**Step 1: Define a Ground Truth Dataset**

* **Ground Truth**: Collect and manually label about 50-70 apps (as discussed) as "should be regulated" or "not regulated." For this, we can use a mix of descriptions from known regulated apps and those suspected of being non-compliant.
* **Manual Reviewing**: Manually review app descriptions and features to ensure accurate categorization, paying attention to keywords from the FTC guidelines (like "personal data sharing" without consent or "treatment" claims without medical verification).

**Step 2: Text Preprocessing**

* **Feature Extraction**: Use key phrases and terms we identified as relevant (e.g., "misleading claims," "personal health information," "HIPAA compliant," "data sharing agreement").
* **Clean Text**: Convert all app descriptions to lowercase, to be used for comparison to keywords and phrases retrieved from FTC Laws.

**Step 3: Keyword and Key Phrase Matching**

* **Pattern Matching**: Match the keywords and phrases we extracted against the app descriptions to see if they contain any red-flag terms that indicate they might need regulation (e.g., terms like “diagnose,” “treat,” “cure,” “secure data”).
* **Weighted Scoring**: Assign weights to each term based on its importance in indicating non-compliance (e.g., higher weight to “medical treatment” or “user consent”).

**Step 4: Rule-Based Classification**

* **Create Rules Based on FTC Guidelines**: Develop a set of rules based on the FTC’s language. For example:
  + If an app description mentions "diagnosis" without disclaimers, mark it for further review.
  + If an app claims "privacy assurance" but includes terms indicating data sharing with third parties, flag it as potentially deceptive.
* **Apply Rules to Labeled Dataset**: Run these rules on the ground truth dataset to fine-tune them and improve accuracy.

**Step 5: Machine Learning Model for Classification**

* **Train a Model**: Use the labeled ground truth dataset to train a supervised machine learning model (e.g., Naïve Bayes, Support Vector Machine) on whether an app “should be regulated” or “not regulated.”
* **Features for Model**: Use text vectorization (TF-IDF or even embeddings from models like BERT) to represent app descriptions.
* **Fine-Tuning**: Adjust the model based on false positives and negatives from the labeled data to ensure it accurately reflects FTC compliance indicators.

**Step 6: Validation and Testing**

* **Test the Model**: Run the model on a subset of app descriptions outside the training data to evaluate accuracy.
* **Adjust Thresholds**: Fine-tune thresholds for keyword matches, rule triggers, and model predictions to ensure both precision and recall meet project goals.

**Step 7: Deployment and Continuous Update**

* **Automate Classification**: Once validated, apply this model to classify all the scraped apps.
* **Periodic Review**: Periodically review new FTC updates and update the keywords, phrases, and rules accordingly.