Final Report: Amharic E-commerce Data Extractor

1. Introduction

This project addresses EthioMart's vision to consolidate real-time e-commerce data from diverse Telegram channels into a single, unified platform. With Telegram's growing prominence in Ethiopian business transactions, the decentralization of independent e-commerce operations presents significant challenges for both vendors and customers. The core objective of this initiative is to develop an Amharic Named Entity Recognition (NER) system capable of extracting critical business entities such as product names, prices, locations, and contact information from unstructured text, images, and documents shared across these channels. The extracted, structured data will populate EthioMart's centralized database, facilitating a seamless customer experience and, crucially, enabling FinTech initiatives like micro-lending assessments for promising vendors.

Our approach spans the complete data science lifecycle: from programmatic data collection and rigorous preprocessing to advanced LLM fine-tuning, model comparison, interpretability analysis, and the development of a practical vendor analytics scorecard. The goal is to transform messy, unstructured Telegram posts into actionable insights, empowering EthioMart to identify high-potential vendors for strategic financial partnerships.

2. Methodology

The project adopted a structured methodology, leveraging a dedicated GitHub repository with modular scripts and notebook-based development to ensure reproducibility and clarity throughout the data science pipeline.

Task 1: Data Ingestion and Data Preprocessing

- **Objective:** To programmatically collect and preprocess raw messages and metadata from selected Ethiopian Telegram e-commerce channels.
- **Data Collection:** Messages, reactions, and images were scraped from five primary Telegram channels: @ZemenExpress, @ethio_brand_collection, @Leyueqa, @Fashiontera, and @marakibrand. Data was saved to data/raw/telegram_data.csv and images to photos/.
- **Preprocessing:** A dedicated ReviewPreprocessor class (src/preprocessor.py) was

developed to clean and normalize the raw text data. Key steps included:
    - Normalizing Amharic character variations.
    - Strictly removing emojis and pictorial symbols.
    - Removing URLs and hashtags.
    - Standardizing currency expressions (e.g., "1500ብር" to "1500 ETB").
    - Retaining Telegram usernames and phone numbers.
    - Removing extra spaces and miscellaneous characters.
    - Cleaned data was saved to data/processed/clean_telegram_data.csv.
- **Exploratory Data Analysis (EDA):** Notebooks (notebooks/data_ingestion_eda.ipynb and notebooks/data_preprocessing_eda.ipynb) were used to analyze raw and cleaned data characteristics.
    - **Insights:** Identified that ~46% of messages had missing text (suggesting a need for OCR on images), 88% contained images, and high-engagement posts had 27+ reactions.

Task 2: Label a Subset of Dataset in CoNLL Format

- **Objective:** To label a portion of the preprocessed dataset in the CoNLL format, a standard for Named Entity Recognition tasks.
- **Process:** A rule-based labeling script (src/data_labeler.py) was implemented to identify and label entities within the Amharic text.
    - **Entity Types:** PRODUCT, PRICE, LOCATION were primary, with optional CONTACT_INFO and DELIVERY_FEE also included using standard B-, I-, L-, U-, O (BILUO) tagging scheme.
    - **Data Size:** A subset of 50 messages was labeled.
    - **Output:** data/labeled/telegram_ner_data_rule_based.conll.
- **Status:** This task was successfully completed, generating the necessary labeled dataset for model fine-tuning.

Task 3: Fine-Tune NER Model

- **Objective:** To fine-tune a pre-trained multilingual transformer model for Amharic NER.
- **Model:** Davlan/afro-xlmr-large was selected as the initial model, known for its multilingual capabilities.
- **Process:** The src/model_finetuner.py script handled the entire fine-tuning pipeline:

    - **Data Loading & Splitting:** The CoNLL data was parsed and split into 80% training, 10% validation, and 10% test sets. Stratification was robustly managed for class distribution.
    - **Tokenization & Alignment:** The tokenizer converted words to subword tokens, and labels were aligned, converting BILUO tags to a model-compatible BIO scheme.
    - **Training:** The model was fine-tuned for 5 epochs using Hugging Face's Trainer API, with a batch size of 8, learning rate of 2e−5, and evaluation at each epoch.
    - **Evaluation:** Precision, Recall, and F1-score were computed on the test set.
- **Output:** The fine-tuned model and tokenizer were saved to models/afro_xlmr_ner_fine_tuned/.

