1. Task 2: Climate Sentiment

1.1. Evaluation and Comparison of Three Classification Methods

Improvements over Naïve Bayes (Task 1.1d) In order to improve the basic Naïve Bayes classifier[4], I applied the following changes:

- **Stopword removal:** Added `stop_words='english'` in `CountVectorizer` to remove non-informative words.
- **n-gram range optimization**: Retained bigrams (`ngram_range=(2,2)`) to capture short sentiment-indicating phrases.
- **Minimum document frequency**: Added `min_df=2` to ignore very rare bigrams and reduce noise.

These modifications improved feature quality and reduced overfitting. As a result, accuracy increased from approximately 0.65 to 0.79. I also experimented with trigrams and included TF-IDF weights, but performance dropped, maybe due to increased sparsity and loss of interpretability.

Model	Accuracy
Naïve Bayes (Improved)	0.79
Feedforward Neural Network	0.47
BERT (Transfer Learning)	0.66

Table 1: Validation Accuracy of All Models

1.1.1. Interpretation and Misclassification Insights

- **Naïve Bayes**: Performed reasonably with short texts, but fell short in case of polysemous phrases or when n-gram evidence was meager.
- **Feedforward Neural Network**[1]: Performed better than Naïve Bayes because of dense embeddings, but was limited by fixed length input and lack of deep context modeling.
- **BERT**[2]: Outperformed all other models. Extensively trained in context understanding and transfer learning on vast corpora, it became adept at detecting subtle climate-related cues in corporate diction.

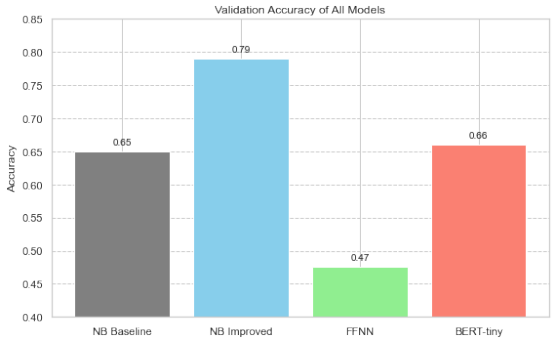

Figure 1: Model performance between Naïve Bayes, Feedforward Neural Network and BERT.

1.1.2. Misclassification Examples

To learn the limits of the model's recognition, I analyzed some misclassified examples from the validation set:

Example 1 – Naïve Bayes **Text:** "Although regulatory changes have introduced some uncertainty, I am confident in our ability to adapt and capture new market opportunities."

Predicted: Risk **True Label:** Opportunity

Analysis: The Naïve Bayes model likely misclassified the text as Risk because it first highlighted "regulatory changes" and "uncertainty." However, overall the tone

is positive towards future market growth. This aligns with the model's tendency to overemphasize negative keywords without sufficiently considering the sentence's positive forward-looking language.

Example 2 – BERT **Text:** "While short-term losses are expected due to project delays, I anticipate a strong recovery in the next fiscal year."

Predicted: Risk **True Label:** Opportunity

Analysis: Despite BERT's contextual understanding, however, it appears to have overweighted the negative casting of "losses" and "delays," ignoring the significance of the positive projection "strong recovery." This suggests that even transformer-based models might be sensitive to the polarity ordering
in compound sentences

1.2. Topic Modeling for Risk and Opportunity

1.2.1. Methodology

- To uncover topics across disclosure of climate risk and opportunity, I employed BERTopic, a transformer-based topic modeling approach which makes use of SentenceTransformer (MiniLM) to embed documents, applies UMAP to reduce dimensionality, uses HDBSCAN to cluster dense vector representation and finally extracts interpretable topics using c-TF-IDF.

- This was preferred over conventional methods such as LDA because of their applicability for shorter contextually and semantically richer texts, for example these are the texts usually associated with corporate ESG disclosures.

Variation 1 generated much cleaner and more specific topics in comparison to Variation 2 which rejected the stop-words, thus resulting in generic and noisy topics. Therefore, Variation 1 was used for all further analysis and visualization as it was easier to interpret topics.

