# ECE1786 | Project Name: DeCo.

## Project Final Report

**Team members:** Tan Xu (994567269), Mukesh Reddy Kayadapuram (1007993913)

**Word Count (Excluding Figures/References): 1988** **Penalty:** 0%

## Introduction

The goal of Project DeCo. is to create an application that can classify companies into multiple industry groups by analyzing their descriptions, annual reports, and other text artifacts that detail their business strategy and operations. Leading classification standards in the market only label companies with a single industry group, providing an incomplete picture that often ignores a company's new expansions or business ventures.

The GICS classification standard [1] is biased towards the largest business segment by revenue, leading to inaccurate labeling of companies, below are some examples of the limitations of a single industry label.

| 2018 Fiscal year | Sales |
|---|---|
| **Online stores (Internet retail)** | **54%** |
| Physical stores | 7% |
| **Third-party seller services (Internet retail)** | **18%** |
| Subscription services | 6% |
| Amazon Web Services | 11% |
| Sales of advertising services | 4% |
| **Total** | **100%** |

**Figure 1.** Amazon is classified as an industry group "Retailing", despite its significant operations, such as AWS, which is not represented in the classification. This overlooks fast-growing and profitable segments of a company.

| 2018 Fiscal Year | Power | Renewable Energy | Aviation | Oil & Gas | Healthcare | Transportation | Lighting | Capital |
|---|---|---|---|---|---|---|---|---|
| Revenue | 22% | 8% | 24% | 18% | 16% | 3% | 1% | 8% |
| Earnings | -8% | 3% | 63% | 4% | 36% | 6% | 1% | -5% |

**Figure 2.** General Electric is classified as an industry group of "Capital Goods" using the traditional single-industry group classification approach even though none of its 8 business segments is a majority. This groups all companies in the same situation into one category.

Our new multi-industry group classification approach allows for a more detailed and nuanced view of a company's operations. This can provide investors with additional insights when building a diversified stock portfolio with targeted industry exposures. A weighting system can be further developed to help compute the industry group ratio of a portfolio. For example, Amazon could be

further decomposed into "Retailing", "Software & Services", and "Media & Entertainment" to better reflect its e-commerce, cloud services, and content creation segments.

In today's rapidly changing business environment, companies often enter and exit new businesses quickly. NLP and ML models can be used to monitor companies' up-to-date information from different sources to detect changes in their operations. This allows for real-time updating of classifications and potential integration into data engineering pipelines. This approach shifts the decision-making process from relying only on revenue to using text-based data to classify a company.

## Illustration / Figure

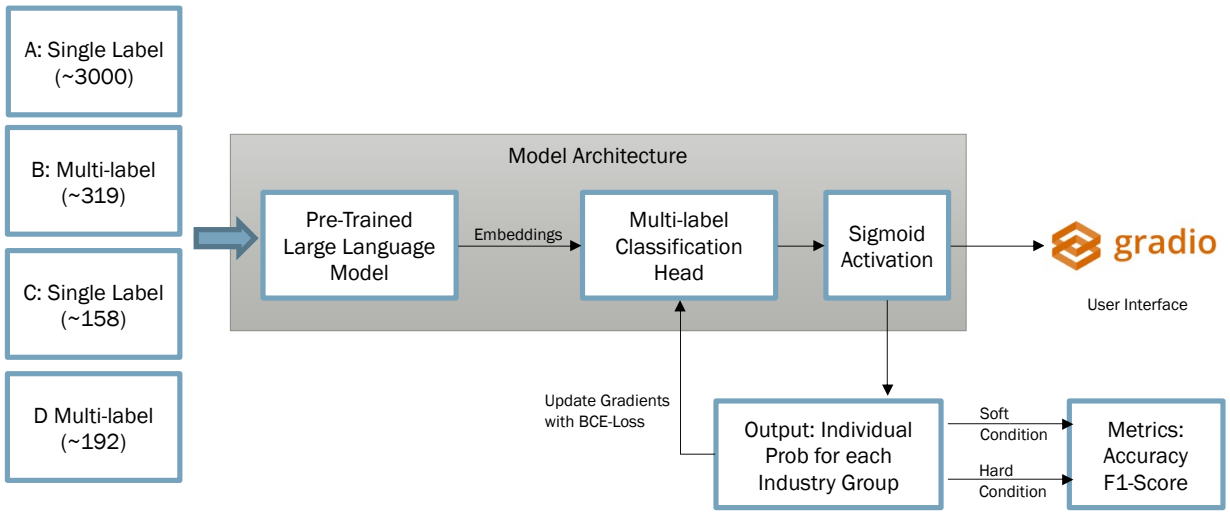



**Figure 3**. A general illustration of the project

## Background & Related Work

Several past attempts to improve company classification using machine learning, despite different approaches and goals. A recent work by S. Husmann, A. Shivarova, and R. Steinert (2022) [2] used the t-SNE algorithm to perform dimensional reduction on company balance sheet data. This data was then used for visual and exploratory analysis by financial experts or fed to a clustering algorithm to arrive at a data-driven classification system. This new clustering was then used to improve the decision-making process for classifying companies differently. The paper also explored the use of this new classification for investment portfolio optimization and company valuation multiplier estimation. Empirical studies showed positive results for both use cases thanks to the new company classifications.

Both our project and the above project aim to improve existing company classification methods, but we have taken a different approach. Our project uses NLP classification techniques based on companies' text artifacts to produce multi-label targets, while the above uses unsupervised clustering techniques based on financial metrics.

## Data and Data Processing

Our goal is to produce a multi-label classification for companies' text artifacts using the 24 industry groups defined by the GICS.

```
label_ls = ['Automobiles & Components','Banks','Capital Goods',
 'Commercial & Professional Services','Consumer Durables & Apparel',
 'Consumer Services','Diversified Financials','Energy',
 'Food & Staples Retailing','Food, Beverage & Tobacco',
 'Health Care Equipment & Services','Household & Personal Products',
 'Insurance','Materials','Media & Entertainment',
 'Pharmaceuticals, Biotechnology & Life Sciences','Real Estate',
 'Retailing','Semiconductors & Semiconductor Equipment',
 'Software & Services','Technology Hardware & Equipment',
 'Telecommunication Services','Transportation','Utilities']
len(label_ls)
```

```
24
```

**Figure 4.** 24 possible GICS *Industry Group* as class labels

We found that all existing datasets available on the internet come with only single labels, so we produced additional datasets for training and testing our model. There are four different datasets (referred to below as A, B, C, and D), two of these datasets contain single labels, and the other two contain multi-labels when appropriate.

