TEXT MINING: NATURAL LANGUAGE PROCESSING

SENTIMENT

This project performed sentence-level sentiment analysis on a test set, comparing three approaches: VADER, Naive Bayes (NB) trained on airline tweets, and a pre-trained RoBERTa Transformer. The objective was to evaluate their performance using accuracy and F1-score.

Data Description

The test set was not explicitly detailed but is assumed to be a small, general-domain sentiment dataset provided for the assignment. VADER relies on its internal lexicon, requiring no training data. NB was trained on a dataset of fewer than 5,000 airline tweets, an unrelated domain, using TF-IDF features. RoBERTa, pretrained on vast general-domain corpora and fine-tuned on Twitter data, leverages millions of examples for robust sentiment understanding. The limited size and domain mismatch of NB's training data, combined with the unspecified test set size, likely influenced performance outcomes.

Approach and Motivation

Three systems were evaluated. VADER, a rule-based model (Hutto & Gilbert, 2014), uses predefined lexical scores, chosen for its speed and simplicity as a baseline, though it lacks flexibility. NB, a probabilistic classifier, was trained with TF-IDF features on airline tweets, selected as a traditional machine learning approach, despite its domain-specific constraints. RoBERTa, a Transformer model (Liu et al., 2019), was chosen for its extensive pretraining and ability to capture context-dependent nuances, reflecting advanced NLP techniques from lecture discussions. Preprocessing for NB involved TF-IDF vectorization, while VADER and RoBERTa used raw text inputs, with RoBERTa tokenized via its pretrained tokenizer.

Results and Analysis

Quantitative: VADER achieved the lowest accuracy (33%), followed by NB (39%), while RoBERTa excelled with 72% accuracy and a macro F1 of 0.71. Reference to the table for classification reports, confusion matrices, and comparison plots of accuracy, precision, recall, and F1-score (weighted and macro), which consistently showed RoBERTa's superiority, aligning with Transformer models' strengths in semantic context (Liu et al., 2019). Qualitative: VADER struggled with nuance, misclassifying words like "electric" (e.g., "The vibe was electric" as neutral instead of positive). NB's poor performance likely stems from its small, domain-specific training data (<5,000 airline tweets), limiting generalization. RoBERTa, though superior, misclassified ambiguous neutral sentences (e.g., "It's fine" as positive), indicating difficulty with subtlety.

