Steering Committee

TotalEnergies Consulting Project

Group 7
École Polytechnique x Capgemini Invent
March 10, 2025

Agenda

To derive recommendations for TotalEnergy‘s customer journey, we leverage Natural Language Processing (NLP) to identify pain points across the entire customer journey

PROJECT GOAL

Assess customer relationship strategy of TotalEnergies and derive insights for each stage of customer journey

Step 1: Business analysis

Objective: Understand Total’s current business situation, including competitive landscape and customer journey

Actions:
- Perform competitive analysis
- Conduct customer journey mapping
- Identify key KPIs for customer journey
- Create customer pain point hypotheses

Step 2: Technical analysis

Objective: Leverage Natural Language Processing to identify actual pain points across customer journey

Actions:
- Data sourcing via web scraping
- Word embedding
- Topic extraction & sentiment analysis

Step 3: Synthesis & insights

Objective: Suggest insights to improve customer relationship quality based on identified pain points

Actions:
- Map identified pain points to customer journey stages & KPIs
- Derive recommendations to improve pain points
- Provide overview of potential implementation phase

In our criteria-based competitor analysis, we identified EDF and Mint Énergie to be two major competitors

Our analysis evaluates key factors influencing consumer choice in the energy sector, comparing our client, TotalEnergies, with its two main competitors: EDF and Mint Énergie

TotalEnergies

Score: 3.7
Strengths:
- Strong brand recognition (4)
- Transparency in sustainability (3)
- Decent digital features (3)

Areas to improve:
- Higher costs
- Lower customer trust

EDF

Score: 4.4 (Market leader)
Strengths:
- Lower costs
- Best brand reputation (5)
- High customer trust (4.6)

Areas to improve:
- Less contract flexibility (3)
- Average sustainability rating (3)

Mint Énergie

Score: 3.7
Strengths:
- Best sustainability rating (5)
- High contract flexibility (4)
- High customer trust (4.6)

Areas to improve:
- Low brand recognition (2)
- Higher costs

Examining each step of the customer journey at Total in detail, we built several hypotheses on potential customer pain points

Awareness

Potential customers discover the brand. Overexposure to ads can create negative effects.

Steps:
- Brand Exposure
- Brand Recall
- Word of mouth

Consideration

Potential customers evaluate the brand's offers.

Steps:
- Landing page exploration
- Competitor comparison
- Reviews & testimonials
- Consultation & inquiry

Acquisition

Customers decide to accept Total's offer.

Steps:
- Decision for Total
- Provision of customer info
- Contract acceptance
- Verification & checks
- Contract confirmation

Service

Customers interact with Total for support and issue resolution.

Steps:
- Account setup
- Billing tracking
- Usage tracking
- Customer support
- Technical support

Loyalty

Customers develop an ongoing relationship with Total.

Steps:
- Ongoing engagement
- Loyalty programs
- Contract renewals
- Feedback & satisfaction
- Complaint handling

Pain Points

- Lack of clarity on offers
- Hyper-personalized ads

- Complex navigation
- Previously hidden fees

- Waiting times & delays
- Long & redundant forms

- Mobile app limitations
- Lack of responsivity

- Over communication
- New fees

To measure performance across customer journey stages, we recommend one KPI for each individual stage and the Net Promotor Score as a measurement of overall customer satisfaction

