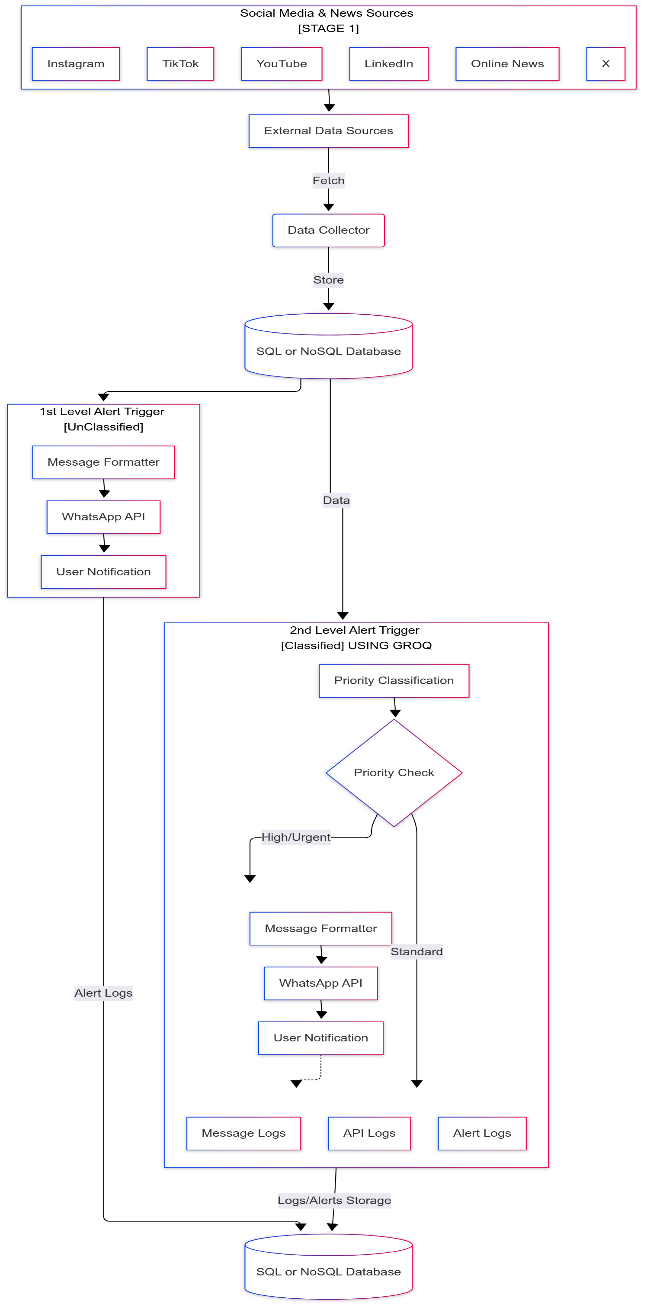
# Social Media Monitoring System

**System Overview**

This document outlines a data processing pipeline that monitors various social media platforms and news sources to generate user alerts. The system collects data from external sources, processes it through classification models, and delivers notifications based on priority levels.

**Architecture Components:**

****

**Stage 1: Data Collection**

* **Social Media & News Sources**: **TikTok**, ~~LinkedIn~~, **Instagram,Online News**, **X (Twitter), YouTube**
* **External Data Sources**: Aggregation point for all source inputs
* **Data Collector**: Component that fetches and normalizes data from multiple platforms
* **Database Storage**: SQL or NoSQL database for storing collected data

**Stage 2: Alert Processing**

* **Alert Triggers**: Two levels of processing
  + **1st Level Alert Trigger [Unclassified]**: Basic filtering of incoming data
  + **2nd Level Alert Trigger [Classified]**: Advanced analysis using GROQ for LLM processing
* **Priority Classification**: Determines the urgency and importance of alerts
* **Message Formatter**: Prepares notification content for delivery
* **WhatsApp API**: Interface for sending notifications to end users
* **User Notification**: Final delivery of alerts to users

