**Project Description: EnergyQuantGPT**

EnergyQuantGPT is an AI-driven tool designed to address the dynamic and complex challenges of the energy sector, with a particular focus on geopolitical risks and energy market analysis. Built using machine learning algorithms, natural language processing capabilities, and dataset integration, the model provides insights tailored for stakeholders such as investors, energy distributors, and policymakers.

**Core Features & Prompts (to name a few!):**

**Energy Dashboard (*prompt: Energy Dashboard*)**:

* In summary, the dashboard gives Energy stakeholder the basic updated information via news API’s of the current rates on key items at the click of a button.
* A dynamic, data-driven visualization tool that integrates key energy market metrics such as natural gas prices, LNG import trends, and gas storage levels.
* Outputs include interactive charts and graphs summarizing historical trends, current market conditions, and potential future scenarios.
* Automatically processes and presents data uploaded by the user, enabling real-time analysis and decision-making.

A screenshot of a graph

Description automatically generated

A screenshot of a graph

Description automatically generated

**Geopolitical Risk Analysis (*eg prompt: Geopolitical Risk for Germany*)**:

* Quantifies the geopolitical risk for individual countries or regions based on factors like dependency on Russian gas imports, LNG import trends, gas storage levels, and price volatility.
* Generates normalized risk scores (0–5 scale) for easy interpretation and comparison.
* **A paper with text and numbers

  Description automatically generated**Provides detailed dashboards for specific countries, summarizing risk metrics in a visually digestible format.

A screenshot of a document

Description automatically generated

Geopolitical events like supply disruptions or sanctions are unpredictable. This tool quantifies those risks with a normalized score, combining factors like Russian gas dependency, LNG trends, storage levels, and price volatility.

Take Germany: a major energy consumer in Europe. Its risk score is driven by a high dependency on Russian gas. On the other hand, countries like Norway, with diversified supply chains, rank far lower.

**Scenario-Based Insights**

1. **Energy Distributor Scenario analysis (*eg prompt: What happens if Russian gas exports to Europe decrease by 20%? Show the impact on geopolitical risk scores for all countries in the dataset*)**

* Enables users to simulate potential events, such as a 20% decrease in Russian gas exports, and assess their impact on geopolitical risk and market trends.
* Facilitates strategic planning by modelling how changes in policy or market conditions affect energy dependency and pricing.

Screens screenshot of a cell phone

Description automatically generated

Imagine a 20% reduction in Russian gas exports. EnergyQuantGPT projects the impact: Germany’s risk score rises significantly, while Norway remains stable. This allows distributors to anticipate supply chain vulnerabilities and make proactive adjustments.

Distributors no longer have to rely on intuition. This model provides clarity and actionable insights in uncertain markets.

1. **Investment Recommendations (*eg prompt: Where should I invest £50 million in a green hydrogen electrolyser? Consider geopolitical risk and energy infrastructure readiness in Europe*)**

* Offers location-based analysis for investment opportunities, such as building a green hydrogen electrolyser or expanding LNG infrastructure.
* Uses geopolitical risk scores and energy infrastructure readiness to recommend optimal locations for investments.
* Screens screenshot of a phone

  Description automatically generatedTailored outputs designed for investors to evaluate risk-adjusted returns effectively.

A screenshot of a graph

Description automatically generated

This model evaluates geopolitical risk and infrastructure readiness to guide your decision.

For example, Norway’s low risk score and high renewable infrastructure make it the ideal choice for a long-term investment. Alternatively, Germany might score higher on risk but offers a larger market for distribution.

This is not just a tool—it’s a strategy engine for informed investment decisions.

1. **Policy Impact Simulations (*eg prompt: Simulate the impact of stricter LNG import tariffs across Europe. Show risk score changes and energy price trends for key countries*)**

* Evaluates the potential outcomes of policy changes, such as imposing stricter LNG import tariffs.
* A screenshot of a phone

  Description automatically generatedProvides insights into how such changes could influence energy prices, storage levels, and geopolitical risk scores.