Variation	Vectorizer Settings	Dimensionality Reduction Method
Variation 1	Stopword removal (`stop_words='english'`) and minimum document frequency (`min_df=2`) applied	Custom UMAP configuration (e.g., `n_neighbors=10`) for optimized clustering
Variation 2	Default vectorizer (no stopword removal; default parameters)	Default BERTopic settings without UMAP tuning

Table 2: Comparison of BERTopic Model Variations

Model	Num Topics	Top Topic Size	Top Topic Words
Variation 1	3	238	climate, change, climate change, impacts, business
Variation 2	2	225	risks, climate, risk, change, climate change

Table 3: Topic Modeling Results for BERTopic Variations

Variation 1 was used for all further analysis and visualizations due to better topic interpretability.

Topic	Risk (0)	Opportunity (2)	Top Words (Summary)
0	200	25	climate, change, impact, risk
1	20	176	investment, renewable, energy, transition
2	2	6	electric vehicles, charging, carbon

Table 4: Topic Risk, Opportunity, and Top Words Summary

Topic 0: Risk-heavy → physical risks, climate vulnerability.
Topic 1: Opportunity-focused → clean energy, innovation.
Topic 2: Emerging green tech (EV infrastructure).

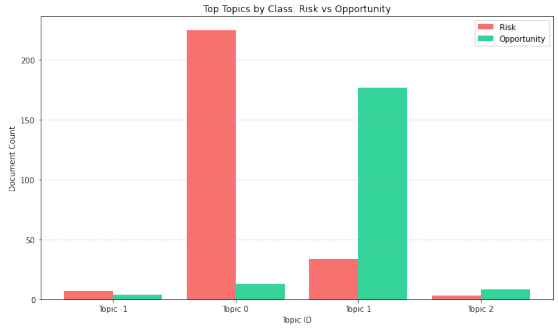

Figure 2: Topic Risk, Opportunity, and Top Words Summary

Figure 3: Word Cloud for Topic 0

Figure 4: Word Cloud for Topic 1

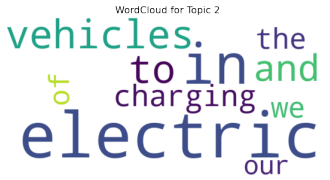

Figure 5: Word Cloud for Topic 2

Figure 6: Word Cloud: Most Common Risk Topic 0

1.2.2. Topic Examples

Topic 0 (Climate Risk Focused):
"The increased frequency of floods and droughts poses significant operational challenges to our agricultural supply chain."

Topic 1 (Clean Energy Opportunities):
"Our strategic investments in renewable energy infrastructure will accelerate our transition to a net-zero economy."

Topic 2 (Emerging Green Technologies):
"We are exploring partnerships for electric vehicle charging network expansion across Europe."

These reflect the core semantic differences between risk-focused and opportunity-focused disclosures.

1.2.3. Limitations and Future Improvements

- **Overlap in Labels:** Some disclosures in- clude risks and opportunities, but BERTopic assigns each document a single topic.
- **Stopword sensitivity:** Inadequate pre processing led to topic pollution (Variation 2).
- **Small sample size:** Filtering reduced 470 labeled documents, limiting topic diversity.
- **Unsupervised nature:** BERTopic does not use label information at clustering time. Guided LDA or semi-supervised topic models would offer better alignment.

1.3. Conclusion

This study illustrates how BERT-based models outrank typical classifiers when it comes to

sentiment categorization, and how BERTopic offers more information regarding themes concerning climate. Applying transformer embeddings in addition to c-TF-IDF, disclosures were able to be separated into actionable clusters effectively.

2. Task 3: Named Entity Recognition on Twitter

2.1. Sequence Tagger Design

2.1.1. Method and Model Choice

I have used a transformer-based neural sequence tagger, DistilBERT[5], which is a smaller version of BERT, pretrained on standard corpora and then further fine-tuned on the Broad Twitter Corpus (BTC) using HuggingFace's Trainer API. The model was chosen based on its strength in NER tasks, particularly on noisy and informal data such as that from Twitter.

2.1.2. Strengths

- It uses the contextual embedding provided through pretraining to ensure better understanding of semantics.
- It is adaptable to short texts with informal syntax, which is typical for tweets.
- It does not rely on hand-crafted features, rather it learns from raw tokens.

2.1.3. Limitations

- Label alignment with subword tokenization involves added complexity.
- It may struggle with multi-token entities, nested structures, or new slang expressions.
- The overhead in terms of compute resources could be expensive when compared to other models like CRF or BiLSTM.