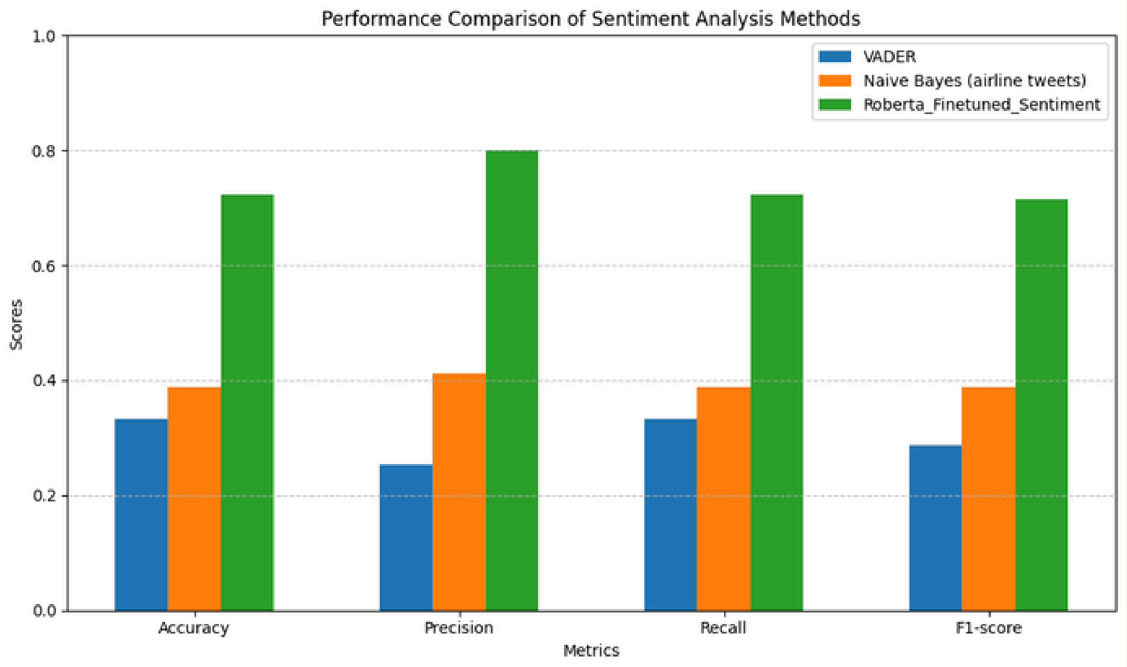
Comparative Discussion

RoBERTa (72%) significantly outperformed VADER (33%) and NB (39%), leveraging its contextual understanding, though VADER and NB showed resilience in simplicity and domain-specific scenarios, respectively. RoBERTa's advantage reflects its pretraining scale, while NB's domain mismatch and VADER's rigidity explain their lower scores.

Conclusions and Limitations

RoBERTa excels in sentiment analysis due to its extensive pretraining, while VADER's rule-based approach and NB's limited, mismatched training data hinder performance. Limitations include the small, unspecified test set, NB's constrained training data, and RoBERTa's errors on neutral or sarcastic text. Improvements could involve fine-tuning RoBERTa with sarcasm detection modules, expanding NB's training data across domains, or testing on a larger, diverse dataset for robustness.

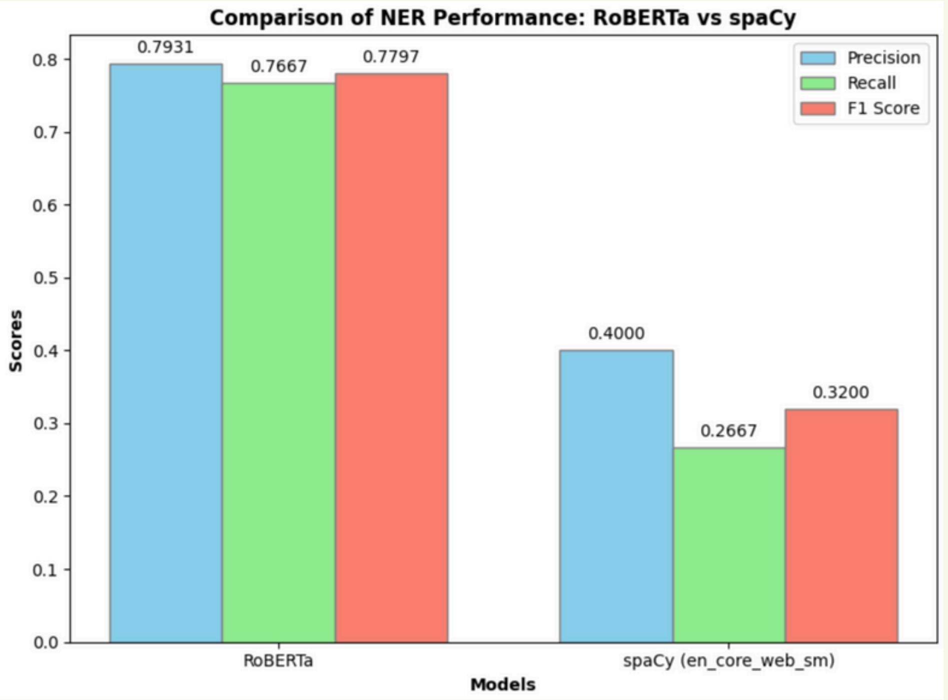
SENTIMENT



Ranking

VADER (33%) -> NB (39%) -> RoBERTa (72%)