We first consolidated data from several sources, including company long descriptions with Industry Group labels from Yahoo Finance [3], and GitHub [4], [5], [6], totaling 3483 samples.

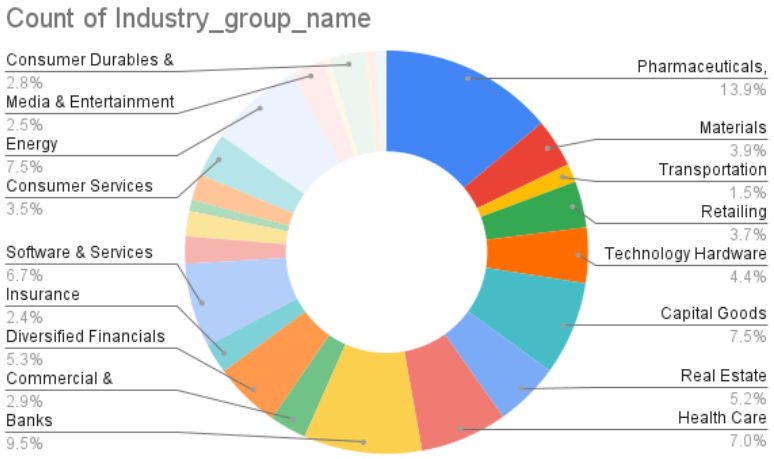



**Figure 5.** Distribution by Industry Group of data downloaded from Internet Open Sources (Yahoo and GitHub)

In phase 1 of the project, we split the above single labeled dataset into two datasets: A and B. Dataset A contains non-S&P500 stocks (~3000) and dataset B contains S&P500 stocks (~490). We then train one of our target models (See below section for details) on dataset A with some initial hyperparameter tuning to arrive at a relatively high-performing model. We run inference on dataset B to generate possible multi-labels, which are reviewed and updated manually. Due to time constraints, we were able to complete a total of 319 samples.

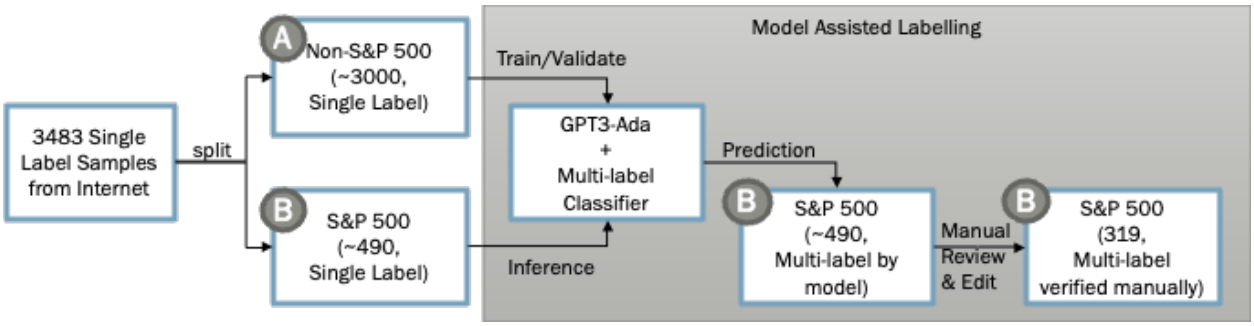



**Figure 6.** Process flow diagram to generate dataset A and model-assisted labeled dataset B

To create dataset C, we gather the definitions of the sub-industries from GICS documentation [1] and assign the corresponding industry group as a label. The resulting dataset C has 158 samples.

| Industry Group | Sub Industry Group | Sub Industry Group Description |
|---|---|---|
| Energy | Oil & Gas Drilling | Drilling contractors or owners of drilling rigs that contract their services for drilling wells. |
| Energy | Oil & Gas Equipment & Services | Manufacturers of equipment, including drilling rigs and equipment, and providers of supplies and services to companies involved in the drilling, evaluation and completion of oil and gas wells. |
| ... | ... | ... |
| Materials | Commodity Chemicals | Companies that primarily produce industrial chemicals and basic chemicals. Including but not limited to plastics, synthetic fibers, films, commodity-based paints and pigments, explosives and petrochemicals. Excludes chemical companies classified in the Diversified Chemicals, Fertilizers & Agricultural Chemicals, Industrial Gases, or Specialty Chemicals sub-industries. |
| ... | ... | ... |
| Capital Goods | Aerospace & Defense | Manufacturers of civil or military aerospace and defense equipment, parts or products. Includes defense electronics and space equipment. |
| ... | ... | ... |

**Figure 7.** GICS Sub-Industry and its definitions rolled up to Industry Group.

To create dataset D, we extracted text from various companies' annual reports and labeled them to create a multi-labeled dataset. Each sample has a description of up to 400 words and is associated with one or more labels. For example, below are two samples from Amazon's annual report:

