Text Analytics Summative Coursework

1. Task 2.1 - Climate Sentiment Classification

1.1. Naïve Bayes Modifications

To improve classification performance, the CountVectorizer was modified to include both unigrams and bigrams and to exclude rare features using a minimum document frequency of two. These adjustments increased the validation accuracy from 78.0% to 79.0%. Other preprocessing options, such as stopword removal and lowercasing, were tested but resulted in lower accuracy and were therefore discarded. Including unigrams helped capture short contextual patterns, while the frequency filter reduced noise from outlier tokens. This configuration offered a simple yet effective improvement without increasing model complexity.

1.2. Classifier Comparison and Discussion

1.2.1. Performance Summary

Three classifiers were evaluated for climate sentiment classification: a Naïve Bayes model with count-based n-gram features, a feedforward neural network, and a fine-tuned transformer model (BERT-tiny). Table 1 presents their validation accuracies.

Table 1: Validation accuracy of each classifier.

Model	Accuracy (%)
Naïve Bayes	79.0
Neural Network	52.5
BERT-tiny (fine-tuned)	53.0

Naïve Bayes achieved the highest accuracy, outperforming both deep learning models. This highlights the effectiveness of sparse features and careful preprocessing in low-resource settings.

1.2.2. Discussion and Model Comparison

Naïve Bayes The Naïve Bayes classifier used a unigram and bigram count vectoriser with a document frequency threshold to filter rare terms and reduce noise. Despite its assumption of feature independence, it proved effective in this setting (Simpson, 2024c). (The sentiment labels are defined as risk (0), neutral (1), and opportunity (2).)

Misclassifications showed reliance on surface-level patterns, such as labelling a neutral supply chain disclosure as an opportunity due to the presence of terms like “countermeasures.” These errors reflect the model’s limited semantic understanding. Future improvements could involve stopword removal or hybridising with word embeddings for richer representations.

Neural Network The feedforward neural network employed an embedding layer followed by a dense layer, but its simplicity limited the capacity to model complex semantics. Neural networks can capture non-linear relationships (Simpson, 2024a), but require greater depth and richer embeddings.

Misclassified examples showed confusion between sentiment-bearing and neutral texts, such as labelling a green mortgage scheme promoting sustainable housing as neutral. These errors suggest limited contextual awareness and a bias toward dominant classes. Improvements may include pretrained embeddings or deeper sequence-based models (Simpson, 2024b).

Transformer (BERT-tiny) BERT-tiny was fine-tuned on 800 examples using the Trainer API with 128-token truncation. The model’s compact size and limited data constrained performance.

It often predicted neutral labels for clearly polarised content, such as misclassifying a disclosure focused on sustainability-driven R&D as neutral. While fine-tuning enables transfer learning (Simpson, 2025a), its effectiveness

declines with small datasets and compact architectures (Simpson, 2025b).

2. Task 2.2 – Topic Modelling of Climate Risks and Opportunities

2.1. Method and Preprocessing

This task extended the analysis from Task 2.1 by applying unsupervised topic modelling to the same climate sentiment dataset, focusing on disclosures labelled as either *risk* or *opportunity*. Latent Dirichlet Allocation (LDA) was selected for its ability to discover interpretable topic structures in large text corpora using bag-of-words input (Simpson, 2024d).

To reduce ambiguity, neutral samples were excluded. All models were trained on the training split created in Task 2.1, using the same fixed seed based on the student number to ensure reproducibility.

Text preprocessing was tailored to improve topic coherence. This included lowercasing and tokenisation. Stopwords, punctuation, short words (fewer than three characters), and numeric tokens were then removed to reduce noise and retain meaningful content. These steps aimed to retain only semantically informative content while reducing noise. Table 2 summarises the pipeline.

Table 2: Summary of preprocessing steps.

Step	Purpose
Lowercasing	Standardise case
Tokenisation	Split text into words
Stopword removal	Eliminate frequent, uninformative terms
Punctuation removal	Discard non-lexical characters
Short word removal	Remove filler tokens under 3 characters
Numeric filtering	Exclude standalone numbers

2.2. Model Variations

To explore how input representation affects topic quality, two LDA models were trained on the risk-labelled training data using different vectorisation strategies. The first used a bag-of-words matrix from *CountVectorizer*, which encodes raw token frequency and aligns with LDA's assumption that word counts follow a Dirichlet distribution (Simpson, 2024d).

