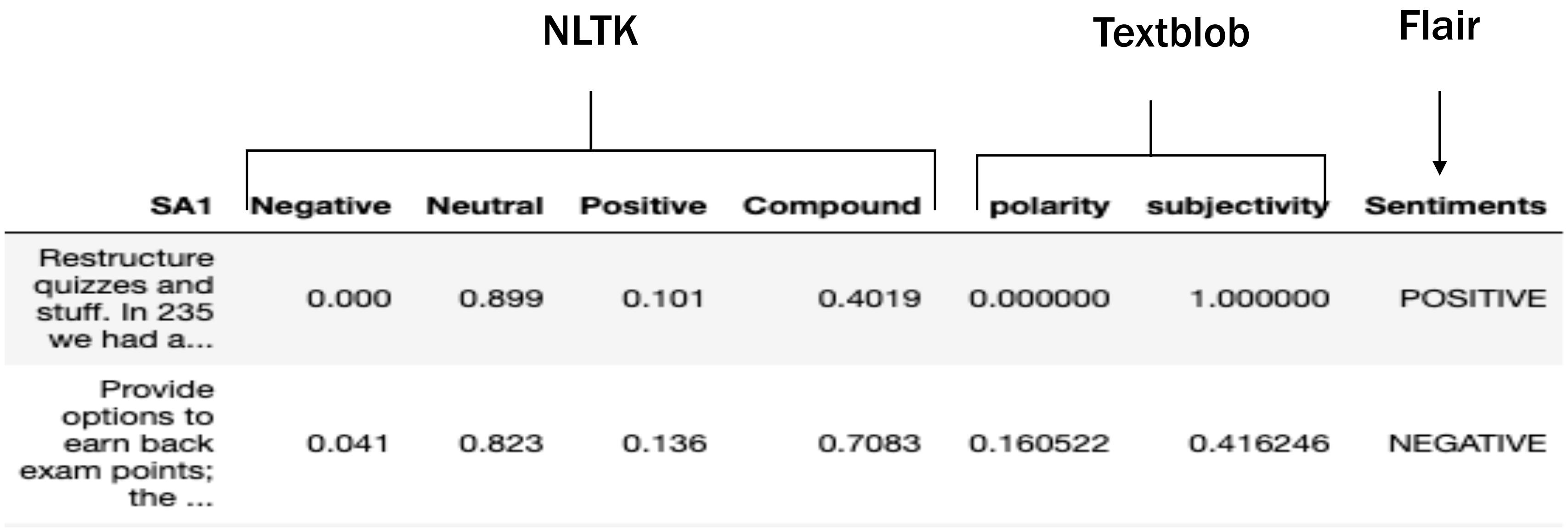
1. Latent Dirichlet Allocation (LDA)
2. Non-Negative Matrix Factorization (NMF)

Topics emerged:

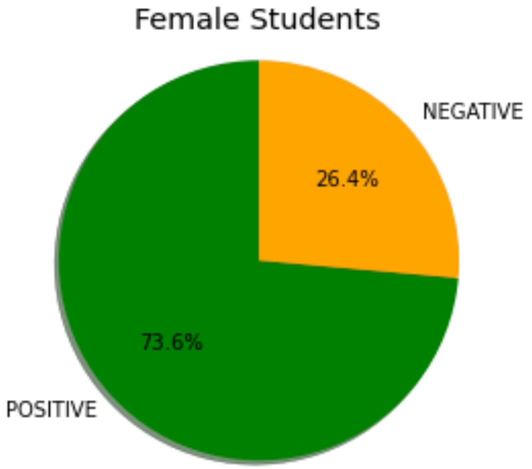
1. Assessment
2. Supporting Materials
3. Faculty Interactions