| Ticker | Company | Annual Report Extract | Industry Groups |
|---|---|---|---|
| AMZN | Amazon.com Inc. | Our businesses are rapidly evolving and intensely competitive, and we have many competitors across geographies, including cross-border competition, and in different industries, including physical, e-commerce, and omnichannel retail, e[1]commerce services, web and infrastructure computing services, electronic devices, digital content, advertising, grocery, and transportation and logistics services. ... | ['Retailing'] |
| AMZN | Amazon.com Inc. | We serve consumers through our online and physical stores and focus on selection, price, and convenience. We design our stores to enable hundreds of millions of unique products to be sold by us and by third parties across dozens of product categories. Customers access our offerings through our websites, mobile apps, Alexa, devices, streaming, and physically visiting our stores. ... | ['Retailing'] |
| AMZN | Amazon.com Inc. | In the early days of AWS, people sometimes asked us why compute wouldn't just be an undifferentiated commodity. But, there, a lot more to compute than just a server. Customers want various flavors of compute (e.g. server configurations optimized for storage, memory, high-performance compute, graphics rendering, machine learning), multiple form factors (e.g. fixed instance sizes, portable containers, serverless functions), various sizes and optimizations of persistent storage, and a slew of networking capabilities. ... | ['Retailing', 'Software & Services'] |
| AMZN | Amazon.com Inc. | We have organized our operations into three segments: North America, International, and Amazon Web Services. These segments reflect the way the Company evaluates its business performance and manages its operations. As we were defining AWS and working backwards on the services we thought customers wanted, we kept triggering one of the biggest tensions in product development, where to draw the line on functionality in V1. ... | ['Retailing', 'Software & Services'] |
| AMZN | Amazon.com Inc. | This track record of frequent invention is not only why more sports entities are choosing to work with Prime Video, but also why so many large entertainment companies have become Prime Video Channels partners. ... | ['Retailing', 'Media & Entertainment'] |
| AMZN | Amazon.com Inc. | We started in 2006 with an offering called Amazon Unbox where customers could download about a thousand movies from major studios. This made sense as bandwidth was slower those days (it would take an hour to download a video). ... | ['Retailing', 'Media & Entertainment'] |

**Figure 8.** Texts extracted from Amazon's annual report (truncated due to space limitations) and split into various samples with different labels.

Amazon's 2021 annual report was used to create 6 samples, each labeled with the relevant industry groups. Labels were checked for accuracy and then one-hot encoding vectors were generated using the Scikit-learn library's "preprocessing" package.

## Architecture and Software

In our industry group classification task, we have selected the best language models currently available in the natural language processing (NLP) industry, GPT-3, and BERT, to produce sentence embeddings. The architecture for our target and baseline models is similar, with both models using a multi-label classification head with a varying number of linear layers. We experimented with various sizes for the classification head in our target models, but the models were only able to generalize well with four or more linear layers.

The multi-label classification head with sigmoid activation at the end receives the input sentence embedding from the pre-trained model, and the weights of the linear layers are updated using a binary cross-entropic loss function and the Adam optimizer. The linear layers have leaky ReLU activations between the neurons, and we carefully selected the best set of hyperparameters for each model.

All three, one Baseline & two Target models are deployed in Gradio UI and ready for the user to use.

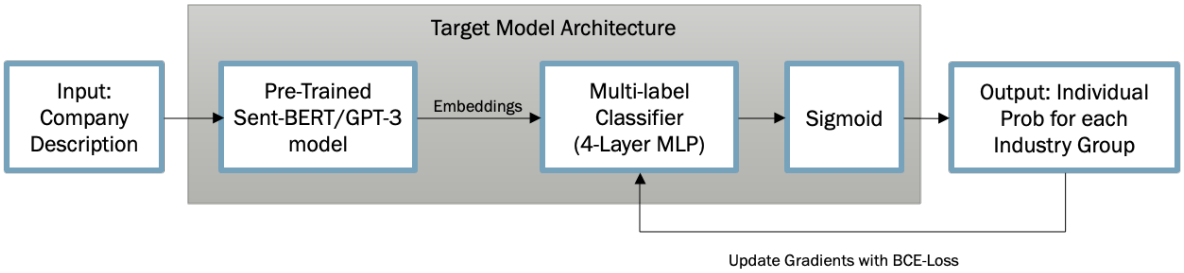



**Figure 9.** Flow diagram for Target Models

Training our target models was difficult due to the nature of the data, but we were able to make the models generalize well enough to achieve satisfactory accuracy and predictions. We chose to use a multi-label classification approach because a company may belong to multiple industry groups that are not recognized by commonly available APIs and methods from sources such as yfinance, Eikon, and Crunchbase.

Initialization for our target models:

```
class Industry_Group_Classification_Model(torch.nn.Module):
  def __init__(self, emb_length=768, pred_length=24):
      super().__init__()
      self.layer1 = nn.Linear(emb_length, int(emb_length/2))
      self.layer2 = nn.Linear(int(emb_length/2), int(emb_length/4))
      self.layer3 = nn.Linear(int(emb_length/4), int(emb_length/12))
      self.layer4 = nn.Linear(int(emb_length/12), pred_length)

  def forward(self, x_emb):
    x_emb = torch.as_tensor(x_emb, dtype=torch.float)
    if torch.cuda.is_available(): x_emb = x_emb.cuda()

    out1 = F.leaky_relu(self.layer1(x_emb))
    out2 = F.leaky_relu(self.layer2(out1))
    out3 = F.leaky_relu(self.layer3(out2))
    prediction = self.layer4(out3)

    return prediction
```

**Figure 10.** Target Models initialization

## Target Model 1: BERT-based

This model obtains sentence embeddings from the "sentence-transformers/all-mpnet-base-v2" [7] model in the Hugging Face Sentence BERT family. This model is trained on PLM tasks and uses the auxiliary position information to reduce position discrepancy, providing a good balance of performance and speed. The pre-trained BERT model is used to generate feature vectors of 768 dimensions, which are then fine-tuned using an MLP multi-class classification head with 383,128 trainable parameters.

Hyperparameters:
*Linear layers count: 4*
*Batch size: 64*
*Epochs: 25*
*Learning rate: 0.001*
*Cut off: 0.65*
*Optimizer: Adam*

## Target Model 2: GPT-3-based

This model obtains sentence embeddings from the "text-similarity-ada-001" [8] model in the OpenAI GPT-3 Ada family. The GPT-3 Ada is capable of simple tasks and is typically the fastest and lowest-cost model in the GPT-3 series. The pre-trained GPT-3 Ada model is used to generate feature vectors of 1024 dimensions, which are then fine-tuned using an MLP multi-class classification head with 680,037 trainable parameters.