**Stage 3: Logging & Storage**

* **API Logs**: Records of API interactions
* **Alert Logs**: Records of alerts generated
* **Message Logs**: Records of messages sent
* **Logs/Alerts Storage**: Persistent storage of all system activity

**Data Flow**

1. Content is **collected** from various social media platforms and news sources
2. The Data Collector fetches and normalizes this information
3. Collected data is stored in a primary database
4. Data is processed through one of two alert paths:
   * **First-level (unclassified)** alerts are formatted and sent directly
   * **Second-level (classified)** alerts undergo priority classification using RAG based Approach.
5. Based on **priority (Low/Medium/High/Urgent)**, messages are formatted and delivered
6. All activities are logged and stored for analysis and auditing
7. **Service Pricing Details**

| **Name** | **Service** | **Links** | **Cost** | **Unit** | **Fixed/Variable** | **Remark** |
| --- | --- | --- | --- | --- | --- | --- |
| **News API** | Newsapi | [NewsAPI](https://newsapi.org/) | 250 $ | Fixed | Fixed cost | Fixed cost |
| **Social Data** | Apify | [Apify](https://apify.com/) | 150 $ | Variable | Can increase $0.4 per post as per needs |  |
| **Whats API** | Meta | [WhatsApp API](https://business.whatsapp.com/products/platform-pricing?country=India&currency=Indian%20Rupee%20(INR)&category=Marketing) | 18 $ | Variable | Per message (Rs. 0.7846) for 2000 messages |  |
| **Code Execution on Cloud** | AWS Lambda | [AWS Lambda](https://aws.amazon.com/pm/lambda/) | 20 $ | Variable | Will increase depending on number of requests |  |
| **Data Storage and Retrieval** | AWS S3 | [AWS S3](https://aws.amazon.com/s3/pricing/) | 4 $ | Variable | Assuming 100 GB, will increase as data increases |  |
| **Data Storage** | MongoDB | [MongoDB](https://www.mongodb.com/pricing) | 60 $ | Variable | Pay per use |  |
| **LLM** | GROQ | [GROQ](https://groq.com/pricing/) | 30 $ | Variable | To analyze around 2000 articles per month |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Scaling Considerations**