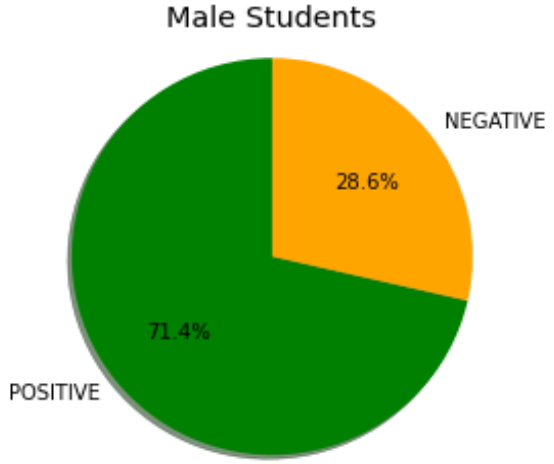
# RESULTS…

## Problem #2B:  Sentiment Analysis by Gender



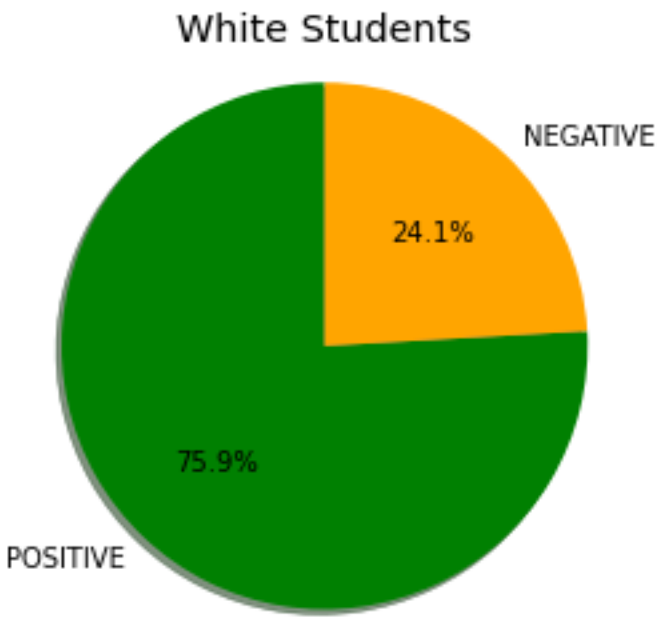
| | NLTK | | | | Textblob | | Flair |
|---|---|---|---|---|---|---|---|
| SA1 | Negative | Neutral | Positive | Compound | polarity | subjectivity | Sentiments |
| Restructure quizzes and stuff. In 235 we had a... | 0.000 | 0.899 | 0.101 | 0.4019 | 0.000000 | 1.000000 | POSITIVE |
| Provide options to earn back exam points; the ... | 0.041 | 0.823 | 0.136 | 0.7083 | 0.160522 | 0.416246 | NEGATIVE |