Hyperparameters:
*Linear layers count: 4*
*Batch size: 16*
*Epochs: 25*
*Learning rate: 0.001*
*Cut off: 0.9*
*Optimizer: Adam*

## Baseline Model

For the baseline model, we are fine-tuning the Sentence-BERT with a single-layer MLP classification head with leaky ReLU activation. The prediction is then fed to the sigmoid to produce probability-like outputs for each label class. It is trained and validated on dataset B with a 90/10 split and tested on dataset D with a batch size of 32, a learning rate of 0.005, and 3000 epochs. The decision threshold used for accuracy is 0.3, and accuracy is determined by the exact match of predicted and target components (hard condition).

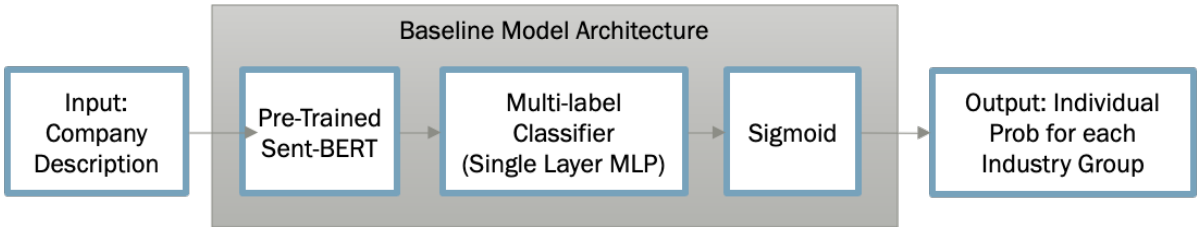
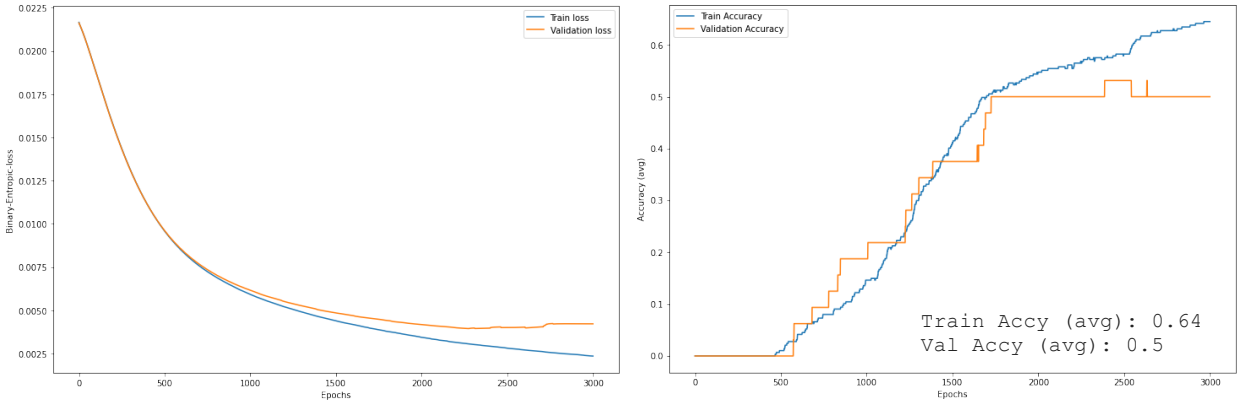
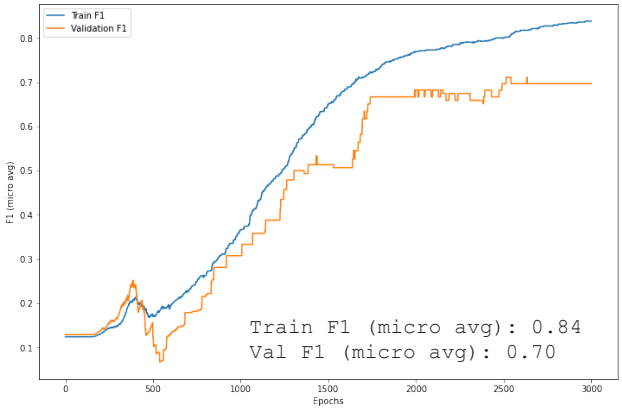



**Figure 11.** Baseline model architecture with a single-layer MLP






$$\text{MicroAvg} = \frac{(TP_1+TP_2+\ldots+TP_n)}{(TP_1+TP_2+\ldots+TP_n+FP_1+FP_2+\ldots+FP_n)}$$

**Figure 12.** Top left, loss curve showing training and validation loss/
Top right, avg of accuracy score (hard condition, has to match exactly)
Bottom left, micro avg of F1 score
Bottom right, micro avg calculation

## Quantitative Results

During training, we experimented with various methods and found that the best performers were models trained on single-labeled dataset A first, and then fine-tuned with hand-labeled datasets B and GICS definition dataset C. Both models are then validated/tested with hand-labeled company annual report dataset D. The saved weights and implementation for our target and baseline models are available in our GitHub repository for reproducibility.

To evaluate the performance of the target models quantitatively, we used two measures:

- **Soft Condition** ({True}⊆ {Prediction}): In this measure, a prediction is considered true as long as the model predicted all of the true labels, even if it also included extra labels. This is to tolerate small signals from hidden contexts from inputs.
- **Hard Condition** ({True}={Prediction}): In this measure, a prediction is only considered true if it exactly matches the true labels, without extra labels included.

### Target Model 1: BERT Based

Among the target models, the BERT-based model had the highest metrics when the decision threshold was set to 0.9 based on the validation/test data. The threshold value may vary depending on the underlying data used for evaluation. We also observed that the models performed well at a lower threshold of 0.5.

```
outputs = torch.rand(5)
print("Original:",outputs)
cut_off = 0.5
y_pred = (outputs > cut_off).float()
print("Cut-off=0.5:",y_pred)

cut_off = 0.9
y_pred = (outputs > cut_off).float()
print("Cut-off=0.9:", y_pred)
```

```
Original: tensor([0.5724, 0.5066, 0.1565, 0.2541, 0.9899])
Cut-off=0.5: tensor([1., 1., 0., 0., 1.])
Cut-off=0.9: tensor([0., 0., 0., 0., 1.])
```

**Figure 13.** Example for working of cut off

As the number of epochs increased, the training loss gradually decreased, indicating that the model was learning well. We stopped training before the model overfit the data and could generalize well to the validation/test data.