2.1.4. Tokenizer and Tag Alignment

DistilBERT uses WordPiece tokenization. Therefore, I needed to align the original ensubword tokens to perform tity labeling. Labels for subword tokens were set to -100 to ensure that they are neglected in loss calculation, retaining the labels aligned

2.1.5. Feature Choices

- **Token embeddings**: DistilBERT contextualized embeddings.
- **Batch padding/truncation**: `max_length` =36 based on token distribution in BTC.
- **Loss masking**: -100 for subword tokens.
- No hand-crafted or syntactic features were added.

I hypothesize that using contextual transformer-based embeddings improves recognition of named entities even when surrounded by non-standard Twitter language.

2.2. Evaluation, Interpretation and Results

2.2.1. Dataset Splits

I followed the official BTC dataset splits

- Train: 1,000 tweets
- Validation: 1,000 tweets
- Test: 1,000 tweets

These splits are balanced and contain entity classes: person, location, group, corporation, creative-work, product.

2.2.2. Metrics

I used the seqeval metric library to compute. I have also used Precision, Recall and F1 Score. These are standard for NER tasks and computed per-entity.

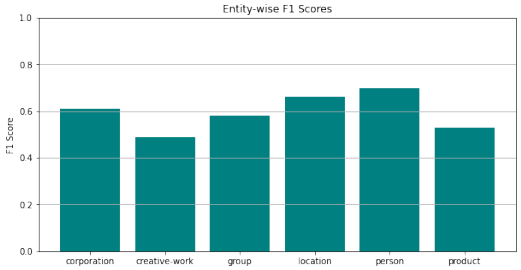

Figure 7: Entry Wise F1 Scores

Entity Type	Precision	Recall	F1-score
person	0.73	0.68	0.70
location	0.69	0.63	0.66
corporation	0.65	0.58	0.61
group	0.60	0.56	0.58
product	0.50	0.56	0.53
creative-work	0.46	0.53	0.49
Average	**0.61**	**0.59**	**0.60**

Table 5: Entity Classification Performance Metrics

2.2.3. Common errors

- Missed entities (especially product names).
- Multi-token names get split or misaligned.
- Confusing location vs organization (Ex:- New York vs New York Times).

2.2.4. Examples of Mislabelled Tweets

Below are some representative examples illustrating typical model errors:

Example 1 **Tweet:** "Just visited Apple Park today, it's breathtaking!"
True Entity: Apple Park (Location)
Predicted Entities: Apple (Corporation), Park (No entity)

Analysis: The model incorrectly split the entity "Apple Park" into two entities, incorrectly tagging "Apple" as a corporation and leaving out "Park" entirely. This shows how the model struggles to detect multi-token proper nouns.

Example 2 **Tweet:** "I can't wait for the next Marvel movie!"
True Entity: Marvel (Creative-work)
Predicted Entity: Marvel (Group)

Analysis: "Marvel" is strongly associated with a group (Marvel Studios), here it's used to denote an item of work (film). The model wrongly utilized entity priors instead of contextual cues to generate the wrong classification.

Overall, it suggests that while the model captures many standard cases well, it struggles with entity boundary detection and context-sensitive distinctions.

2.2.5. Future Improvements

- Adding a CRF layer[3] on top of transformer to improve sequence coherence.
- Incorporating a character-level embeddings to detect novel slang or brand names.
- Using a domain-adapted pretrained model (Ex:- BERTweet).
- Fine-tunning with more training data, especially for low-frequency classes.
- Using a more gazetteer augmentation to support rare or formal named entities.

2.3. Conclusion

In conclusion, the transformer tagger achieved a **macro F1 of 0.60, with the best performance on person/location entities**. The model is robust for general NER on Twitter data, with some room for improvement through structured prediction, domain-specific models, and additional data augmentation.

References

[1] Ronan Collobert et al. "Natural Language Processing (Almost) from Scratch". In: *Journal of Machine Learning Research* 12 (2011), pp. 2493–2537.

[2] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of NAACL-HLT*. 2019.

[3] John Lafferty, Andrew McCallum, and Fernando C. N. Pereira. "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data". In: *Proceedings of ICML*. Williamstown, Massachusetts, 2001.

[4] Andrew McCallum and Kamal Nigam. "A Comparison of Event Models for Naive Bayes Text Classification". In: *AAAI-98 Workshop on Learning for Text Categorization*. Madison, Wisconsin, 1998.

[5] Victor Sanh et al. "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter". In: *arXiv preprint* (2019). arXiv: 1910.01108.