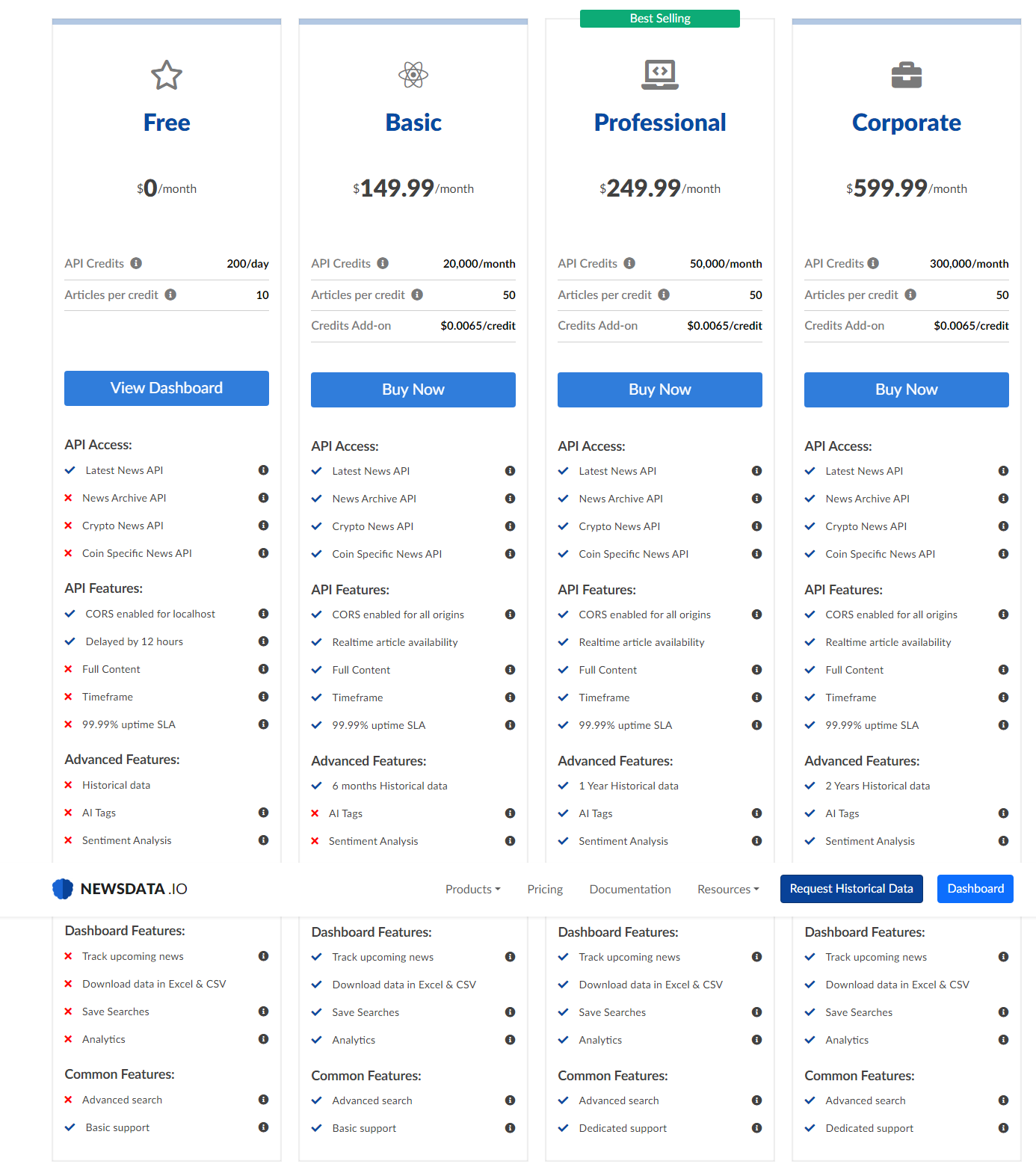
* Lambda functions will scale automatically based on incoming data volume
* MongoDBcan be configured with auto-scaling for read/write capacity

**Data Collection Implementation**

**API’s to be Used :**

**News API Integration**

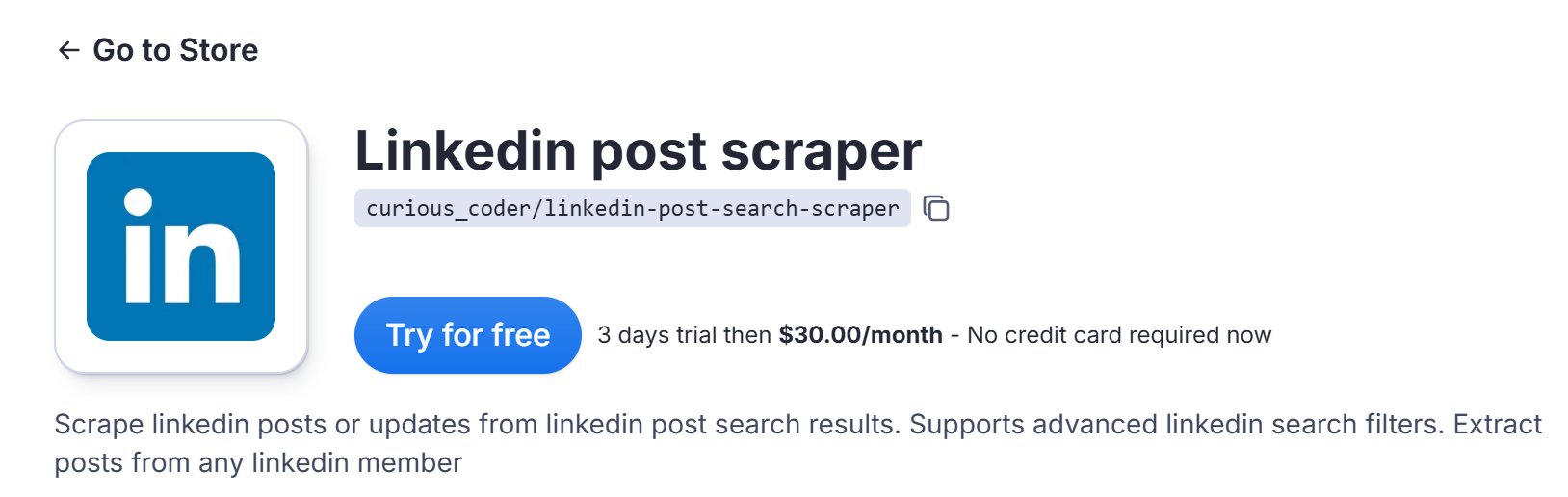
* **Service:** [Newsdata.io](https://newsdata.io/news-sources)
* **Pricing:** [Link](https://newsdata.io/pricing)



