# 2. Technical Documentation

## Approach and Algorithms

The solution to extract key details from invoices uses a combination of **Visual Language Models (VLMs)** and **direct text extraction techniques**. Our core model, **Qwen-2VL**, processes the PDFs, utilizing natural language prompting to extract key values like supplier details, item names, rates, tax amounts, etc., directly into JSON format.

### Why Qwen-2VL?

**Qwen-2VL** is a Visual Language Model that is particularly well-suited for tasks involving complex documents like invoices due to the following reasons:

- **Comprehensive Training on Language and Images**: Qwen-2VL has been trained on trillions of tokens of language data and millions of images, providing it with a deep understanding of both **visual layout** and **natural language**. This enables the model to recognize and process structured information in documents such as tables, itemized lists, and tax breakdowns with high accuracy.
- **Enhanced Visual Layout Understanding**: Since Qwen-2VL's encoder is specifically trained on large-scale image data, it excels at identifying and interpreting visual elements in documents. This includes tables, borders, multi-column layouts, and different font sizes—all of which are crucial for accurate data extraction from invoices.
- **Natural Language Understanding**: The model's deep exposure to diverse language data allows it to comprehend and process unstructured or semi-structured text with high contextual accuracy. It understands the meaning of terms like "Rate," "Tax," and "Total Amount" within the invoice layout, allowing for reliable extraction of key fields without needing to rely solely on fixed positional patterns like regex.
- **Flexibility with Natural Language Prompts**: Unlike traditional methods that depend on strict extraction rules, Qwen-2VL can be prompted using natural language to extract data. This flexibility significantly reduces the complexity involved in creating different extraction rules for various invoice formats, as you can adapt the prompt instead of the extraction logic.
- **Scalability and Adaptability**: Traditional extraction methods like regular expressions are highly dependent on specific layouts, which can vary widely across invoice formats. Qwen-2VL adapts to different formats seamlessly without requiring additional coding efforts to handle variations, making it more scalable for large-scale use.
- **Cost-Effective Long-Term Solution**: While regex-based solutions may offer short-term benefits in terms of processing speed, they require continuous updates to handle new invoice formats. Qwen-2VL, on the other hand, is a cost-effective long-term solution due to its ability to handle a wide range of document types without frequent manual intervention.

## Method Justification

- **Cost-Effectiveness vs Accuracy**: While regex-based extraction may seem more cost-effective initially, it faces significant challenges when dealing with diverse invoice formats. In a sample of **24 PDFs**, we found that around **3 to 4 distinct templates** required custom regex patterns to properly extract the necessary information. This number of templates can increase exponentially when scaling to larger datasets or multiple sources, particularly when the data is noisy. Such template variability can become a bottleneck in data pipelines, as each new source or layout introduces the need for more manual intervention to create or modify extraction rules.
- **Scalability with VLM**: In contrast, the **Qwen-2VL** approach is inherently more scalable. Even if new data sources introduce different layouts, the extraction logic can be adapted directly through **prompt engineering** without the need for complex regex adjustments. Additionally, the model can be **fine-tuned** to handle specific changes in live data streams, ensuring that the extraction process remains efficient and accurate over time.

This adaptability and flexibility make the VLM-based method much more suitable for large-scale, real-world use cases, where maintaining extraction rules across many data sources can quickly become unmanageable with regex.

## Trust Determination

To ensure 99% trustworthiness in extracted data, we implemented a multi-step approach:

1. **Ground Truth Creation**: For PDFs where the text can be extracted directly, we generated a **ground truth** by extracting key values and matching them directly in the text. Any mismatch in these key values (such as item names, rates, or tax amounts) is flagged, and trust can be improved through **prompt engineering** and **multiple sampling** within the VLM.
2. **Handling Scanned Images**: For PDFs containing scanned images where text extraction isn't straightforward, we used multiple OCR systems to create a reliable ground truth for each data point. Since a single OCR (like **Tesseract**) only provided around **89% accuracy**, combining outputs from multiple OCRs helped improve the reliability of the extracted data. This approach significantly improves trust determination when direct text extraction is not possible.
3. **Combining OCRs and VLMs**: For trust prediction in cases where the PDFs are images and not text-based, the system can combine results from multiple OCRs to ensure the extracted values are consistent. This process enhances trust determination, as a single OCR’s accuracy may not match the performance of the VLM in generating accurate ground truth.

By using this hybrid approach of direct extraction from text-based PDFs and multiple OCR systems for scanned documents, we ensure a high level of trust and accuracy in the final extracted data.

# 3. Accuracy and Trust Assessment Report

## System Accuracy

We observed that the performance of the **Visual Language Model (VLM)** depends significantly on **prompt engineering**. In the initial inference phase, the model struggled to extract specific fields like **Tax Rates** (CGST/SGST) from invoices. However, with careful prompt modifications, the extraction accuracy improved. For example, the modified prompt:

**"CGST/SGST: Percentages, Amounts (These are in a pair, meaning in a pair CGST and SGST amounts will be the same, but there may be many pairs of them)"**

helped the model correctly extract both the CGST and SGST fields, ensuring precise data capture for tax information.