Task 4: Model Comparison & Selection

- **Objective:** To compare different NER models and select the best-performing one based on accuracy and efficiency.
- **Model Compared:** distilbert-base-multilingual-cased was chosen as a lighter-weight alternative to afro-xlmr-large.
- **Process:** A dedicated script (src/distilbert_finetuner.py) replicated the fine-tuning process for DistilBERT using the same dataset and training parameters.
- **Output:** The fine-tuned DistilBERT model and tokenizer were saved to models/distilbert_ner_fine_tuned/.

Task 5: Model Interpretability

- **Objective:** To use model interpretability tools (SHAP and LIME) to explain NER model predictions.
- **Tools:** SHAP (SHapley Additive exPlanations) was implemented in notebooks/model_interpretability.ipynb to show word-level contributions to predictions for specific examples. LIME (Local Interpretable Model-agnostic Explanations) was conceptually discussed due to its more complex adaptation for token-level NER.
- **Status:** The implementation faced a TypeError due to shap.maskers.Text initialization, which was identified and addressed, though full interactive rendering was limited to Jupyter environments. The low overall model performance also impacted the depth of meaningful insights from interpretability.

Task 6: FinTech Vendor Scorecard for Micro-Lending

- **Objective:** To develop a vendor analytics engine that combines NER-extracted entities with Telegram post metadata to generate key performance metrics and a "Lending Score."
- **Script:** src/vendor_scorecard_engine.py.
- **Process:**
  - The script loaded the clean_telegram_data.csv and used the best-performing NER model (DistilBERT) to extract entities from all messages.
  - For each vendor channel, the following metrics were calculated:
    - **Posting Frequency:** Average number of posts per week.
    - **Average Views per Post:** Indicator of market reach and customer engagement.
    - **Top Performing Post:** Identification of the post with the highest view count, along with its extracted product and price.
    - **Average Price Point:** The average price of products listed by the vendor, derived from NER extractions.
  - A simple **Lending Score** was designed as a weighted sum: Score=(Avg Views×0.5)+(Posting Frequency×0.5).
  - The results were presented in a summary table and saved to outputs/vendor_scorecard.csv.
- **Status:** This task was successfully completed, providing a preliminary vendor scorecard.

3. Results and Key Findings

3.1. Model Performance Comparison

The fine-tuning efforts on the small labeled dataset yielded initial performance metrics for both afro-xlmr-large and DistilBERT as follows:

Model Performance Comparison (on Test Set - 10% of 50 labeled sentences):

Metric	afro-xlmr-large	DistilBERT
Eval Loss	2.845	2.960
Precision	0.010	**0.055**
Recall	0.039	**0.132**
F1-Score	0.016	**0.078**
Train Runtime	~48 minutes	**~3.7 minutes**

Summary: DistilBERT significantly outperformed afro-xlmr-large in terms of F1-score (0.078 vs. 0.016) and demonstrated remarkable efficiency with a much faster training time (~3.7 minutes vs. ~48 minutes). Despite this relative improvement, the overall F1-scores for both models remain very low. This indicates that neither model is performing effectively for real-world NER extraction, primarily due to the severely limited size of the labeled training data.

3.2. FinTech Vendor Scorecard Analysis

The vendor scorecard provided preliminary insights into vendor activity and potential for micro-lending.

Vendor Scorecard Sample Output:

Vendor_Channel	Posting_Frequency_per_Week	Average_Views_per_Post	Top_Product	Top_Price	Average_Price_Point_ETB	Lending_Score
Zemen Express®	42.42	5417.89	None	None	1.66e+07	2730.16
EthioBrand®	10.49	39753.98	##ge	SD Size	1.62e+13	19882.24
ፅዩ እቃ	41.67	26020.60	LeM Hመናዊ	1300	1.53e+10	13031.13
Fashion tera	5.36	9385.30	2	ፋሽን ተራ	3.18e+09	4695.33
ማራኪ ሬጸፈ ŋの™	21.67	11434.00	None	None	3.29e+08	5727.84

Observations & Limitations:

- **Poor Entity Extraction:** The Top_Product and Top_Price columns often contained None or nonsensical extractions. This is a direct consequence of the NER model's low F1-scores, which limits the reliability of business profile metrics derived from extracted entities.
- **Price Parsing Issues:** The Average_Price_Point_ETB values were drastically inflated (e.g., in the billions/trillions). This indicates significant challenges in the price extraction and numeric conversion logic, likely due to the NER model misidentifying non-price numbers as prices or improper parsing of complex price strings.
- **Lending Score Reliance:** The Lending_Score primarily reflects raw engagement metrics (views, posting frequency) rather than actual business profile insights (product, price) due to the NER model's limitations. Its current utility for micro-lending assessment is therefore minimal.

