

## Brody's Guide: Setting Up the Crypto Market Analyzer n8n Workflow

This automation aggregates crypto news from multiple sources, analyzes sentiment using Google Gemini, and sends reports via Telegram/Discord. Follow these steps:

---

### 1. Prerequisites

- **n8n Account:** Sign up at [n8n.io](#) (Cloud) or [install locally](#).

- **Google Gemini API Key:**

Get it from [Google AI Studio](#).

- **Telegram Bot Token:**

Create via [@BotFather](#).

- **Discord Webhook URL:**

Create in your Discord channel:

*Channel Settings → Integrations → Webhooks → New Webhook.*

---

### 2. Import the Workflow

1. In n8n, go to **Workflows → Import**.
  2. Upload **Crypto\_Market\_Agent.json** or paste its content.
  3. Click **Import**.
- 

### 3. Configure Critical Nodes

#### A. Google Gemini (AI Analysis)

- **Nodes:** **Google Gemini Chat Model** & **Google Gemini Chat Model1**
- **Steps:**
  1. Open each node.

2. Paste your **Google Gemini API Key**.
3. Set parameters (e.g., temperature=0.7 for balanced creativity).

## B. Telegram/Discord (Output)

- **Telegram Nodes:**
  - **Send Message To Telegram** (ETH) & **Send Message To Telegram1** (BTC)
  - Add your **Bot Token** and **Chat ID** (get from @userinfobot on Telegram).
- **Discord Nodes:**
  - **Discord - Send Message Node** (BTC) & **Discord - Send Message** (ETH)
  - Add your **Webhook URL**.

## C. RSS Feed Sources

- **Example Configuration** (for **RSS Read - Cointelegraph BTC**):

json

None

```
"url": "<https://cointelegraph.com/rss/tag/bitcoin>" ,  
"dateFormat": "YYYY-MM-DD"
```

- **Repeat** for all 18 RSS nodes using these sources:

- Cointelegraph, NewsBTC, Bitcoinist, CryptoSlate, etc.
- *Tip:* Replace placeholder URLs with active RSS feeds.

---

## 4. Customize Filters & Logic

### A. Date Filters

- **Nodes:** **Date Filter** (BTC) & **Date Filter1** (ETH)
- **Set Conditions:**
  - Keep items where **pubDate** is  $\geq$  **today - 24 hours**.

- *Example Rule:*

js

None

```
{  
  "field": "pubDate",  
  "type": "date",  
  "operation": "after",  
  "value": "{{ $now.minus({ hours: 24 }) }}"  
}
```

## B. Content Organizers

- **Nodes:** [Organize Content for GPT \(ETH\)](#) & [Organize Content For GPT \(BTC\)](#)
- **Add Code** to structure RSS data for Gemini:

js

None

```
return items.map(item => ({  
  json: {  
    title: item.title,  
    summary: item.description,  
    link: item.link,  
    date: item.pubDate  
  }  
}));
```

## 5. Schedule Execution

- **Node:** [Schedule Trigger](#)

- **Set Interval** (e.g., daily at 9 AM UTC):

json

None

```
"rule": {  
    "interval": [24],  
    "hour": 9,  
    "minute": 0  
}
```

## 6. Activate & Test

1. **Save** the workflow.
2. **Toggle Active** (top-right) to enable.
3. **Test Manually**:
  - Click "Execute Node" on the **Schedule Trigger**.
  - Check execution logs for errors.
  - Verify Telegram/Discord outputs.

## 7. Troubleshooting Common Issues

- **RSS Failures**:
  - Check feed URLs are valid (test in browser).
  - Use [RSS Validator](#).
- **Gemini Errors**:
  - Ensure API key has sufficient quota.
  - Reduce text length if hitting token limits.
- **Telegram/Discord**:
  - Verify bot permissions/webhook URLs.

## 8. Optimization Tips

- **Rate Limits**: Add **Wait** nodes between RSS calls to avoid bans.
- **Error Handling**: Use **IF** nodes to catch failures (e.g., empty feeds).
- **Enrich Data**:

- Add nodes to fetch BTC/ETH prices from CoinGecko API.
  - Combine with TA-Lib for technical analysis.
- 

## Final Workflow Structure

### Diagram

Code

**Done!** Your workflow now scans crypto news daily, analyzes sentiment, and sends alerts.

### Next Steps:

- Add email/SMS alerts via Twilio.
- Integrate with TradingView for price alerts.
- Extend to other coins (e.g., SOL, XRP).