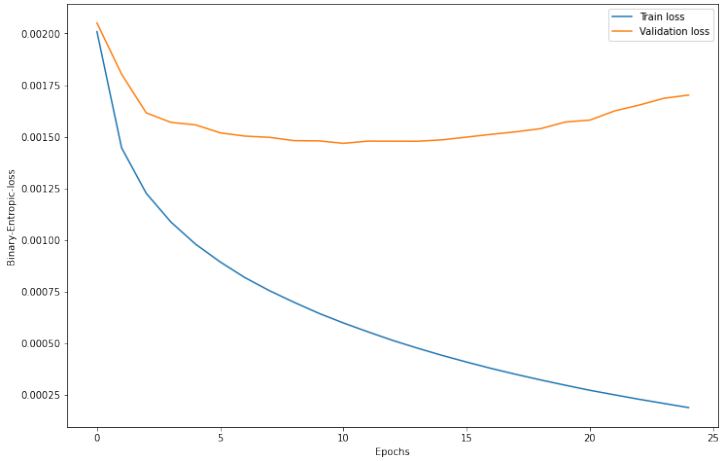



**Figure 14.** Loss curves for BERT Based Target model

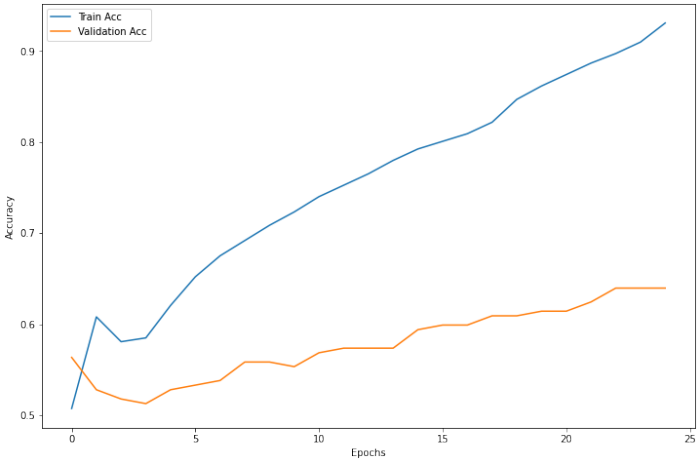



**Figure 15.** Accuracy curves for BERT Based Target model with soft condition

Hard condition measures are lower as expected, with validation/test accuracy at 0.54 and F1-score (micro Avg) at 0.71.

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Automobiles & Components | 1.000000 | 0.785714 | 0.880000 | 14.0 |
| Banks | 0.888889 | 0.727273 | 0.800000 | 11.0 |
| Capital Goods | 0.705882 | 0.923077 | 0.800000 | 13.0 |
| Commercial & Professional Services | 0.357143 | 0.500000 | 0.416667 | 10.0 |
| Consumer Durables & Apparel | 0.857143 | 0.666667 | 0.750000 | 9.0 |
| Consumer Services | 0.857143 | 1.000000 | 0.923077 | 6.0 |
| Diversified Financials | 0.800000 | 0.705882 | 0.750000 | 17.0 |
| Energy | 0.538462 | 1.000000 | 0.700000 | 7.0 |
| Food & Staples Retailing | 0.833333 | 1.000000 | 0.909091 | 5.0 |
| Food, Beverage & Tobacco | 0.857143 | 0.857143 | 0.857143 | 14.0 |
| Health Care Equipment & Services | 0.833333 | 0.833333 | 0.833333 | 6.0 |
| Household & Personal Products | 1.000000 | 0.600000 | 0.750000 | 5.0 |
| Insurance | 0.800000 | 1.000000 | 0.888889 | 8.0 |
| Materials | 0.411765 | 1.000000 | 0.583333 | 7.0 |
| Media & Entertainment | 1.000000 | 0.727273 | 0.842105 | 22.0 |
| Pharmaceuticals, Biotechnology & Life Sciences | 0.571429 | 0.800000 | 0.666667 | 5.0 |
| Real Estate | 0.600000 | 0.857143 | 0.705882 | 7.0 |
| Retailing | 0.909091 | 0.588235 | 0.714286 | 17.0 |
| Semiconductors & Semiconductor Equipment | 0.714286 | 1.000000 | 0.833333 | 5.0 |
| Software & Services | 0.375000 | 0.176471 | 0.240000 | 17.0 |
| Technology Hardware & Equipment | 0.600000 | 0.352941 | 0.444444 | 17.0 |
| Telecommunication Services | 0.666667 | 0.571429 | 0.615385 | 14.0 |
| Transportation | 1.000000 | 0.800000 | 0.888889 | 10.0 |
| Utilities | 0.500000 | 1.000000 | 0.666667 | 4.0 |
| micro avg | 0.719512 | 0.708000 | 0.713710 | 250.0 |
| macro avg | 0.736529 | 0.769691 | 0.727466 | 250.0 |
| weighted avg | 0.752770 | 0.708000 | 0.708127 | 250.0 |
| samples avg | 0.762690 | 0.760575 | 0.738240 | 250.0 |

**Figure 15.** Classification Report for BERT-Based Target model

## Target Model 2: GPT3 Based

GPT3-based target model has the highest metrics when the decision threshold = 0.5.

The training loss gradually decreased as the number of epochs increased, indicating that the model was learning well. We stopped training before the model overfit the data and could generalize well to the validation/test data. The validation loss may not have fully converged, but it was the best result achieved with the given hyperparameters, resulting in high accuracy.