## Handling Variability in Formats

In our analysis of PDFs, we noticed significant variation in format and label names across different invoices. For instance, one PDF had a unique field labeled **"Payment Left"**, which did not appear in any other invoice. A traditional regex-based extraction approach would fail to handle such variability without predefined patterns, resulting in missed information. In contrast, the VLM is flexible enough to capture this data under general categories like **"Total Details"**, ensuring minimal loss of information and a more comprehensive dataset for analysis.

## Impact of Prompt Engineering

This adaptability is one of the key advantages of using VLMs over regex-based methods. While regex requires highly specific patterns that can easily break with new formats or fields, VLMs allow for real-time prompt adjustments and model fine-tuning to account for unexpected variations in the data. This leads to improved accuracy, particularly when working with noisy, unstructured, or diverse invoice formats, as we observed in our tests.

By leveraging **prompt engineering** and model flexibility, we reduced the chances of data loss and ensured more accurate extractions, leading to a richer dataset for downstream analysis.

## Trust Determination and Accuracy Metrics

To achieve the desired **99% trust determination**, we employed an **exact match criterion** where the extracted text was directly compared to the ground truth based on **overlap** in key values. However, this method resulted in an overall accuracy of **96%** across the sample data.

Upon manually reviewing the PDFs with lower accuracy, we identified several instances of **false negatives**. For example:

- The **name of an item** might differ due to minor variations like a **space** or **hyphen**, which caused the exact match criterion to fail.
- Similarly, in some cases, the **amount values** were mismatched by as little as **0.5 units**, leading to inaccurate classifications.

### False Negative Analysis

Through manual correction of these false negatives, we observed that the lowest accuracy across the PDFs was **around 93%**, while the highest accuracy reached **100%** for some documents. This indicates that the model's overall accuracy is likely higher than what the strict exact match criterion suggests.

### Real Accuracy Estimation

Considering the corrections and further review, we estimate the **real accuracy** of this VLM-based setup to be approximately **97%**. We define accuracy as:

$$Accuracy = \frac{Number\ of\ Correct\ Fields\ in\ Extracted\ JSon}{No\ of\ Total\ fields\ in\ Extracted\ Json}$$

This estimation reflects a more nuanced understanding of the model's performance, particularly where small discrepancies like character spacing or fractional differences in amounts are concerned. By improving prompt engineering and refining the matching criteria, we aim to further reduce false negatives and improve the trustworthiness of the extracted data.

# 4. Performance Analysis

### Resource Utilization

The performance and resource utilization of the **Qwen-2VL** model depends on the workload type:

- **Batch Processing**: In scenarios where real-time processing is not required, the resource usage can be significantly optimized by taking advantage of **Dynamic Cloud GPU pricing**. For instance, using an **L4 GPU**, the model can process around **5-8 documents per second**, with a cost of approximately **$0.43 USD per hour**. This means that for **20,000 documents**, the total cost would be around **$0.43 USD**, leading to a per-document cost of approximately **$2.15e-5 USD** (around **2 paise per document**). This is highly efficient for batch processing workloads, particularly for larger-scale document extractions.
- **Real-Time Processing**: For real-time applications, the system can be **auto-scaled** using tools like a **Kubernetes Cluster** to dynamically manage resource allocation based

on demand. This setup can easily scale to accommodate spikes in document inflow, with inference engines like **TensorRT** or **VLLM** further optimizing performance for low-latency requirements. These inference engines provide improved processing speed, helping to maintain system scalability while controlling costs.

## Model Distillation for Cost Optimization

At a company scale, where data is continuously processed, the model can be **distilled** into a smaller, more cost-efficient version, such as the **Qwen-VL 2B Instruct Model**. Although this requires an initial investment in model training, the inference costs can be significantly reduced in the long term. For example, the inference cost could be cut down to a quarter, resulting in a per-document cost of **0.25 paise**. This approach would be ideal for large-scale enterprises looking for continuous processing with lower resource overhead.

## Cost-Benefit Analysis

In terms of cost-effectiveness:

- **VLM vs OCR/Regex**: Direct extraction using **OCR** or **regex**-based methods may be beneficial in cases where the dataset is **homogeneous** and doesn't involve much variability in document formats. For such cases, clustering the dataset based on source types and applying specific regex rules can be efficient in a batch processing environment. However, this method involves significant **engineering hours** for creating and maintaining regex patterns for each format. It becomes less viable as the dataset grows more diverse or contains noisy data, as we saw in our tests with varied invoice templates.
  On the other hand, the **Qwen-2VL** model offers more flexibility and scalability, particularly for datasets with diverse formats. The **prompt engineering** required for VLMs can be adapted with ease to changing document structures without the need for continuous regex updates, making it more efficient in the long run, especially for **dynamic or real-time use cases**.

In conclusion, while regex and OCR may work in specific scenarios with well-defined and stable templates, VLM-based extraction is more scalable and cost-effective when dealing with varied and unstructured data sources, particularly when optimized using cloud infrastructure and scaling strategies.

---