	Awareness	Consideration	Acquisition	Service	Loyalty
Factors	Brand exposure & recognition	Effectiveness in engaging potential customers	Efficiency in converting leads into customers	Customer support quality and efficiency in resolving issues	Long-term engagement, retention
KPI Category	*Brand Awareness*	*Lead Engagement*	*Conversion Efficiency*	*Service Efficiency*	*Customer Retention*
Main KPI	**Brand Recall** • **Description:** Percentage of potential customers who remember the brand • **Reason:** Shows the effectiveness of brand awareness campaigns • **Data:** Surveys asking brand recall	**Lead Conversion Rate** • **Description:** Percentage of leads with purchase intent. • **Reason:** Measures how well marketing turns interest into potential sales. • **Data:** Tracked via CRM and analytics tools.	**L/C Conversion Rate** • **Description:** Percentage of leads with intent that become paying customers. • **Reason:** Indicates the efficiency of the acquisition process. • **Data:** Measured through sales data and CRM.	**First Contact Resolution** • **Description:** Percentage of issues resolved on first contact. • **Reason:** High FCR boosts satisfaction and reduces support costs. • **Data:** Analyzed from support ticket data.	**Customer Lifetime Value** • **Description:** Total revenue expected per customer over time. • **Reason:** Evaluates the long-term value of customer relationships. • **Data:** Calculated using sales data.
Alternatives	• Impressions • Share of Voice (SOV)	• Engagement Rate • Average Session Duration	• Sales Cycle Length • Abandonment Rate	• Average Resolution Time • Support Ticket Volume	• Retention Rate • Repeat Purchase Rate

Net Promoter Score (NPS)

- Overall comprehensive factor of satisfaction and loyalty
- Measures likelihood of customers recommend the brand
- Survey-based, using a 0-10 recommendation system

% Promoters - % Detractors

Our data collection process consisted of scratching real customer reviews from TripAdvisor, allowing us to efficiently build a large-scala dataset

Why TripAdvisor?

- It contains real, user-generated reviews from a diverse set of customers.
- Reviews offer insights into customer satisfaction, complaints, and service experiences.
- Publicly available data allows for ethical data collection.
- Provides structured metadata like review dates, ratings, and user locations, useful for trend analysis.

Why Web Scraping?

- Allows us to extract large amounts of data efficiently.
- It has no costs.
- Automates the process, ensuring scalability for multiple reviews and pages.
- Provides structured data output, enabling analysis (sentiment, trends, issue tracking).

1 Connect to TripAdvisor

- We use different proxy servers and mimic human breaks to avoid bot detection and blocks while scraping.
- The script selects a proxy randomly and connects to Tripadvisor

2 Navigate & Accept Cookies

- The scraper opens the Tripadvisor page and clicks on the cookie acceptance button.

3 Extract Number of Review Pages

- It determines how many pages of reviews exist to scrape all available data.

4 Extract Reviews

- For each review page:
- Waits for elements to load.
- Extracts title, body, rating, and date of each review.
- Saves data into a structured format (e.g., a list or JSON).

5 Navigate to Next Page & Repeat

- Clicks on the "Next Page" button until all reviews are collected.

6 Store Data for Analysis

- Saves extracted reviews into a dataset for further analysis (sentiment analysis, keyword extraction, etc.).