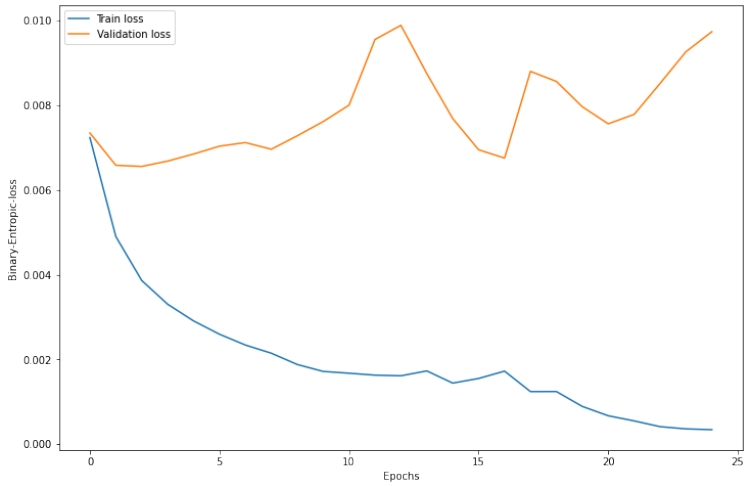



**Figure 16.** Loss curves for GPT3 Based Target model

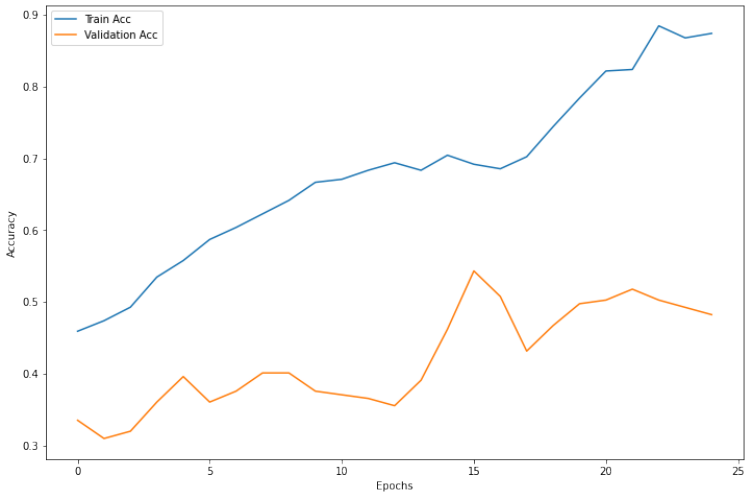



**Figure 17.** Accuracy curves for GPT-3 Based Target model with soft condition

Hard condition measures are also lower with validation/test accuracy at 0.42 and F1-score (micro Avg) at 0.65.

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Automobiles & Components | 1.000000 | 0.642857 | 0.782609 | 14.0 |
| Banks | 1.000000 | 0.636364 | 0.777778 | 11.0 |
| Capital Goods | 0.636364 | 0.538462 | 0.583333 | 13.0 |
| Commercial & Professional Services | 0.250000 | 1.000000 | 0.400000 | 10.0 |
| Consumer Durables & Apparel | 1.000000 | 0.444444 | 0.615385 | 9.0 |
| Consumer Services | 1.000000 | 0.666667 | 0.800000 | 6.0 |
| Diversified Financials | 0.777778 | 0.823529 | 0.800000 | 17.0 |
| Energy | 1.000000 | 0.428571 | 0.600000 | 7.0 |
| Food & Staples Retailing | 0.400000 | 0.800000 | 0.533333 | 5.0 |
| Food, Beverage & Tobacco | 1.000000 | 0.357143 | 0.526316 | 14.0 |
| Health Care Equipment & Services | 1.000000 | 0.500000 | 0.666667 | 6.0 |
| Household & Personal Products | 1.000000 | 0.800000 | 0.888889 | 5.0 |
| Insurance | 1.000000 | 0.625000 | 0.769231 | 8.0 |
| Materials | 1.000000 | 0.571429 | 0.727273 | 7.0 |
| Media & Entertainment | 0.944444 | 0.772727 | 0.850000 | 22.0 |
| Pharmaceuticals, Biotechnology & Life Sciences | 0.571429 | 0.800000 | 0.666667 | 5.0 |
| Real Estate | 0.777778 | 1.000000 | 0.875000 | 7.0 |
| Retailing | 0.727273 | 0.470588 | 0.571429 | 17.0 |
| Semiconductors & Semiconductor Equipment | 0.571429 | 0.800000 | 0.666667 | 5.0 |
| Software & Services | 0.571429 | 0.470588 | 0.516129 | 17.0 |
| Technology Hardware & Equipment | 0.800000 | 0.235294 | 0.363636 | 17.0 |
| Telecommunication Services | 1.000000 | 0.500000 | 0.666667 | 14.0 |
| Transportation | 1.000000 | 0.800000 | 0.888889 | 10.0 |
| Utilities | 0.666667 | 0.500000 | 0.571429 | 4.0 |
| micro avg | 0.703704 | 0.608000 | 0.652361 | 250.0 |
| macro avg | 0.820608 | 0.632653 | 0.671139 | 250.0 |
| weighted avg | 0.829104 | 0.608000 | 0.662512 | 250.0 |
| samples avg | 0.658206 | 0.641286 | 0.623689 | 250.0 |

**Figure 18.** Classification Report for GPT3-Based Target model

## Qualitative Results

Upon examining UI with test dataset D, we observed that the Sent-BERT model sometimes made non-sensible predictions, but not with the GPT-3 model. Therefore, we think GPT-3 model performs better qualitatively. Examples of this are shown below.