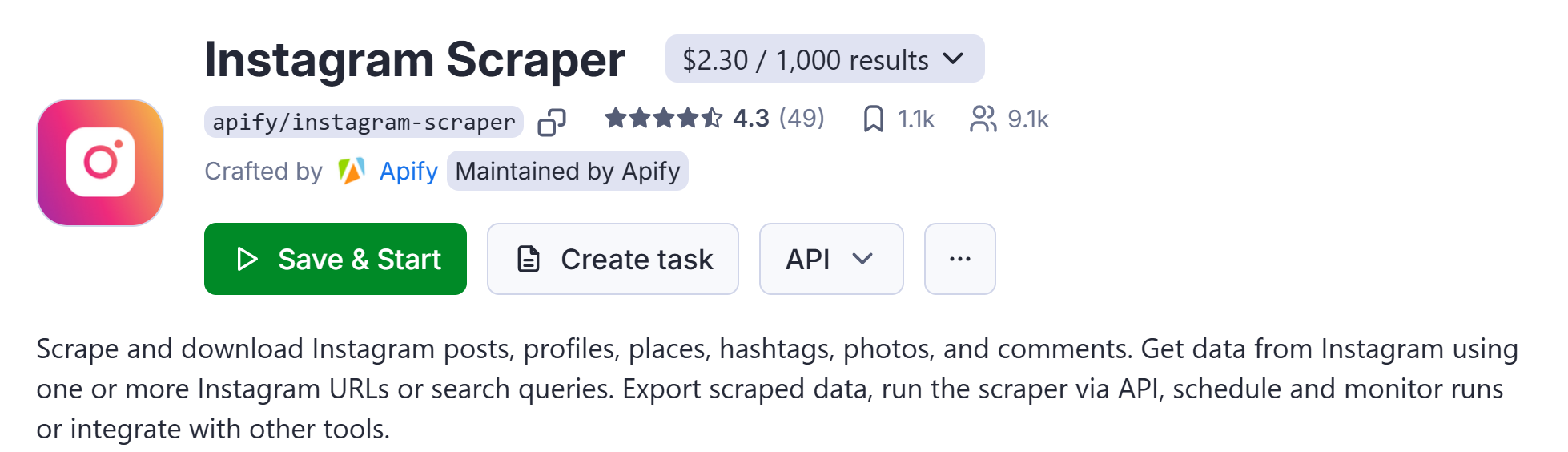
* **Fields:**
  + **Title-**The title of the news article.
  + **Article link-**Link to the news article on the web.
  + **PubDate-**The publish date of the news article.
  + **Author-**The author of the news article.
  + **Publisher-**The publisher’s domain name.
  + **Country-**The country name of the publisher.
  + **Language-**The language of the news article
  + **Description-**The description of the news article.