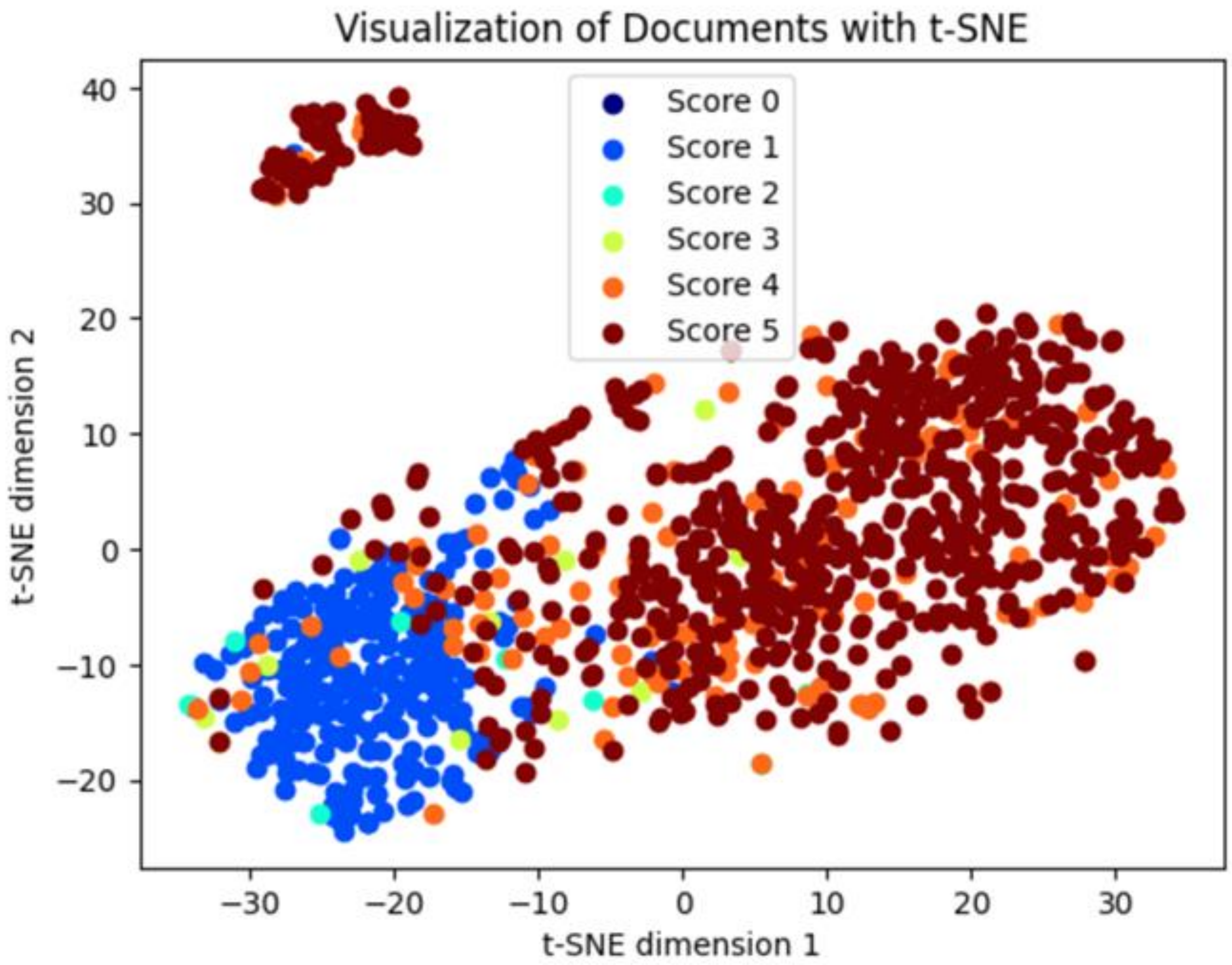
To prepare the data for the sentiment analysis, we implemented a three-step data processing methodology

Word Embedding

- **Definition**: A technique that converts words into numerical vectors so that a computer can understand their meanings.
- **How It Works**: Words with similar meanings are placed closer together in this numerical space.
- **Example**: Words like "good" and "positive" will have similar vectors, while "bad" will be far away.

	I: Preprocessing	II: TF-IDF	III: Word2Vec
INPUTS	**Steps:** • **Text Cleaning:** Removed unnecessary elements like punctuation, special characters, and stopwords (e.g., "and," "the," "a"). • **Tokenization:** Split the text into individual words, transforming each review into a list of terms. • **Lemmatization**: Reduced words to their root form to ensure consistency and improve analysis accuracy.	**How It Works:** • **Term Frequency (TF):** Measures how often a word appears in a document. • **Inverse Document Frequency (IDF):** Weighs the word's uniqueness across all documents. • The product of TF and IDF gives the TF-IDF score, highlighting important words that are not too common. **Top words with high relevance**: "service," "expérience," "client," "tarif," "prix".	**What is Word2Vec?** • A machine learning model that learns word associations based on their context in sentences. • Words with similar meanings have similar vector representations. **Our Approach:** • Trained the model on cleaned customer reviews. • Generated word vectors that allowed us to find words with similar meanings.
Goal	Prepare raw customer reviews for analysis by cleaning and standardizing text data.	Identify important words in the reviews based on relevance.	Convert words into numerical vectors to analyze meanings and relationships.

