📊 RAG Evaluation Report

Overview

This report presents the evaluation of two chunking strategies (**Fixed** and **Recursive**) combined with two similarity search algorithms (**Cosine** and **Dot Product**) using **Qdrant** as the vector database and **Groq** for LLM inference.

The evaluation was performed on five representative user queries against the *Dropit Nepal Terms & Conditions* dataset. Each setup was tested on:

- **Accuracy**
- **Precision**
- **Recall**
- **F1-score**
- **Latency**

Results Summary

Strategy + Similarity	Accuracy	Precision	Recall	F1-score	Latency
Fixed + Cosine	0.80	0.60	0.90	0.70	**7.09s**
Fixed + Dot Product	0.80	0.60	0.90	0.70	**5.31s**
Recursive + Dot Product	0.80	**0.70**	**1.00**	**0.80**	14.72s
Recursive + Cosine	0.80	**0.70**	**1.00**	**0.80**	7.01s

Key Findings

1. **Recall**: Both recursive methods consistently achieved **perfect recall (1.00)**, showing they capture all relevant chunks better than fixed strategies.

2. **Precision & F1**: Recursive methods yielded higher **precision (0.70)** and **F1 (0.80)** compared to fixed chunking (0.60 / 0.70).

3. **Latency**:
    - Fastest: **Fixed + Dot Product (5.31s)**
    - Slowest: **Recursive + Dot Product (14.72s)**
    - Recursive + Cosine provided the **best balance** between accuracy and speed (F1 = 0.80, Latency = 7.01s).
4. **Cosine vs. Dot Product**: Performance differences were minor, but **Cosine** paired better with recursive chunking (slightly lower latency, equal accuracy).

Recommendations

- **Production Setup**: Use **Recursive + Cosine** as the default strategy for better balance between retrieval quality and latency.
- **Latency-Critical Scenarios**: Use **Fixed + Dot Product**, which is fastest but with slightly lower retrieval quality.