**Apify Integration**

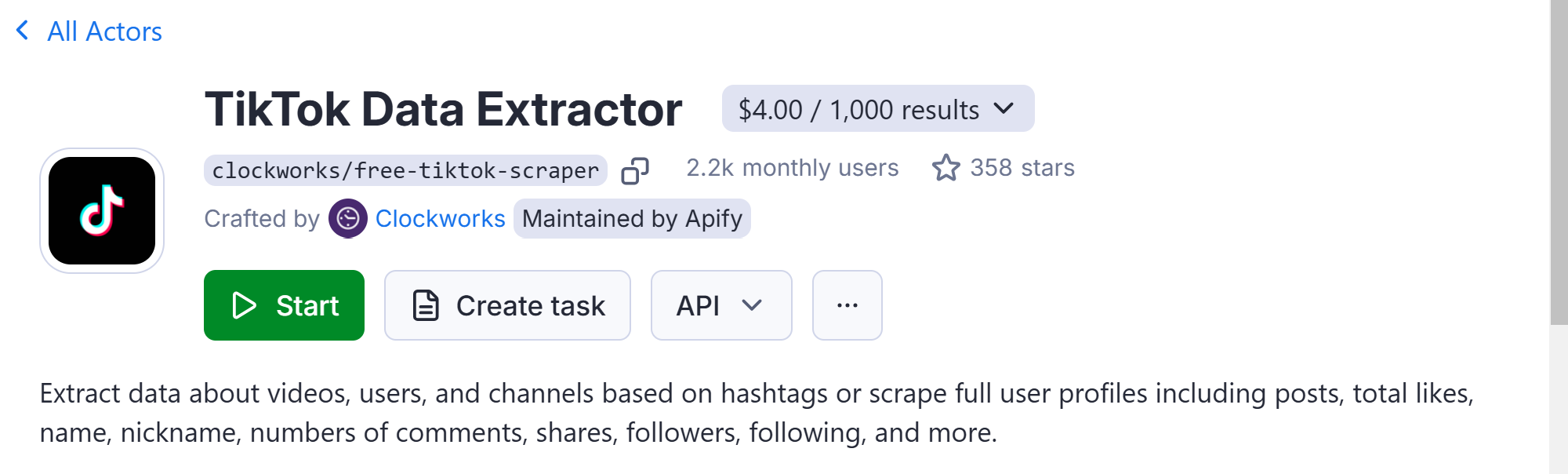
* **Service**: [Apify](https://apify.com/)
* **Actors Used**:
  + **LinkedIN**