The second used *TfidfVectorizer*, which penalises common terms across documents. While this violates LDA's generative assumptions, it can suppress generic terms and surface more distinctive vocabulary (Sievert and Shirley, 2014). Both models used identical preprocessing, five topics, and the same random seed for fair comparison.

2.3. Evaluation and Visualisation

To assess topic quality across the two LDA models, both quantitative and qualitative evaluations were conducted.

Coherence scores were used to measure the semantic similarity of top words within each topic. The model using CountVectorizer achieved a score of 0.4347, while the TF-IDF-based model scored slightly higher at 0.4445. Although this difference was modest, it suggests that TF-IDF reweighting helped enhance topic cohesion by reducing the influence of overly frequent terms. Coherence was measured using the c-v metric, which has been shown to correlate well with human topic interpretability (Röder et al., 2015).

A qualitative inspection of topic interpretability revealed clearer thematic distinctions in the TF-IDF model. The CountVectorizer output featured broader and overlapping topics, often dominated by generic terms such as *climate*, *change*, and *risk*. In contrast, the TF-IDF model surfaced more specific and domain-relevant vocabulary, including *mortgage*, *divested*, and *eni*, indicating sharper topic boundaries.

To illustrate this contrast, Topic 3 from each model was selected for visual comparison. Figure 1 shows the top terms in the CountVectorizer model, while Figure 2 displays the corresponding topic from the TF-IDF model. While both focused on fossil fuel exposure, the TF-IDF version additionally captured a divestment-related narrative that was not as evident in the count-based model.

These findings support the use of TF-IDF input in LDA for small corpora, where common terms may dilute thematic clarity.

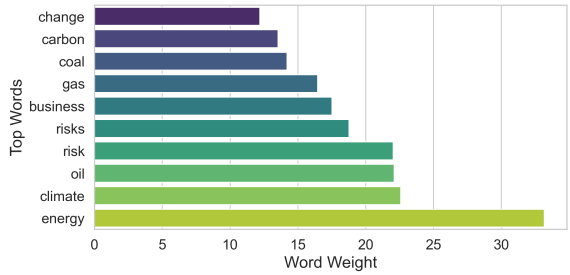

Figure 1: Top words in Topic 3 (Count-based LDA)

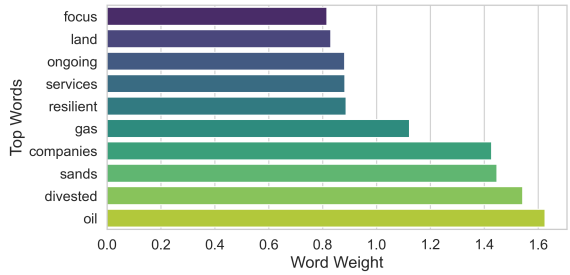

Figure 2: Top words in Topic 3 (TF-IDF-based LDA)

2.4. Opportunity Topics

To address both sentiment classes in the dataset, a third LDA model was trained on the opportunity-labelled documents. This model used the TF-IDF vectorisation strategy and maintained the same number of topics (five) to ensure comparability with the risk-based experiments. The TF-IDF configuration was chosen based on its slightly higher coherence score and enhanced interpretability observed in the earlier comparison.

The aim of this model was to extract themes associated with climate-related opportunities in corporate disclosures. The topics uncovered reflected a broad range of forward-looking themes, including green investment, renewable energy, sustainability initiatives, and corporate innovation.

For example, Topic 1 (Figure 3) highlighted large-scale investment in clean technologies, featuring terms such as *billion*, *renewable*, and *sustainable*. Topic 2 focused on solar energy and energy efficiency, while Topic 3 revealed green finance and infrastructure development through terms like *bond*, *goals*, and *infrastructure*. Topic 5 included company names such as *Hyundai* and *Eni*, suggesting organisation-specific innovation strategies.

Compared to the risk-based models, the opportunity topics were more positive and forward-looking, often focused on new technologies, investment, and sustainability efforts. These results support that filtering by sentiment helped distinguish between themes of risk and those related to progress.

