NewsBot Reflection Group Journal NewsBot Intelligence System Group Reflective Journal (2 pages)

Bryan Tzorin

ITAI 2373

BBC News (Kaggle)

8th November 2025

1) Teamwork Reflection (Individual Submission) This is an individual submission, but I treated it like a team submission: Pre-planning: I filled out a little project charter with scope, responsibilities (Data, Modeling, Analysis, Docs), milestones and measures for success (macro-F1 baseline, clean Run-All). File controlling: I used an open-source GitHub repo as version control with atomic commits and descriptive messages as well as a README file noting how to replicate the results from scratch in a new session in Colab. Communication: I've noted where I've commented on certain decisions made along the way in the notebook markdown cells (why TF IDF 1 2 grams, why LinearSVC baseline, why sampling down to 2,000 docs). This would be how teams pass off information between team members (like hand offs). What I'd keep if in a team: daily standups, a Kanban board for visibility and due dates, a necessity of code review (another person had to sign off on big changes). What I'd change: a more definite definition of done per section (what I assumed pre-existing success would have taken had more time devoted). Definitions of done per project completed would have kept expectations in check (inputs, outputs, visualizations, approval checks).

2) Technical Integration Issues Data inconsistencies The BBC data was provided as different csv outputs, I autodetected the text/label columns and standardized them into article/category. Runtime limitations I did use Colab free and not paid for upgrade so was limited to sample size to 2000 and quicker pipelines. I did use caching with intermediate csv and spacy small for time efficiency. Pipeline separation There was a need to ensure that all preprocessing, tf idf, pos/dependency, sentiment, ner and classification included universal text normalization (stripped from url/html, lemmization with stop word filtering, consistent lower case tokens). Metrics and comparisons made I compared logisitic regression, LinearSVC and multinomialnb based on

accuracy and macro-F1 (which favors balance across features) and also included the confusion matrix which demonstrate overlap (tech vs business). How I remedied them Vectorizer parameters (min_df=2, ngram_range=(1,2)); random_state to stabilize findings; drop those which are too short; nan in middle; low for plotting to avoid OOM.

3) Business Value Assessment Use cases: Editorial Routing: new articles can be auto-routed by category and editors can quickly scan top TF IDF terms and entities to understand the nature of the story. Competitor and People Monitoring: Categories that show NER counts assess which ORG/PERSON/GPE are mentioned most; valuable for a beat and PR risk. Sentiment Tracking: Sentiment trends at the category level indicate when something is askew (i.e., Business articles become really negative for some reason). Value to users: Faster insights: desk editors need to sort routing but with little devoting time to hands-on sorting. Explainability: TF IDF key terms and summary paragraphs by POS/Dependency allows for justifying routing and corrections. Alerting: a light rules layer above NER and sentiment can create alerts for named entities or tone thresholds.

4) Individual Contributions I was the sole submitter and therefore, responsible for the following contributions: Module 1 (Context): Wrote the business case and value proposition for newsroom/PR users Module 2 (Preprocessing): Conducted cleaning, tokenization, lemmatization; standardized columns; sampling Module 3 (TF IDF): Tuned features (max_features, n grams) and retrieved top terms per category Module 4 (POS): Created category POS distributions and explained stylistic differences Module 5 (Syntax): Retrieved simple S V O

triples to demonstrate relationship patterns Module 6 (Sentiment): Utilized VADER for tone summaries per category Module 7 (Classification): Trained LogReg, LinearSVC, and MultinomialNB; compared results and reported accuracy, macro F1 and confusion matrix Module 8 (NER): Counted PERSON/ORG/GPE/DATE/MONEY by category and summarized results in a section. Documentation: Created README and annotated notebook for reproducibility in Colab.

5) Future Improvements Data: Supplement with temporal analysis (weekly entity/sentiment trends), deduplicate near duplicate wire stories. Modeling: Adjust calibrated probabilities and thresholding for no response for unclear articles; train with class weight adjustments to mediate imbalance. Features: Add char-level TF IDF for spelling variations; test contextual embeddings (CPU friendly) for better generalizability. NER+: Link entities to knowledgebases (i.e., company tickers) and add simple relationship graphs (ORG PERSON GPE). Ops: Export a small Gradio app for real-time demonstration and batch inference scripts with CSV input/output.

6) Professional Development Contribution This project is reminiscent of an industry pipeline: data ingestion, preprocessing, feature extraction modeling evaluation insights. I applied/learned about: Systems thinking: Maintaining compatibility of outputs across modules and anticipating future review (acceptance) requirements. Reproducibility: One Run All notebook, all seeds deterministic, and environment bootstrapped. Communication: Translating model output into business-oriented insights and tangible next steps.

Link to public repository: Bee3200/Bee-Repository: Class-Midterm-2025-Fall