4. Challenges & Solutions

Challenge	Solution / Mitigation
Limited Labeled Data	Initial rule-based labeling of a small subset (50 sentences).
Low NER Model Performance	Compared afro-xlmr-large and DistilBERT; DistilBERT showed relative improvement and higher efficiency. Still requires more data.
seqeval Warnings (L- and U- tags)	Addressed by effective conversion within tokenize_and_align_labels to BIO scheme for model compatibility.
TypeError in shap.maskers.Text	Resolved by correcting parameter passing in shap.Explainer instantiation. (Will verify interactive plotting post-submission).
High Average_Price_Point_ETB Values	Identified as an issue with NER price extraction and parsing; requires more robust regex/logic for price string conversion and better NER performance.
Computational Resources (Training Time)	Selected DistilBERT which offers significantly faster training (~3.7 mins vs ~48 mins for afro-xlmr-large).
Model Interpretability Complexity	Prioritized SHAP for its more direct applicability to token classification, while outlining conceptual LIME approach.

5. Recommendations

Based on the project's findings, the following recommendations are crucial for enhancing the Amharic E-commerce Data Extractor:

- **Prioritize Labeled Data Expansion:** The most critical step is to drastically increase the size and diversity of the labeled Amharic NER dataset. Manual annotation, potentially combined with active learning or semi-supervised methods, will significantly improve model accuracy.
- **Refine Price Extraction Logic:** Implement more robust parsing logic within vendor_scorecard_engine.py to accurately extract numeric prices from NER-identified price strings, filtering out non-price numbers or misclassifications. This may involve post-processing NER outputs.
- **Explore Data Augmentation:** Investigate techniques like back-translation or synonym replacement for Amharic text to artificially increase the training data size, if manual labeling is resource-intensive.
- **Re-evaluate Model Architectures (Post-Data Expansion):** Once sufficient data is available, re-evaluate and potentially fine-tune larger models (like afro-xlmr-large or mBERT) which may yield higher performance with more data, or explore Amharic-specific pre-trained models if available.
- **Improve Model Interpretability:** Once the NER model's performance is acceptable, revisit the model_interpretability.ipynb to fully leverage SHAP and potentially custom LIME implementations to gain deeper insights into model decisions, particularly for correctly and incorrectly identified entities.
- **Integrate Image-based OCR:** Given that 88% of messages contain images and 46% have missing text, incorporating Amharic OCR (Optical Character Recognition) will be essential to extract entities from product images, enriching the dataset and improving the overall system.
- **Refine Lending Score Calculation:** Once the NER outputs for Top_Product, Top_Price, and Average_Price_Point_ETB are reliable, re-evaluate the weighting and components of the Lending_Score to ensure it accurately reflects a vendor's business potential.

6. Conclusion

This project successfully established an end-to-end data pipeline for the EthioMart Amharic E-commerce Data Extractor, from scraping unstructured Telegram data to a preliminary FinTech vendor scorecard. Key achievements include the programmatic collection and preprocessing of Amharic e-commerce messages, the development of a rule-based labeling system, and the fine-tuning and comparison of multilingual transformer models for NER. While DistilBERT showed promise in terms of efficiency and relative accuracy over afro-xlmr-large, the current model performance is limited by the small labeled dataset.

Despite these limitations, the project demonstrates a foundational framework for leveraging NLP in a real-world business context. It highlights the critical importance of high-quality labeled data for effective machine learning. The preliminary vendor scorecard provides a starting point for assessing vendor activity, though its business profile metrics require significant enhancement contingent on improved NER accuracy. The insights gained from this project, particularly regarding data challenges and the need for more comprehensive labeling, will guide future development towards building a robust and impactful solution for EthioMart's FinTech aspirations.

Report by Kaletsidike Mekonnen