* + **Instagram Scraper**



* + **TikTok Scraper**

****

* + Twitter Scraper

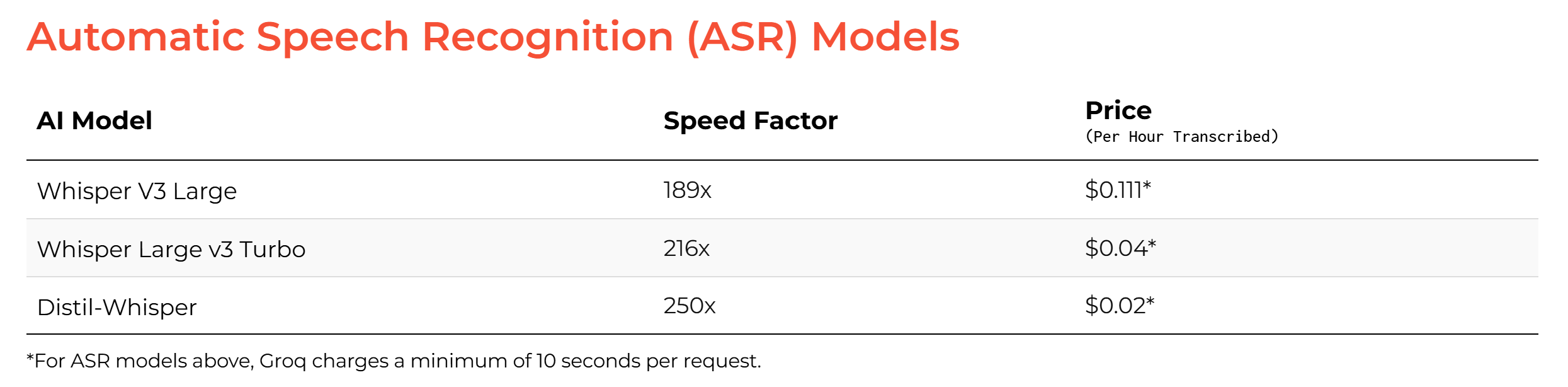
Checking IP

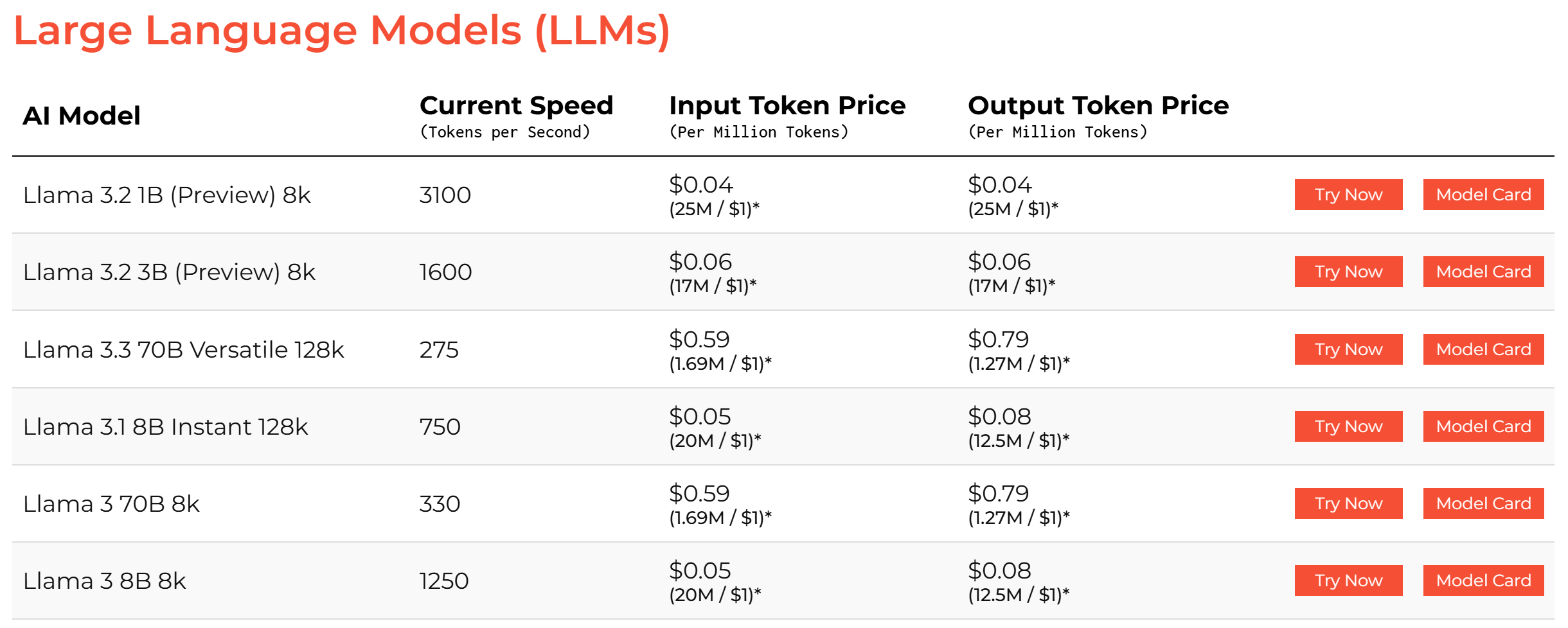
* + YouTube Scraper

Using **Youtube API** – Can use the Youtube API for pulling the data from youtube

**GROQ Integration for LLM Processing**

* **Service**: [GROQ](https://groq.com/)
* **Models Used**:
  + LLaMA 3 70B
  + Mixtral 8x7B
  + Deepseek R1
* **Pricing:** [**Link**](https://groq.com/pricing/)





**Maintenance and Support**

**Regular Maintenance Tasks**

* API key rotation every 90 days
* Database optimization monthly
* Log analysis and cleanup weekly