NERC



spaCy:
F1 = 0.32
ORG = 0.57,
PERSON = 0.37,
LOC/*WORK_OF_ART*<0.30

RoBERTa:
F1 = 0.7797
PERSON = 0.88,
WORK_OF_ART = 0.88,
LOC = 0.67,
ORG =0 .00

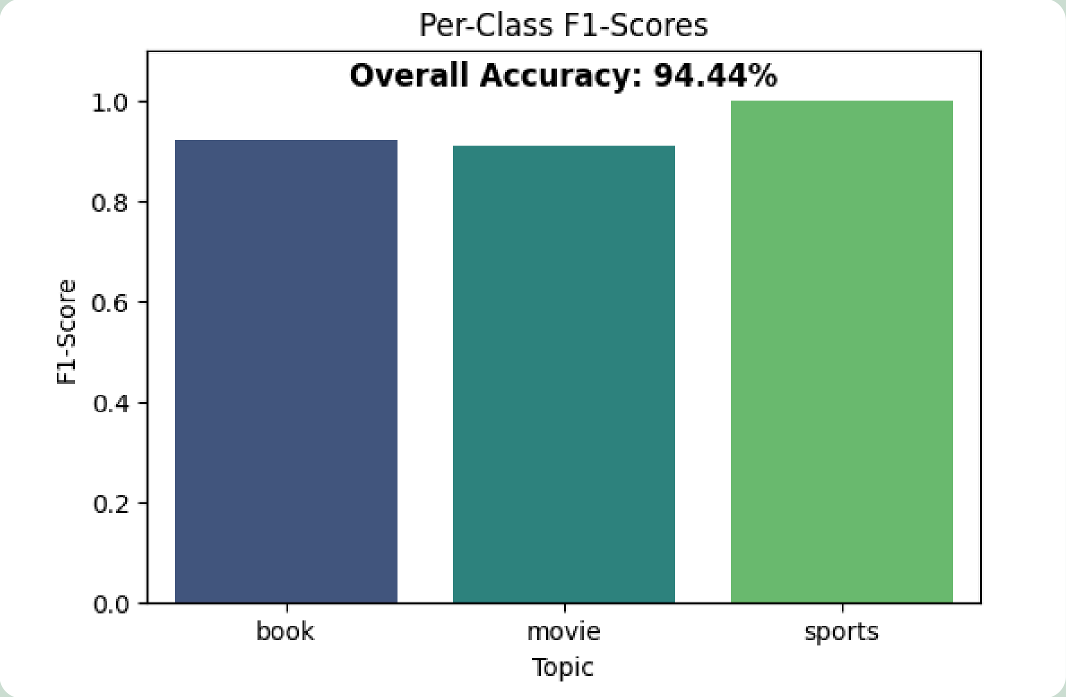
TASK DIVISION

Nasreddine was responsible for the sentiment analysis, adding his results to the poster and writing the task division. Innocent and Selma worked on Named Entity Recognition (NERC) and added their findings to the poster. Tunahan handled the topic classification and added his results to the poster. Each team member wrote the code for their respective tasks.

BIBLIOGRAPHY

- Honnibal, M., Montani, I., Van Landeghem, S., & Boyd, A. (2020). spaCy: Industrial-strength natural language processing in Python. Zenodo. https://doi.org/10.5281/zenodo.1212303
- Hutto, C. J., & Gilbert, E. (2014). VADER: A parsimonious rule-based model for sentiment analysis of social media text. Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media (ICWSM-14), 216–225.
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30, 5998–6008.

TOPIC ANALYSIS



Sports: The model achieved a perfect F1-score of 1.00 for sports

Book: With an F1-score of 0.92, the performance on book sentences is very strong

Movie: The F1-score of 0.91 for movies is also really high.

NERC

This project compares two NER systems, paCy (en_core_web_sm) and RoBERTa (RoBERTa-base), to classify entities using the WNUT 17 dataset (3,398 Twitter sentences, 1,914 entities, noisy text) for training and a custom test set (15 sentences, 30 entities: 11 PERSON, 3 ORG, 7 LOC, 9 WORK_OF_ART, sports/literature focus).

