Financial Data Extraction Pipeline for Revenue Analysis

1. Introduction

This document outlines the comprehensive data extraction pipeline developed to gather financial and textual data for the analysis of corporate revenue recognition practices. The primary objective is to build a robust dataset by sourcing information from two key public repositories: the **U.S. Securities and Exchange Commission (SEC) EDGAR database** and **Yahoo Finance**. This aggregated data forms the foundation for the subsequent machine learning and analytical models used to detect aggressive revenue recognition patterns.

---

2. Core Components and Data Sources

The pipeline relies on two primary Python libraries to interface with public data sources.

2.1. SEC EDGAR Database (sec-edgar-downloader & Direct API)

The SEC's EDGAR database is the authoritative source for official corporate filings, including the crucial 10-K annual reports.

- **Methodology**: We utilized the sec-edgar-downloader library to programmatically download the latest 10-K filings for a predefined list of companies. For compliance with SEC's fair access policy, all requests included a custom User-Agent string (BDA-project, iib2022034@iiita.ac.in).
- **Data Extracted**: The full text of the latest 10-K filing for each company.

2.2. Yahoo Finance (yfinance)

To supplement the qualitative data from SEC filings with quantitative metrics, we used the yfinance library.

- **Methodology**: This library provides a straightforward interface to access historical and current financial statement data.
- **Data Extracted**:
    - **Company Information**: Ticker, company name, sector, and industry.
    - **Financial Statements**: Key line items from the Income Statement, Balance Sheet, and Cash Flow Statement.
    - **Core Metrics**: Specifically, we targeted Total Revenue, Total Cash From Operating Activities, and Accounts Receivable for the last three fiscal years.

---

3. Data Extraction and Preprocessing Workflow

The extraction process is structured to handle a diverse list of companies and aggregate the required data efficiently.

Step 1: Company Selection

An initial list of 39 companies was curated from various sectors (Technology, Healthcare, Financial Services, Retail, etc.) to ensure the dataset was diverse and representative of different revenue models and reporting styles.

Step 2: Financial Metric Retrieval

For each company in the list, the get_financial_metrics function was executed to:

1. Fetch financial statements using yfinance.
2. Extract the **Total Revenue** for the last three years to calculate **Revenue Growth Rate** and **Revenue Volatility**.
3. Extract **Total Cash From Operating Activities** to later compute a Revenue Quality Score.
4. Extract **Accounts Receivable** to calculate Days Sales Outstanding (DSO).

Step 3: SEC Filing and Textual Data Extraction

This is a critical step for our textual analysis component.

1. **Ticker-to-CIK Mapping**: A utility function maps each company's stock ticker to its Central Index Key (CIK), which is the unique identifier used by the SEC.
2. **Filing Download**: The Downloader class from sec-edgar-downloader is used to fetch the complete text of the most recent 10-K filing.
3. **Text Preprocessing and Feature Extraction**:
    - The initial approach, as seen in the notebook, uses **Regular Expressions (Regex)** and **Beautiful Soup** to locate and extract the "Revenue Recognition" section from the raw HTML/text of the filing. The clean_text function strips HTML tags and normalizes whitespace.
    - **Addressing Feedback**: It was noted that this textual feature extraction could be refined. The current Regex method serves as a baseline for identifying the relevant section. For a more advanced implementation, this step should be enhanced using:
        - **Tokenization and Stemming**: To break down the text into its core components and reduce words to their root form.
        - **Domain-Specific Models (e.g., FinBERT)**: A model like FinBERT, pre-trained on financial text, would provide much richer contextual embeddings of the text, leading to a more nuanced understanding of the company's stated policies than simple keyword matching.

Step 4: Data Aggregation and Output

The quantitative financial metrics and the qualitative textual data are merged into a single, structured **Pandas DataFrame**. This final DataFrame is then saved as a CSV file (FDA3_Revenue_Recognition_Dataset_... .csv), serving as the clean input for the ML/Analytics pipeline.