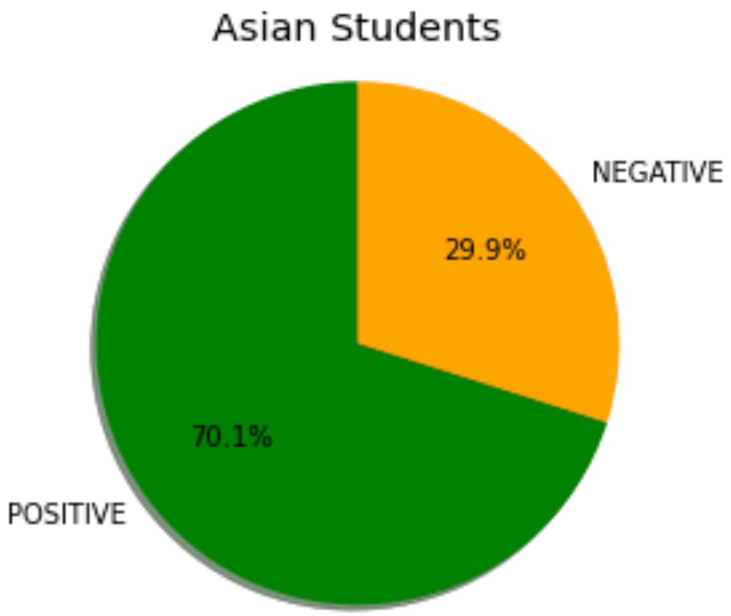


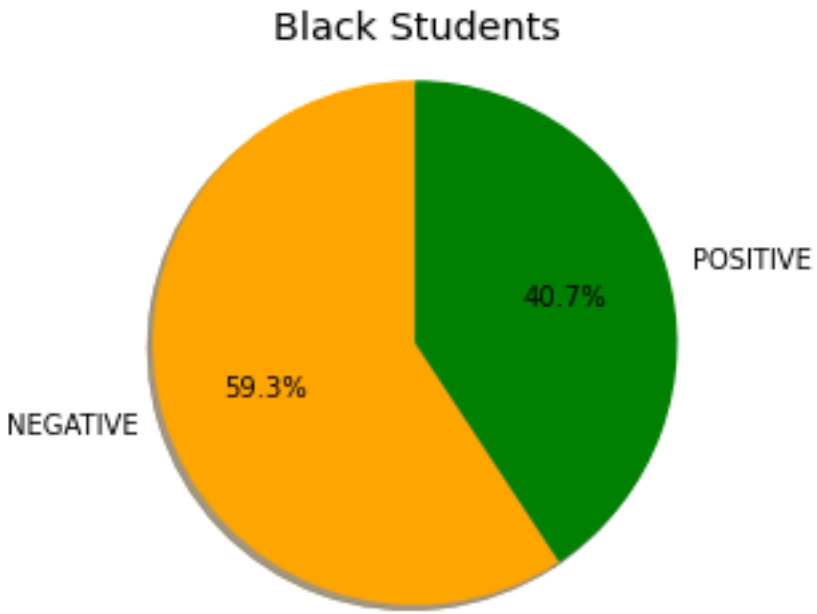


# RESULTS…

## Problem #2B: Sentiment Analysis by Demographic

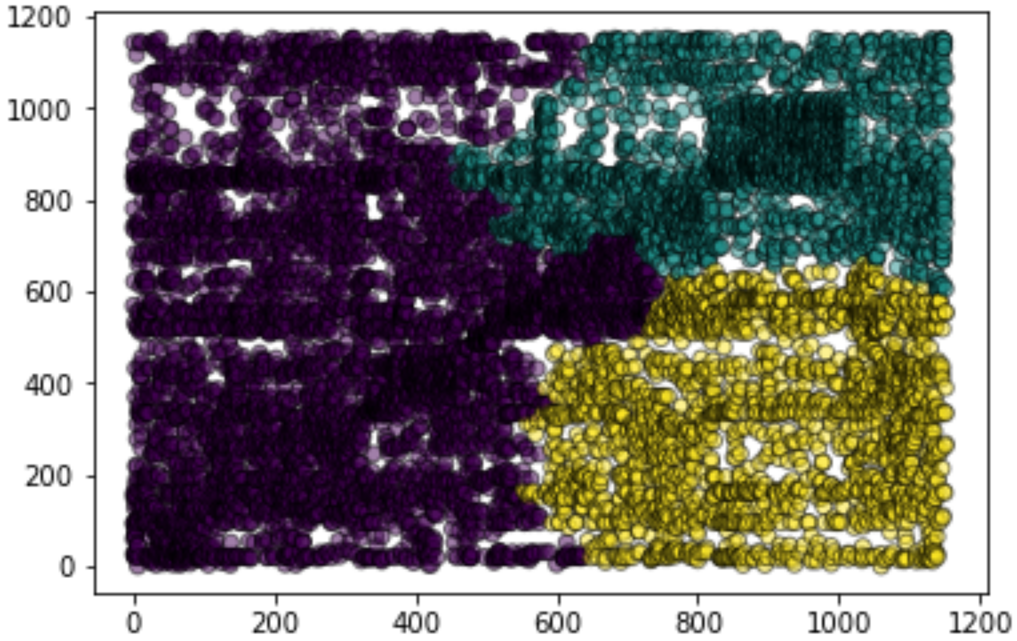






# RESULTS...

## Problem #2C: Semantic Similarity (work in progress)



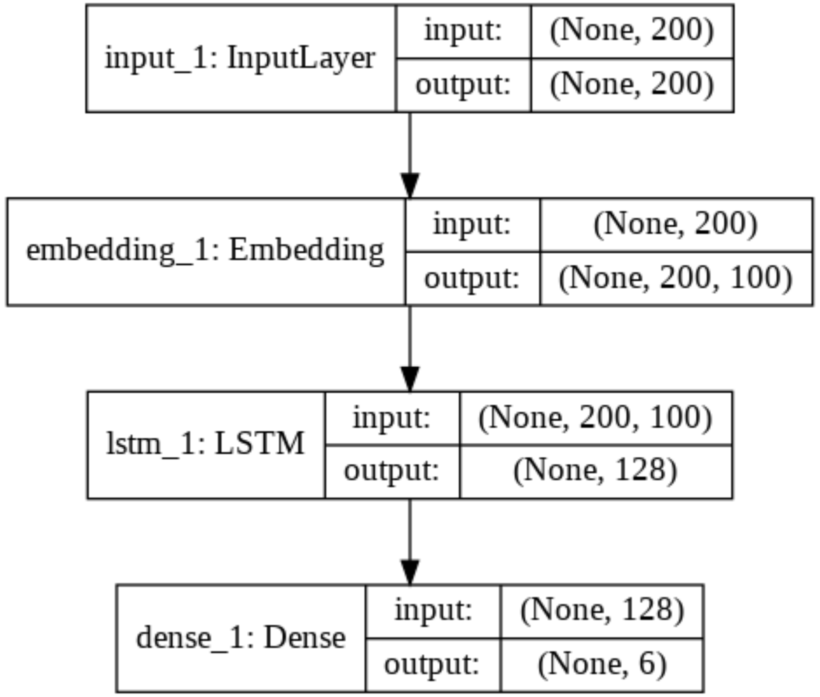
# THANK YOU!