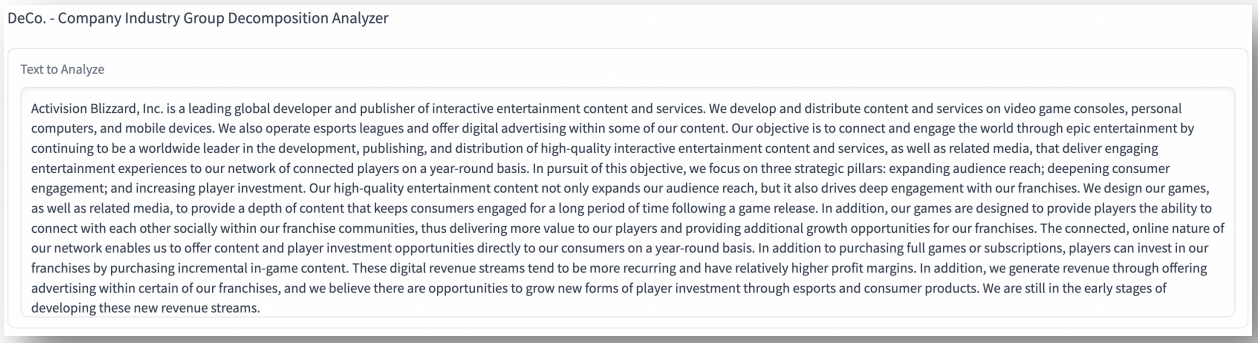



**Figure 19.** Input text to the Gradio UI, extracted from Activision Blizzard, a well-known game developer

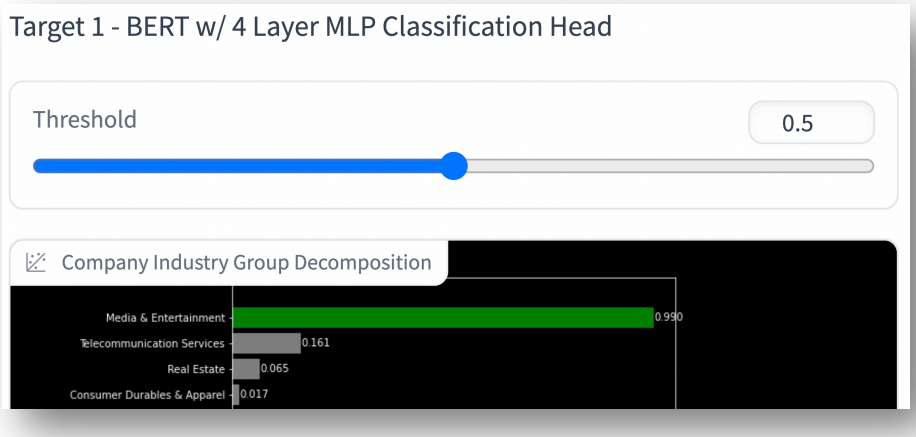



**Figure 20.** Sent-BERT target model prediction on Activision Blizzard

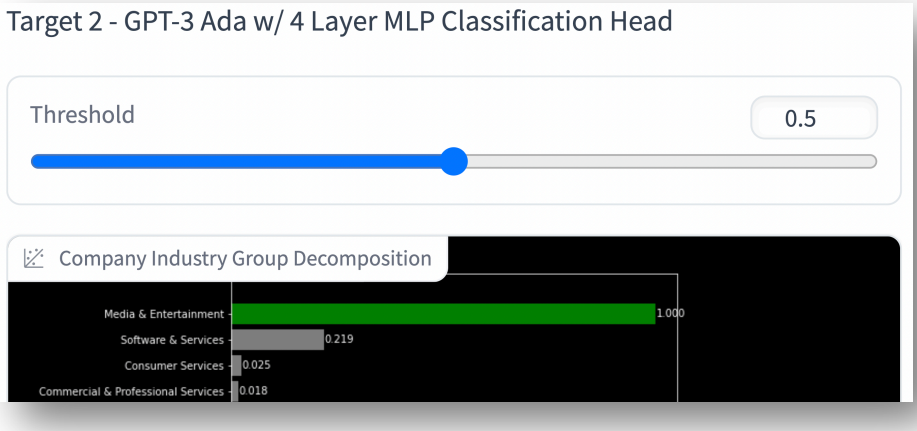



**Figure 21.** GPT-3 target model prediction on Activision Blizzard

The GPT-3 model accurately predicted the "Media & Entertainment" label with confidence, while Sent-BERT's top prediction was correct but the following two labels were less relevant.