Approach and Motivation

- **spaCy** (en_core_web_sm): A lightweight CNN-based model (Honnibal et al., 2020) chosen as a baseline to compare classical sequence labeling with transformer-based models. It was preprocessed by converting WNUT 17 (CoNLL format) into spaCy-compatible format, transforming BIO tags into entity span annotations (text, {"entities": [...]}) tuples, and fine-tuned with 50 iterations, dynamic batch size (compounding (4.0, 32.0, 1.001)), dropout=0.2, and SGD optimizer with adaptive learning rate. Its transition-based NER system relies on local token context and embeddings, making it fast but less effective on noisy text due to limited contextual understanding.
- **RoBERTa** (RoBERTa-base): A transformer-based model (Liu et al., 2019; Vaswani et al., 2017) chosen for its enhanced pretraining and ability to capture contextual nuances in noisy text. Preprocessing involved converting WNUT 17 tags to BIO format, tokenizing with RobertaTokenizerFast (max_length=64), and aligning subword labels. Training was performed with a learning rate of 2e-5, batch size of 64, 10 epochs, early stopping, and FP16 precision, all optimized to leverage Google Colab's computational power.

Results and Analysis

- **Quantitative**: Reference to the table on the left.
- **Qualitative**: spaCy missed or misclassified multi-token entities like "Cristiano Ronaldo" (PERSON) and "London" (LOC), struggling with noisy text and inconsistent WORK_OF_ART handling due to limited context. RoBERTa missed "Manchester United" (ORG → O) and misclassified "Coldplay" (ORG → PERSON) and "Inter Miami" (ORG → LOC), showing strength in PERSON and WORK_OF_ART but poor ORG recognition, likely due to WNUT 17 data imbalance.

Comparative Discussion

RoBERTa (F1=0.7797) outperformed spaCy (F1=0.32), leveraging superior contextual understanding, though spaCy scored higher on ORG (0.57 vs. 0.00), possibly due to variance and small test size.

Conclusions and Limitations

RoBERTa excels on noisy text, while spaCy struggles yet shows ORG potential. spaCy's en_core_web_sm CNN-based pipeline lacks contextual depth and wasn't fine-tuned for noisy text like WNUT 17; it could improve with en_core_web_trf or domain-specific training. RoBERTa's performance on ORG was limited by scarce examples in WNUT 17, small test size, and lack of domain adaptation; enhancements could include OntoNotes 5.0, fine-tuning, or adjusting learning rate (e.g., 3e-5).

TOPIC ANALYSIS

The goal was to assign predefined topics (book, movie, sports) to individual sentences from a small test dataset, evaluating performance at the sentence level using accuracy, F1-score, and qualitative analysis.

Data Description

The test data comes from sentiment-topic-test.xls, containing 18 sentences with gold-standard topic labels (provided for the assignment). Fine-tuning data was derived by splitting this small test dataset, though the exact split is unspecified (e.g., 80% train, 20% validation assumed). RoBERTa's pretraining data includes massive corpora (~160GB) such as BookCorpus, English Wikipedia, CC-News, and OpenWebText, ensuring robust language understanding (Liu et al., 2019). With only 18 sentences, the test set is extremely small, limiting insights into generalization.

Approach and Motivation

We fine-tuned RoBERTa, a transformer-based model (Vaswani et al., 2017), chosen for its superior pretraining and ability to capture contextual nuances in noisy text (Liu et al., 2019). Preprocessing involved tokenization with Hugging Face's tokenizer and padding/truncation to a maximum length of 256 tokens. Parameters included a learning rate of 2e-5, batch size of 8, and 30 epochs, selected for convergence on the small dataset. RoBERTa's contextual embeddings were used as features to effectively capture sentence-level semantics.

Results and Analysis

Quantitatively, the model achieved an accuracy of 94.44%, with F1-scores of 0.92 for book, 0.91 for movie, and 1.00 for sports. This high performance highlights RoBERTa's strength, though the tiny dataset (18 sentences) suggests overfitting. Qualitatively, most sentences were correctly classified, but errors emerged in ambiguous cases, such as overlapping topics (e.g., "a book about sports"). For instance, "This was a thrilling read" (gold: book, predicted: movie) indicates contextual confusion. The primary limitation is the small dataset size, which risks overfitting and hampers generalization to unseen data.

Conclusions and Improvements

The fine-tuned RoBERTa model performs exceptionally on this task, but its near-perfect results are likely inflated by the limited test set. With additional time, we would collect a larger, more diverse dataset for robust evaluation, experiment with data augmentation or cross-validation to reduce overfitting, and test the model on noisier, real-world data (e.g., X posts).