# NOTES

# MULTI LABEL CLASSIFICATION

***A.Single Output Layer Model (Baseline):*** single dense layer with 4 outputs with a sigmoid activation functions and binary cross entropy loss functions. Each neuron in the output dense layer will represent one of the 4 output labels. The sigmoid activation function will return a value between 0 and 1 for each neuron. If any neuron's output value is greater than 0.5, it is assumed that the comment belongs to the class represented by that particular neuron.

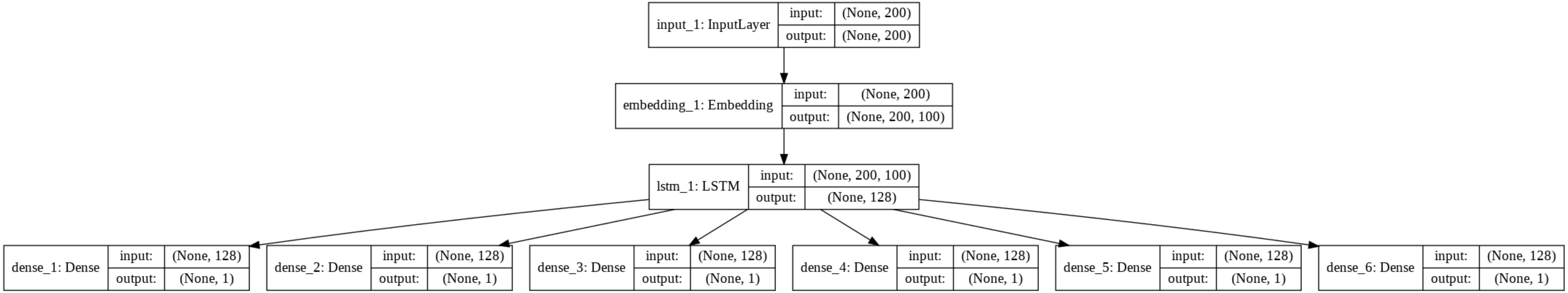
Our model will have one input layer, one embedding layer, one LSTM layer with 128 neurons and one output layer with 6 neurons since we have 6 labels in the output.



# MULTI LABEL CLASSIFICATION

***B.Multiple Output Layers Model:*** we will create one dense output layer for each label. We will have a total of 4 dense layers in the output. Each layer will have its own sigmoid function.

Our model will have one input layer, one embedding layer followed by one LSTM layer with 128 neurons. The output from the LSTM layer will be used as the input to the 6 dense output layers. Each output layer will have 1 neuron with sigmoid activation function. Each output will predict integer value between 1 and 0 for the corresponding label.



# P #2A: TOPIC MODELING

**Latent Dirichlet Allocation (LDA):** The LDA is based upon two general assumptions:

Documents that have similar words usually have the same topic

Documents that have groups of words frequently occurring together usually have the same topic.

**Non-Negative Matrix Factorization (NMF):** Non-negative matrix factorization is also a supervised learning technique which performs clustering as well as dimensionality reduction

# P #2B: SENTIMENT ANALYSIS

**NLTK:** NLTK’s Vader sentiment analysis tool uses a bag of words approach (a lookup table of positive and negative words) with some simple heuristics

**Textblob:** Textblob’s Sentiment Analysis works in a similar way to NLTK — using a bag of words classifier, but the advantage is that it includes Subjectivity Analysis too (how factual/opinionated a piece of text is)

**Flair:** Flair’s sentiment classifier is based on a character-level LSTM neural network which takes sequences of letters and words into account when predicting

# P #2C: SEMANTIC SIMILARITY

I represented each sentence by an embedding using infersent where, inferset is a *sentence embeddings* method that provides semantic representations for English sentences. It is trained on natural language inference data and generalizes well to many different tasks.

Secondly, I found responses that are semantically similar so the idea here is to index representation of each sentence and pick k(=10) NN (nearest neighbor) for each sentence based on distance threshold.

Thirdly, I found the Get prediction probability of responses pairs on semantic similarity classifiers. Think of step 2 as response generation (focusing on recall) and step 3 as focusing on precision. Of all the responses that are considered potential duplicates here we assign probability to each pair.

In the fourth step, Agglomerative clustering is used to merge clusters. Based on responses that are considered duplicates in step 3 we merge clusters using agglomerative clustering implementation in scikit. In agglomerative clustering all observations start as their own clusters and clusters are merged using the merge criteria specified until convergence, at which point no more merges are happening.