A screenshot of a graph

Description automatically generated

What if European nations imposed stricter LNG import tariffs? EnergyQuantGPT simulates outcomes, quantifying impacts on risk scores and energy prices for each country.

For instance, France’s reliance on LNG imports would see a sharper increase in risk, pushing for alternative energy sources. Policymakers can use this foresight to adjust energy diversification plans.

**Data and Inputs:**

EnergyQuantGPT leverages a comprehensive and curated dataset, including:

Historical Data

* **Russian Gas Dependency**: Data on country-level import dependency on Russian gas.
* **LNG Import Trends**: Historical and current trends in LNG imports across Europe.
* **Gas Storage Levels**: Weekly data on gas storage levels within EU countries.
* **Natural Gas Prices**: Historical price trends and volatility in natural gas futures markets.

All data shown is from 1950 - today

Summary:

**European Natural Gas Import Statistics**

A screenshot of a computer

Description automatically generated

A close-up of a document

Description automatically generated

**Impact of the Russia-Ukraine Conflict on Energy Supply**

A screenshot of a computer

Description automatically generated

**Natural Gas Price Data**

A screenshot of a computer

Description automatically generated

Geopolitical News and Updates

**Google News API**

* Provides access to Google News articles and headlines based on specific search queries.
* Can return results from various regions, languages, or categories.

**Thomson Reuters**

* A well-known API for financial and commodity markets, including natural gas.
* Provides sentiment analysis and news metadata tailored to energy and commodity sectors.

**Technical Framework:**

* **Algorithmic Foundation**:
  + **Geopolitical Risk Model**: A weighted scoring algorithm that combines dependency metrics, storage levels, LNG trends, and price volatility into a single normalized risk score.
  + **Data Normalization**: Ensures consistency across datasets, enabling accurate comparisons and visualizations.
* **Visualization Tools**:
  + Automated dashboards with time-series graphs, trend lines, and risk heatmaps.
  + Embedded visuals for seamless integration into user prompts.
* **Natural Language Processing**:
  + Enables users to interact with the system through intuitive prompts like "energy dashboard" or "geopolitical risk for France."
  + Tailors outputs based on user needs, whether it's a high-level summary or granular data-driven insights.

Continued…

**Political Risk Predictive Analytics Code:**

import pandas as pd

import numpy as np

def load\_country\_data():

"""

Simulate loading data for countries.

Replace with actual data loading from your files in GPT knowledge.

"""

data = {

"russian\_imports": pd.read\_excel("Reliance on Russian imports - July 2023.xlsx"),

"lng\_data": pd.read\_excel("LNG plot data 2024-11-07.xlsx"),

"gas\_storage": pd.read\_excel("Weekly Storage EU & UA 2024-11-28.xlsx"),

"natural\_gas\_prices": pd.read\_csv("Natural Gas Futures Historical Data (1).csv"),

}

return data

def calculate\_risk\_scores(country, data):

"""

Calculate geopolitical risk scores for a specific country based on various metrics.

"""

# Russian import dependency (scale 0–1)

russian\_dependency = data["russian\_imports"].set\_index("Country").loc[country, "Import Dependency"] / 100

# Gas storage levels (inverse risk)

avg\_storage = data["gas\_storage"].set\_index("Country").loc[country, "Average Storage Level"]

storage\_risk = 1 - avg\_storage / 100 # Higher levels reduce risk

# LNG import trends (scale 0–1)

lng\_imports = data["lng\_data"].set\_index("Country").loc[country, "Import Trend"]

lng\_risk = 1 - lng\_imports / max(data["lng\_data"]["Import Trend"]) # Normalize

# Natural gas price volatility (scale 0–1)

price\_volatility = data["natural\_gas\_prices"]["Price"].pct\_change().std()

price\_risk = min(price\_volatility / 0.05, 1) # Cap max risk for extreme volatility

# Weighted risk calculation

risk\_score = (

0.4 \* russian\_dependency

+ 0.3 \* storage\_risk

+ 0.2 \* lng\_risk

+ 0.1 \* price\_risk

)

return risk\_score

def normalize\_scores(risk\_scores):

"""

Normalize risk scores to a 0–5 scale for easier interpretation.