Leveraging t-SNE, we visualized the result of our embedding process



In the first step of the modeling, we leveraged LDA to find important topics in customer reviews

Objective

Identify key topics within customer reviews related to energy suppliers using Latent Dirichlet Allocation (LDA).

Methodology

LDA Model Application:

- Applied Latent Dirichlet Allocation (LDA), a probabilistic model that assigns topics to reviews based on word co-occurrence patterns.
- Generated 10 distinct topics, each defined by a set of weighted keywords.
- Keywords with higher weights have greater influence in defining the topic.

Example

Topic 0 (Customer Support & Contracts):
Keywords: *"conseil" (advice), "téléphon" (phone), "contrat" (contract), "expliqu" (explain)*.

Output

- 10 categorized topics, each providing insights into different aspects of customer feedback.
- Structured word importance scores, enabling further clustering and analysis.

To simplify the analysis, we grouped the 10 identified topics into three clusters

Objective

Simplify analysis by grouping the 10 topics into 3 main clusters

Methodology

Examined the meaning and similarities of the topics. Grouped topics with overlapping themes into comprehensive clusters.

Cluster 1: Customer Service & Reviews (Topics 0, 1, 3, 7) - Focus: Interactions with customer support, service quality.
Example Keywords: "professionnel" (professional), "avis" (reviews), "conseil" (advice), "écout" (listening).

Cluster 2: Energy & Billing (Topics 2, 4, 9) - Focus: Billing issues, contract management, pricing concerns.
Example Keywords: "factur" (billing), "énerg" (energy), "prix" (price), "contrat" (contract).

Cluster 3: Projects & Technical Aspects (Topics 5, 6, 8). Focus: Efficiency, installation, solar energy, technical services.
Example Keywords: "solair" (solar), "technicien" (technician), "install" (installation), "rapid" (fast).

Example

Topic 0 (Customer Service) and Topic 7 (Professionalism) were grouped into Cluster 1, as both discuss customer interactions and advice from service representatives.

Output

- Mapped the 10 topics into 3 broad clusters.
- Simplified insights for business decisions, such as identifying areas for customer service improvement or pricing adjustments.

Using a language model, we prepared data for the final sentiment analysis

Objective

Set up the data for topic classification and sentiment analysis using a language model

Methodology

Review Processing:
- Sampled 1000 reviews for analysis.
- Used preprocessed text (meaningful words only).

Classification Process:
- Designed an LLM-based classification system to:
- Assign a topic to each review based on content similarity.
- Extract the most relevant words from the review to describe its key theme.

Example

Review: "*The billing system is unclear, and I was overcharged for my energy usage.*"
Topic: Energy & Billing; **Key Words:** billing-unclear-overcharged-energy: **Sentiment**: Negative

Output

A dataset with:
- Each review's assigned topic.
- Extracted key words summarizing the review.
- Sentiment classification (positive, neutral, negative).

We created a pivot table breaking down the sentiment word-by-word

Objective

Prepare data to visualize sentiment distribution by key topic words

Methodology

Data Exploding & Transformation:
- Split multi-word key phrases into individual words.
- Each word was assigned a separate row to allow granular analysis.

Sentiment Frequency Calculation:
- Grouped data by word and sentiment classification.
- Counted the number of positive, negative, and neutral occurrences for each word.

Pivot Table Construction:
- Compared sentiment counts for each word side by side..

Example

Word: "service"
- Positive Sentiments: 57
- Neutral Sentiments: 41
- Negative Sentiments: 20

Output

- Pivot Table Showing Sentiment Breakdown by Word:
- The table helps quickly identify pain points and satisfaction drivers, allowing targeted action to improve customer experience.