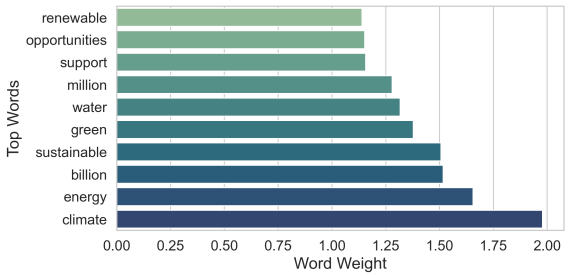

Figure 3: Top terms for Topic 1 (TF-IDF, Opportunity documents)

2.5. Interpretation and Limitations

This task demonstrated how unsupervised topic modelling can reveal key narratives in climate-related corporate disclosures. The comparison between input representations showed that using TF-IDF in LDA led to more distinct and interpretable topics, particularly by reducing the influence of high-frequency terms. Risk-related disclosures tended to cluster around themes of uncertainty, fossil fuel exposure, and physical impacts. In contrast, opportunity texts highlighted investment, innovation, and sustainability strategies.

Despite these insights, several limitations should be noted. The bag-of-words approach used in LDA discards word order and contextual meaning, which can blur nuanced differences between topics. Manual interpretation introduces subjectivity, and some term overlap (e.g., “climate”, “energy”) appeared across both sentiment classes. Moreover, short document length may limit topic separation in small corpora.

Future improvements could involve contextual topic models, such as BERTopic, which leverage transformer-based embeddings to capture semantic similarity more effectively. Guided LDA with predefined topic lists could also help improve topic discovery in specific domains.

3. Task 3: Named Entity Recognition

This task focused on extracting named entities from Twitter posts using a transformer-based model. Tweets pose unique challenges for Named Entity Recognition (NER) due to informal language, abbreviations, and limited context.

3.1. Dataset and Preprocessing

3.1.1. Dataset Description

The Broad Twitter Corpus (BTC) was used for training and evaluation. It contains pre-tokenised tweets annotated using the BIO tagging scheme, which labels tokens as the beginning (B), inside (I), or outside (O) of an entity span. For example, in the sentence *Shaw's only touch was his goal*, `Shaw` is tagged `B-PER`, while other tokens are labelled `O`. The dataset includes three entity types: person (PER), organisation (ORG), and location (LOC). It is pre-split into training, validation, and test sets for reproducibility.

3.1.2. Preprocessing

No additional text cleaning or normalisation was required. The dataset was already tokenised at the word level and formatted for sequence labelling. Preprocessing focused on converting each sequence into subword token inputs using a BERT-compatible tokenizer, while preserving alignment with the original annotations.

3.2. Model Setup

3.2.1. Model Architecture

The model was based on a pretrained `bert-base-cased` transformer configured for token-level classification. A linear classification head was added on top of the final hidden layer to predict one of the BIO-encoded entity labels (PER, ORG, LOC) for each token.

The cased version of BERT was chosen to preserve capitalisation, which is important for identifying named entities in social media text. Unlike uncased models, it retains cues such as proper nouns and abbreviations, which are often informative in noisy input.

As a pretrained transformer, BERT provides contextual embeddings based on both left and right context, helping resolve ambiguity and capture long-range dependencies. This architecture has shown strong performance on informal text domains (Yoon et al., 2020). Compared to traditional sequence tagging approaches such as Hidden Markov Models or Conditional Random Fields, BERT eliminates the need for handcrafted features and better captures contextual cues in noisy settings like Twitter (Lample et al., 2016).

3.2.2. Label and Token Handling

Entity labels were mapped to numerical class indices for training and converted back to string labels during evaluation using a consistent ID-to-label mapping.

Since BERT uses subword tokenisation, label alignment was necessary to match word-level annotations. Each original entity tag was assigned to the first subword token of a word, while all subsequent subwords and special tokens were assigned `-100` to exclude them from loss computation. This ensured consistent and accurate supervision across all dataset splits.

3.3. Training Procedure

3.3.1. Training Configuration

The model was fine-tuned for token classification over three epochs using the AdamW optimiser. Dynamic padding ensured consistent input shapes across batches without unnecessary padding. A fixed seed was used to ensure reproducibility. Training and evaluation were conducted at the end of each epoch, with the best-performing checkpoint selected based on validation F1-score. Key hyperparameters are summarised in Table 3.

Table 3: Training configuration summary.