"""

min\_score = min(risk\_scores.values())

max\_score = max(risk\_scores.values())

normalized\_scores = {

country: 5 \* (score - min\_score) / (max\_score - min\_score) for country, score in risk\_scores.items()

}

return normalized\_scores

def generate\_country\_dashboard(country, data):

"""

Generate a dashboard for a specific country with risk metrics.

"""

russian\_dependency = data["russian\_imports"].set\_index("Country").loc[country, "Import Dependency"]

avg\_storage = data["gas\_storage"].set\_index("Country").loc[country, "Average Storage Level"]

lng\_imports = data["lng\_data"].set\_index("Country").loc[country, "Import Trend"]

price\_volatility = data["natural\_gas\_prices"]["Price"].pct\_change().std()

dashboard = f"""

\*\*Country: {country}\*\*

- Russian Import Dependency: {russian\_dependency:.2f}%

- Average Gas Storage Levels: {avg\_storage:.2f}%

- LNG Import Trends: {lng\_imports:.2f}

- Natural Gas Price Volatility: {price\_volatility:.4f}

"""

return dashboard

def main\_country\_analysis(prompt):

"""

Main function to handle country-specific geopolitical risk prompts.

"""

data = load\_country\_data()

if prompt.startswith("geopolitical risk for"):

country = prompt.replace("geopolitical risk for ", "").strip()

try:

risk\_score = calculate\_risk\_scores(country, data)

normalized\_score = normalize\_scores({country: risk\_score})[country]

dashboard = generate\_country\_dashboard(country, data)

return f"{dashboard}\n\n\*\*Risk Score (0–5): {normalized\_score:.2f}\*\*"

except KeyError:

return f"Data for {country} not found. Please check the input."

elif prompt == "country risk dashboard":

risk\_scores = {country: calculate\_risk\_scores(country, data) for country in data["russian\_imports"]["Country"]}

normalized\_scores = normalize\_scores(risk\_scores)

dashboard = "\n".join(

[f"\*\*{country}: {score:.2f}\*\*" for country, score in normalized\_scores.items()]

)

return f"\*\*Country Risk Dashboard:\*\*\n{dashboard}"

else:

return "Command not recognized. Try 'geopolitical risk for [country]' or 'country risk dashboard'."

**Historical Analysis Code:**

import pandas as pd

import matplotlib.pyplot as plt

from io import BytesIO

import base64

# Load datasets (simulated loading from files in the knowledge section)

def load\_data():

# Example placeholders for file dataframes (replace with actual loading logic)

data = {

"daily\_data": pd.read\_csv("daily\_data\_2024-11-28.csv"),

"weekly\_storage": pd.read\_excel("Weekly Storage EU & UA 2024-11-28.xlsx"),

"lng\_plot\_data": pd.read\_excel("LNG plot data 2024-11-07.xlsx"),

"natural\_gas\_prices": pd.read\_csv("Natural Gas Futures Historical Data (1).csv"),

"russian\_imports": pd.read\_excel("Reliance on Russian imports - July 2023.xlsx"),

}

return data

# Process data for the dashboard

def process\_dashboard\_data(data):

processed = {

"daily\_summary": data["daily\_data"].groupby("date").sum(),

"weekly\_storage\_trend": data["weekly\_storage"].set\_index("Week"),

"lng\_trend": data["lng\_plot\_data"].set\_index("Date"),

"price\_trend": data["natural\_gas\_prices"].set\_index("Date"),

}

return processed

# Generate dashboard visuals

def create\_dashboard(processed\_data):

fig, axes = plt.subplots(2, 2, figsize=(15, 10))

plt.tight\_layout(pad=5.0)