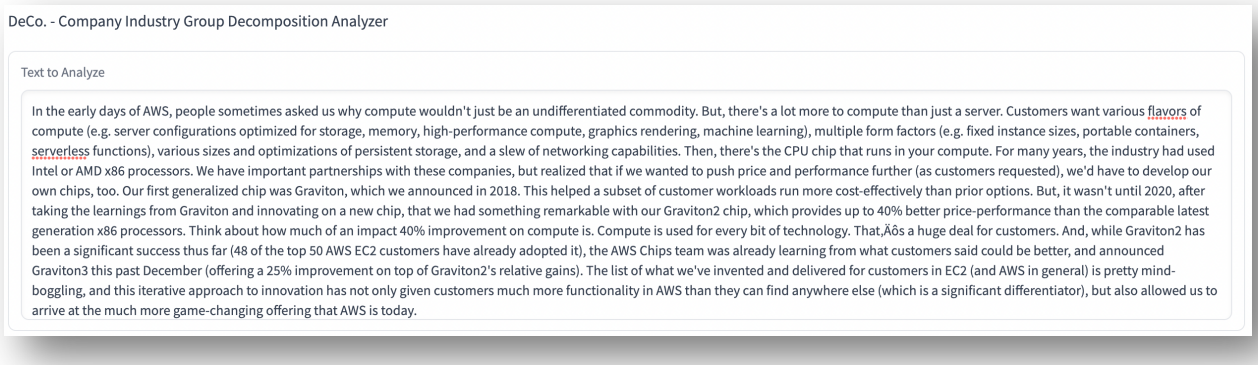



**Figure 22.** Input text to the Gradio UI, extracted from Amazon 2021 annual report, related to AWS.

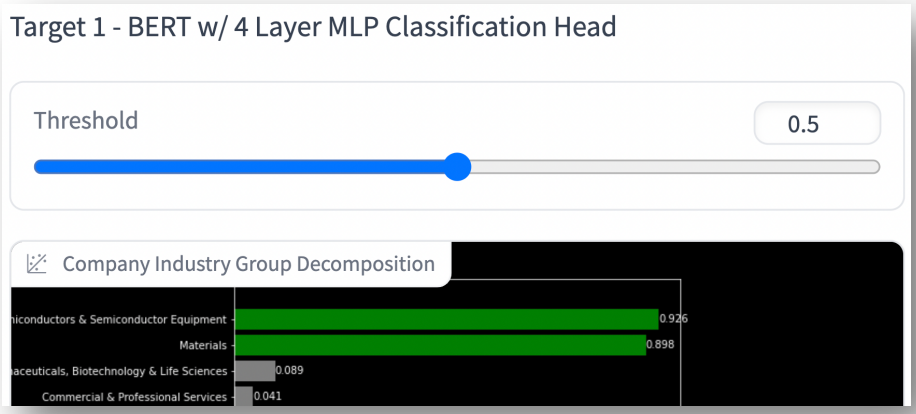



**Figure 23.** Sent-BERT target model prediction on AWS-related texts

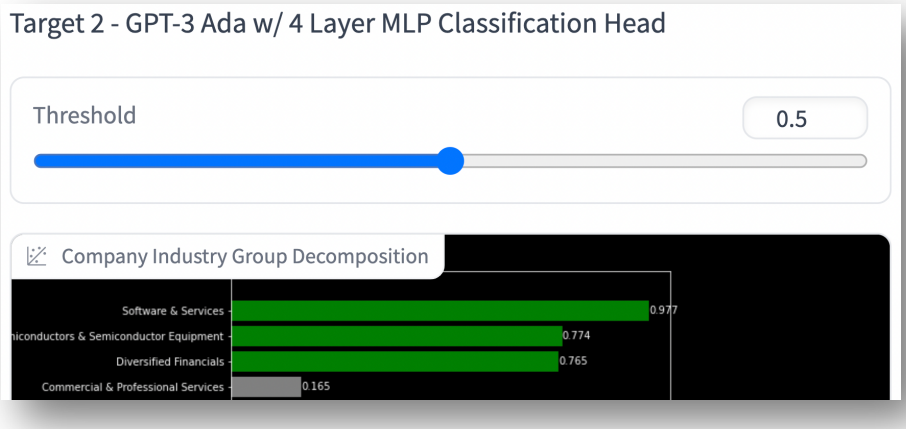



**Figure 24.** GPT-3 target model prediction on AWS-related texts

Sent-BERT provided nonsensical labels, such as "Materials" and "Pharmaceuticals, Biotechnology & Life Science", while GPT-3 accurately classified AWS text as "Software & Services" and "Semiconductors & Semiconductor Equipment" based on mentions of computer chips.

## Discussion and Learnings

1. **BERT-based target model shows better quantitative results:** BERT models utilize semantic relations between words, which may account for their superior performance in the quantitative metrics compared to GPT3-Ada based. The small size embedding of the GPT3-Ada model may also have contributed to its lower performance.

2. **GPT3-based target model shows better qualitative results:** When tested on a hard company description, the GPT3-Ada-based target model produced results closer to expectation, while the BERT-based model made some drastically incorrect predictions.

3. **GPT3-Ada model used for assisted labeling of SNP500 dataset:** A GPT3-Ada-based classification model was trained on single-label data and used to generate multi-label predictions for the SNP500 dataset, which facilitated the hand-labeling process. The model was able to provide partially correct multi-labels even in cases where it had not seen multi-label samples.

4. **Possible improvements:**
    - Using a larger GPT-3 model like Davinci could improve performance.
    - Creating more multi-label data would likely improve model performance.

## Individual Contributions

Tan's contributions:

1. Idea generation and proposal document/presentation slides
2. Manual labeling of dataset D (197 samples) from company annual reports
3. Progress report (50%)
4. Review/update of ~160 model-assisted labeled samples to create multi-label dataset B (50% of efforts)
5. Development, training, and fine-tuning of baseline Sent-BERT model with MLP classification head using dataset B for training/validation and testing on dataset D
6. Research and implementation of accuracy metrics for multi-label classification, review of target model implementations
7. Gradio UI creation and investigation of model predictions
8. Final presentation and Final report preparation (50%)

Mukesh's contributions:

1. Project proposal contribution and fine-tuning of submission files
2. Collection of dataset A (~3000 samples) from the Internet
3. Training of GPT3-based target model with the single label to get top 5 multilabel predictions for model-assisted labeling on dataset B
4. Progress report contribution (50%)

5. Review/update of ~160 model-assisted labeled samples to create multi-label dataset B (50% of efforts)
6. Collection of dataset C (~158 samples) from the GICS official document
7. Development, training, and fine-tuning of 2 target models, generation of quantitative metrics for target models
8. Assistance with a final presentation and Final report preparation (50%)

## References


[1] "S&P Global," [Online]. Available: https://www.spglobal.com/marketintelligence/en/documents/112727-gics-mapbook_2018_v3_letter_digitalspreads.pdf.

[2] A. S. R. S. Sven Husmann, "ScienceDirect," 2022. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0957417422000914.

[3] "Yahoo Finance," [Online]. Available: https://ca.finance.yahoo.com/.

[4] Sivalavida, "GitHub," 2020. [Online]. Available: https://github.com/Sivalavida/Text-based-Industry-Classification/blob/master/data_in/ticker_to_gics.csv.

[5] conwuzurike, "GitHub," 2021. [Online]. Available: https://github.com/conwuzurike/BA-870/blob/main/Final%20Project/classifications.csv.

[6] BinghanJiang0202, "GitHub," 2022. [Online]. Available: https://github.com/BinghanJiang0202/BinghanJiang0202.github.io/blob/main/tic_rev_ind.csv.

[7] "Hugging Face," [Online]. Available: https://huggingface.co/sentence-transformers/all-mpnet-base-v2.

[8] "OpenAI," [Online]. Available: https://beta.openai.com/docs/guides/embeddings/similarity-embeddings.

**Permissions**

| | Tan | Mukesh |
|---|---|---|
| **Permission to post video** | Yes | Yes |
| **Permission to post final report** | Yes | Yes |
| **Permission to post source code** | Yes | Yes |