| Internship Project Title | Automate Emotional Analysis of Textual Comments and Feedback |
| --- | --- |
| Name of the Company |  |
| Name of the Industry Mentor |  |
| Name of the Institute | Amity University |

| Start Date | End Date | | Total Effort (hrs.) | | Project Environment | Tools used |
| --- | --- | --- | --- | --- | --- | --- |
| 30-July-25 | 30-Aug-25 | | ~60 hrs | | Google colab | Python, Hugging Face Transformers, PyTorch, Scikit-learn, Pandas, Matplotlib, W&B, NLTK |
| Milestone #1 |  | Milestone: | | Model training, validation, and initial evaluation completed; prediction pipeline partially implemented. Remaining work includes advanced evaluation, deployment, and report finalization. | | |

**TABLE OF CONTENT**

**Acknowledgements**

I would like to express my sincere gratitude to the vast community of learners, educators, and developers who share their knowledge freely on the internet. This project was completed entirely through my independent efforts, with valuable guidance drawn from freely available online resources, articles, and tutorials on platforms such as YouTube, documentation sites, and developer forums.

Although I did not work under any formal mentor, institution, or company for this project, the collective wisdom and generosity of online educators helped me navigate challenges, deepen my understanding, and refine my skills. I am truly grateful to these contributors, whose work has made learning accessible to all.

Finally, I acknowledge my own dedication, persistence, and curiosity, which allowed me to take this project from concept to completion.

**Objective**

To design and implement an automated system that can identify and classify multiple emotions from textual comments and feedback, enabling organizations to analyze customer sentiment and emotional tone at scale.

**Introduction / Description of Internship**

This internship focuses on developing an **AI-powered multi-label emotion classification system** using the **GoEmotions dataset** and **BERT** architecture. The system should process short text inputs such as customer comments, reviews, or social media feedback and predict the presence of multiple emotions simultaneously.

By automating emotional analysis, the project aims to:

* Reduce manual effort in sentiment review.
* Provide richer, more nuanced emotional insights beyond simple positive/negative classification.
* Enable real-time feedback monitoring for customer experience improvement.

**Internship Activities**

*(Completed so far — ~50% completion)*

**Environment Setup**

* Configured Google Colab environment with GPU support.
* Installed and imported all necessary Python libraries.

**Dataset Acquisition & Preprocessing**

* Loaded **GoEmotions dataset** from Hugging Face Datasets.
* Applied text cleaning (lowercasing, removing punctuation, slang normalization).

**Tokenization**

* Implemented BertTokenizer with padding and truncation to max length 128.
* Batched tokenization for train, validation, and test sets.

**Model Architecture**

* Customized BertForSequenceClassification for **multi-label classification** with BCEWithLogitsLoss.
* Created **Custom Data Collator** to handle label tensors in float format.

**Training**

* Fine-tuned bert-base-uncased model for 3 epochs.
* Used AdamW optimizer with learning rate 2e-5.
* Implemented evaluation metrics (accuracy, macro-precision, recall, F1).

**Initial Evaluation**

* Validated on test set; results show promising accuracy for common emotions.
* Began plotting **label distribution** and **confusion matrices** for top emotions.

**Custom Prediction Function**

* Implemented pipeline for predicting emotions from custom text input.

### Remaining Activities *(Planned for next 50% of project)*

* **Advanced Evaluation**
  + Fine-tune prediction thresholds for better recall on rare emotions.
  + Generate ROC curves and per-class F1 scores.
* **Error Analysis**
  + Identify misclassified examples and analyze patterns.
  + Adjust preprocessing for ambiguous or sarcastic text.
* **Model Optimization**
  + Experiment with RoBERTa/DistilBERT for faster inference.
  + Train with class weighting or oversampling for rare emotions.
* **Deployment**
  + Create a web interface using **Gradio** or **Streamlit**.
  + Integrate real-time feedback input and prediction.
* **Reporting & Documentation**
  + Prepare a final project report with visualizations.
  + Compile code and results into GitHub repository.
  + Record a short demo video of the system in action.

**Approach / Methodology**

**Architecture:** Pre-trained BERT fine-tuned for multi-label classification.

**Loss Function:** BCEWithLogitsLoss for independent sigmoid-based probability per label.

**Evaluation:** Accuracy, macro precision, recall, and F1 score.

**Prediction:** Apply sigmoid threshold (0.5 default) to classify multiple emotions per text.

**Visualization:** Matplotlib and Seaborn for plotting metrics and confusion matrices.

**Assumptions**

* Input text is in English and free from unsupported special characters.
* The dataset sufficiently covers the range of emotions needed for classification.
* The model will be deployed on GPU-backed infrastructure for real-time predictions.

**Exceptions / Exclusions**

* Sarcasm, irony, and figurative speech not explicitly modeled.
* Not designed for multi-lingual emotion detection in current phase.

**Charts, Table, Diagrams** *(In progress)*

* Label frequency distribution for GoEmotions dataset.
* Confusion matrix heatmap for top 5 frequent emotions.  
   *(Final charts to be completed after advanced evaluation.)*

**Algorithms**

**Transformer Architecture:** BERT with self-attention layers.

**Activation:** Sigmoid for independent multi-label probability prediction.

**Optimization:** AdamW with learning rate decay.

**Challenges & Opportunities**

**Challenges:**

* Handling rare emotions with very few examples.
* Balancing recall and precision in multi-label predictions.

**Opportunities:**

* Extending model for multiple languages.
* Integrating with sentiment dashboards for business analytics.

**Risk Vs Reward**

**Risk:** Overfitting to dataset style, limited generalization to unseen domains.

**Reward:** Highly accurate, scalable emotion classifier for various industries (customer service, HR, marketing).

**Reflections on the Internship**

* Learned practical fine-tuning of transformer models for real-world NLP tasks.
* Improved skills in handling multi-label classification challenges.
* Gained experience in integrating model training with live prediction pipelines.

**Recommendations**

* Collect domain-specific datasets for better business alignment.
* Implement adaptive thresholding for better performance on rare emotions.
* Explore explainable AI methods for transparency in predictions.

**Outcome / Conclusion**

* Successfully implemented a working BERT-based multi-label emotion classifier (~50% complete).
* Achieved promising initial accuracy and macro F1 score on validation set.
* Prepared prediction pipeline for unseen text.
* Remaining work will focus on optimization, advanced evaluation, deployment, and final reporting.

**Enhancement Scope**

* Sarcasm detection integration.
* Multi-lingual emotion recognition.
* Interactive dashboard for emotion analytics.

**Link to code and executable file**

[https://github.com/KashishMaurya/Automate-Emotional-Analysis-of-Textual-Comments-and-Feedback] (to be updated in final report)

**Research questions and responses**

**Q1:** Can BERT effectively capture overlapping emotional states in short text?  
**A1:** Yes, initial results indicate strong performance for common emotions, with room for improvement on rarer ones.

**Q2:** Does text cleaning significantly impact model performance?  
**A2:** Yes, preprocessing has shown improvement in macro precision, especially for casual language inputs.