Hyperparameter	Value
Max sequence length	128
Epochs	3
Learning rate	2e-5
Weight decay	0.01
Train batch size	16
Eval batch size	16
Evaluation strategy	End of each epoch
Optimiser	AdamW
Metric for best model	F1-score

3.3.2. Evaluation Strategy During Training

Evaluation used span-level F1-score as the primary metric, balancing precision and recall. Predictions were decoded from model logits and compared to gold labels. The best checkpoint, based on validation F1, was automatically restored after training. The final model and tokenizer were saved locally for downstream evaluation.

3.4. Evaluation Metrics

The fine-tuned model was evaluated on the test split of the Broad Twitter Corpus using two complementary strategies. Token-level evaluation compared labels independently at the token level, treating each tag prediction in isolation. This approach can be misleading for named entity recognition, as it rewards partial matches and does not enforce valid entity spans.

Entity-level evaluation, in contrast, applied strict span matching under the `IOB2` scheme (Tjong Kim Sang and De Meulder, 2003). This required both the entity boundaries and the predicted type to match the ground truth. Entity-level precision, recall, and F1-score were calculated per class and averaged to provide a more rigorous assessment of end-to-end entity extraction performance.

3.5. Results

3.5.1. Token-Level Results

Token-level performance is summarised in Table 4. These metrics reflect general tagging accuracy but do not account for span validity or type consistency.

Table 4: Token-level performance on the test set.

Metric	Score (%)
Precision	77.00
Recall	77.06
F1-score	77.03
Accuracy	95.50

3.5.2. Entity-Level Results

Entity-level results are shown in Table 5. These provide a stricter and more task-relevant view of model performance by requiring full entity-level matches.

Table 5: Entity-level performance on the test set.

Entity	Prec. (%)	Rec. (%)	F1 (%)
PER	89.0	88.4	88.7
LOC	72.2	62.6	67.1
ORG	58.0	56.9	57.4
OVR	79.0	76.8	77.9

Performance was highest for person (PER) entities, which were typically well-formed and unambiguous. Location (LOC) entities performed moderately, while organisation (ORG) entities were the most difficult to detect, likely due to ambiguous names and shorter surface forms. These results highlight the model's strengths on well-structured entities and its limitations in noisier or more context-dependent cases.

3.5.3. Error Analysis

Manual inspection of predictions revealed several recurring error types. Span boundary errors were common, with entity-initial tokens misclassified as `I-` instead of `B-`, leading to invalid spans. Some valid entities, particularly short or rare organisation names, were missed entirely and tagged as non-entities. Type confusion often occurred between organisations and locations due to overlapping surface forms and ambiguous contexts. False positives were also observed, with capitalised non-entity tokens such as roles or initials incorrectly tagged as named entities.

These errors highlight the challenges of NER on Twitter, where informal syntax, abbreviations, and compressed expressions limit the contextual cues needed for span detection. Tweets often lack standard punctuation or grammar, making it harder to infer boundaries from context.

To address these issues, future work could explore domain-adapted models like BERTweet, which are pretrained on social media text. Adding a Conditional Random Field (CRF) layer may improve tag consistency by modelling label transitions. Post-processing heuristics or gazetteers could also help refine span boundaries and reduce systematic errors.

4. References

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of NAACL-HLT*, pages 260–270.

Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the space of topic coherence measures. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining (WSDM)*, pages 399–408. ACM.

Carson Sievert and Kenneth Shirley. 2014. Ldavis: A method for visualizing and interpreting topics. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*, pages 63–70.

Edwin Simpson. 2024a. Deep learning for text. Lecture notes, University of Bristol.

Edwin Simpson. 2024b. Deep text classifiers. Lecture notes, University of Bristol.

Edwin Simpson. 2024c. Text classification with naive bayes. Lecture notes, University of Bristol.

Edwin Simpson. 2024d. Topic modelling with lda. Lecture notes, University of Bristol.

Edwin Simpson. 2025a. Finetuning transformer models. Lecture slides, University of Bristol.

Edwin Simpson. 2025b. Transfer learning concepts. Lecture slides, University of Bristol.

Erik F Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.

Seunghyun Yoon, Jinhyuk Kim, Jungin Jung, and Jaewoo Lee. 2020. Pre-trained language model for named entity recognition: Bert and beyond. In *Proceedings of the 28th International Conference on Computational Linguistics*.