# Daily Summary

axes[0, 0].plot(processed\_data["daily\_summary"].index, processed\_data["daily\_summary"]["value"])

axes[0, 0].set\_title("Daily Summary")

axes[0, 0].set\_xlabel("Date")

axes[0, 0].set\_ylabel("Values")

# Weekly Storage Trend

processed\_data["weekly\_storage\_trend"].plot(ax=axes[0, 1])

axes[0, 1].set\_title("Weekly Storage Trend")

axes[0, 1].set\_xlabel("Week")

axes[0, 1].set\_ylabel("Storage Level")

# LNG Trend

processed\_data["lng\_trend"].plot(ax=axes[1, 0])

axes[1, 0].set\_title("LNG Import Trend")

axes[1, 0].set\_xlabel("Date")

axes[1, 0].set\_ylabel("Import Levels")

# Natural Gas Prices

axes[1, 1].plot(processed\_data["price\_trend"].index, processed\_data["price\_trend"]["Price"])

axes[1, 1].set\_title("Natural Gas Prices")

axes[1, 1].set\_xlabel("Date")

axes[1, 1].set\_ylabel("Price (USD)")

return fig

# Convert dashboard to a format displayable in GPT

def dashboard\_to\_gpt(fig):

buf = BytesIO()

fig.savefig(buf, format="png")

buf.seek(0)

image\_base64 = base64.b64encode(buf.read()).decode('utf-8')

buf.close()

return f"![Dashboard](data:image/png;base64,{image\_base64})"

# Quantify geopolitical risk

def geopolitical\_risk\_analysis(data):

# Example risk calculation logic

russian\_import\_dependency = data["russian\_imports"]["Import Dependency"].mean()

natural\_gas\_price\_volatility = data["natural\_gas\_prices"]["Price"].pct\_change().std()

risk\_score = (russian\_import\_dependency \* 0.6 + natural\_gas\_price\_volatility \* 0.4) \* 100

return f"Geopolitical Risk Score: {risk\_score:.2f}"

# Main function to handle dashboard and risk analysis

def main(command):

data = load\_data()

if command == "energy dashboard":

processed\_data = process\_dashboard\_data(data)

fig = create\_dashboard(processed\_data)

return dashboard\_to\_gpt(fig)

elif command == "geopolitical risk analysis":

return geopolitical\_risk\_analysis(data)

else:

return "Command not recognized. Try 'energy dashboard' or 'geopolitical risk analysis'."

**EnergyQuantGPT Development Timeline**

1. **Project Conceptualization**
   * Identified the need for a tool to analyse geopolitical risks and energy market trends.
   * Defined core functionalities, including dashboards, risk analysis, and investment insights.
   * Collected initial datasets (LNG data, gas storage levels, natural gas prices, etc.).
2. **Data Preparation**
   * Pre-processed datasets for uniformity and resolved missing data issues.
   * Integrated key metrics like Russian gas dependency, LNG import trends, and price volatility.
3. **Algorithm Development**
   * Built a geopolitical risk scoring algorithm using weighted metrics.
   * Designed a Python-based pipeline for visualizations and dashboards.
   * Normalized risk scores for easy cross-country comparisons.
4. **Integration with GPT**
   * Uploaded datasets and scripts to GPT’s knowledge base.
   * Designed and tested specific prompts for generating insights and visualizations.
   * Fine-tuned prompt behaviours for consistency and clarity.
5. **Testing and Refinement**
   * Validated outputs with historical data and use-case scenarios.
   * Refined algorithms and prompt responses based on test results.
6. **Scenario Simulation Development**
   * Added features for dynamic scenario-based analyses, like investment recommendations and policy impact assessments.
7. **Presentation Preparation**
   * Developed materials to showcase EnergyQuantGPT, including example outputs and live demonstrations.
   * Highlighted capabilities with specific use-case scenarios for different audiences.