Core ML tasks:

#### **Goal 1: "Where did my money go?" (What categories am I spending on?)**

**ML Task: Budget Categorization Model**

* **Objective:** To consistently and automatically assign a predefined spending/income category to every transaction, even if the original Category is missing or inconsistent. This forms the bedrock of understanding spending.
* **How your data helps:** We already have Category labels in both datasets! This is our labeled data for supervised learning. We will use TransactionDescription as input features and OriginalCategory (mapped to Final Categories) as your target labels.
* **Data Used (after cleaning):**
  + TransactionDescription (your cleaned text input).
  + OriginalCategory (used as the ground truth, after standardization to Final Categories).
* **Model(s) to Leverage:**
  + **Support Vector Machine (SVM) with TF-IDF Features:** Excellent for classifying text data, robust.
  + **Multinomial Naive Bayes:** Simple, fast baseline, good for a first pass.
  + **Logistic Regression:** Good balance of performance and interpretability.

#### **Goal 2: "What are my fixed costs?" (What bills/subscriptions come out regularly?)**

**ML Task: Recurring Expense & Subscription Identification**

* **Objective:** To pinpoint regular, predictable financial outflows that occur at consistent intervals (e.g., monthly rent, streaming service subscriptions, utility bills).
* **How your data helps:** The TransactionDescription, Amount, and TransactionDate columns contain the patterns needed to identify regularity.
* **Data Used (after cleaning):**
  + TransactionDescription (cleaned text).
  + Amount.
  + TransactionDate.
* **Derived Features (Crucial):**
  + Calculate time differences between similar transactions.
  + Measure the frequency of similar transactions over time.
* **Model(s) to Leverage:**
  + **DBSCAN Clustering:** Group transactions based on density in a feature space defined by Amount consistency, TransactionDescription similarity, and Time Difference regularity.
  + **Hybrid Rule-Based/Pattern Matching with Frequency Analysis:** Identify patterns like "same description + same amount + ~30-day interval".

### **Core Problem 3: "How much can I realistically save?" (Am I overspending, or can I afford my goals?)**

**Expanded ML Task: Granular Cash Flow Forecasting & Spending Prediction**

* **Objective:** To provide a detailed projection of expected income and expenses for future periods (e.g., next month, next quarter), broken down by category. This allows users to understand not just their potential savings, but *where* their money is likely to go, enabling proactive budgeting and goal planning.
* **What it does:**
  + **Learns Historical Patterns:** The model will analyze our past income and spending trends from the cleaned and categorized data. It looks for:
    - **Seasonality:** Do expenses in certain categories always spike in specific months (e.g., higher utilities in winter, holiday spending in December)? Do they get paid on a certain day of the month?
    - **Trends:** Is your spending on "Dining Out" gradually increasing or decreasing over time? Is the income stable?
    - **Recurring Events:** It implicitly leverages the patterns of the recurring bills (identified by the previous ML task) by seeing their regular occurrence in the historical data.
  + **Projects Future Income & Expenses:** Based on these learned patterns, the model will forecast:
    - **Expected Total Monthly Income:** "Based on your past, we predict your income next month will be approximately $X."
    - **Expected Total Monthly Expenses:** "Your total expenses next month are forecasted to be around $Y."
    - **Expected Spending by Category:** **This is the key detail you asked for!** "You are likely to spend about $A on 'Groceries', $B on 'Transportation', and $C on 'Dining Out' next month." This breakdown makes budgeting actionable.
    - **Projected Net Cash Flow/Savings:** The difference between forecasted income and expenses.
* **How the data helps:**
  + **Rich Historical Data:** our combined datasets (with TransactionDate, Amount, TransactionType, and the Final Categories from the categorization model) provide the essential time-series data needed for robust forecasting. The more historical data, the better the predictions.
  + **Currency Consistency:** If we decided to convert all amounts to a single currency during data preparation, the forecasts will be unified. Otherwise, you'd forecast for each currency separately.
* **Model(s) to Leverage:**
  + **Prophet (Facebook's Forecasting Tool):**
    - **Strengths:** Highly effective for business forecasts, automatically handles daily/weekly/monthly seasonality, trends, and can even incorporate known future events (like annual bonuses if we manually input them). We can train a separate Prophet model for *each* important category (e.g., one for "Groceries", one for "Transportation", one for "Income") to get granular forecasts, then sum them for total spending.
  + **Seasonal ARIMA (SARIMA):**
    - **Strengths:** A powerful statistical method for time series that explicitly models autoregressive, integrated, and moving average components, including seasonal variations. Good for capturing complex dependencies in the data. We would similarly build separate SARIMA models for each category's spending.
* **Output and "Other Things" it can help with:**
  + **Proactive Budgeting:** Instead of reacting to spending, users can *plan* for it. "I see I'm projected to spend $600 on 'Dining Out' next month. I only want to spend $400, so I need to adjust."
  + **Goal Attainment:** "To save $1000 by next month, based on my projected income, I need to reduce my forecasted spending by $200. Where can I cut back?" The chatbot can then suggest categories with high forecasted spending.
  + **Affordability Checks:** "Can I afford that $800 gadget next month?" The chatbot checks the projected disposable income/savings for that month.
  + **Early Warning System:** If the forecast shows a potential deficit, the chatbot can alert the user early.