The final output of the sentiment analysis is a table displaying the word count by sentiment

sentiment topic_word_list	negative	neutral	positive	total
service	4.0	5.0	2.0	11.0
prix	3.0	5.0	1.0	9.0
client	2.0	5.0	1.0	8.0
nan	0.0	7.0	0.0	7.0
réponse	0.0	4.0	2.0	6.0
date	0.0	6.0	0.0	6.0
expérience	0.0	6.0	0.0	6.0
consommation	2.0	2.0	1.0	5.0
facture	1.0	4.0	0.0	5.0

Based on the sentiment analysis and prior hypotheses, we determined four major areas of focus for our recommendations

	Improve Customer Service & Billing	Address Pricing Transparency	Strenghten Customer Engagement & Loyalty	Capitalize on Strengths: Speed & Efficiency
Sentiment	High negative sentiment for „service“, „client“, and „facture“	Negative sentiment around “euro” and “payer”	Mixed sentiment on customer experience	Positive sentiment around “rapide”, “efficace”, ”simple”
Potential Cause	Slow response times, unclear billing details, lack of self-operations	Perceived lack of pricing clarity, affordability concerns	Lack of personalized engagement, unclear retention strategies	Customers appreciate fast service
Customer Journey Stage	**Service**	**Consideration & Acquisition**	**Loyalty**	**Acquisition**
Recommendations	• Enhance support response time and efficiency. • Proactive billing resolution & clear communication on charges. • Expand self-service features to handle payments and inquiries.	• Introduce flexible payment plans to ease financial pressure. • Improve price breakdown communication to enhance customer understanding. • Implement loyalty discounts for long-term customers.	• Launch NPS-driven improvement programs based on feedback. • Provide personalized recommendations to enhance engagement. • Encourage customer testimonials & referrals to boost positive sentiment.	• Use marketing campaigns to emphasize speed & efficiency. • Showcase customer testimonials that reinforce quick service.
KPI to monitor	• First Contact Resolution	• Lead Conversion Rate • L/C Conversion Rate	• Customer Lifetime Value • Net Promoter Score	• L/C Conversion Rate

For the implementation phase, we suggest a structured four step process with progressive improvements and constant KPI monitoring

Month 1

Assessment & Planning

- Conduct deep-dive analysis on sentiment insights.
- Identify critical gaps in service, billing, pricing, and engagement.
- Align KPIs with customer journey stages.

Low ($)

Month 2-3

Quick Wins

- Improve billing communication & customer service scripts.
- Launch marketing messages on efficiency & speed.
- Implement initial self-service options.

Medium ($$)

Month 4-6

Core Enhancements

- Introduce automated billing resolution tools.
- Develop flexible payment plans & loyalty discounts.
- Expand customer engagement strategies (NPS-driven updates, feedback loop).

High ($$$)

Month 6+

Deployment & Monitoring

- Fully integrate enhanced support response system.
- Scale personalized recommendations & pricing transparency initiatives.
- Track performance using KPIs (NPS, FCR, CLV, Conversion Rates).

Medium-High ($$-$$$)

Thank you!

Do you have questions?

APPENDIX

To identify the key drivers in consumers' energy provider choice, we determined three dimensions of decision-making factors along which we identified specific influences

35%	Cost (Pricing)
10%	Contract Terms
25%	Brand Recognition
10%	Commitment to Environmental Practices
10%	Reviews & Word-of-Mouth Recommendations
5%	Sign-Up Simplicity
5%	Digital Features

Functional Factors

- Objective elements that define the affordability and practicality of the service.
- Consumers seek transparency, predictability, and flexibility to reduce their energy costs and manage consumption effectively.

Emotional Factors

- Subjective elements that influence consumer perception, trust, and alignment with personal values.
- Consumers align with brands that reflect their values, especially sustainability and social responsibility.
- Strong reputation and positive reviews build credibility and reduce switching hesitation.

Convenience Factors

- Efforts required to start, manage, and maintain a relationship with the provider.
- A simple sign-up process reduces friction in switching.
- Modern digital tools improve customer control, engagement, and overall user experience.

In selecting the five key competitors of TotalEnergies in the B2C distribution sector, we focused on major market presence, disruptive innovation, and differentiation strategies

Engie

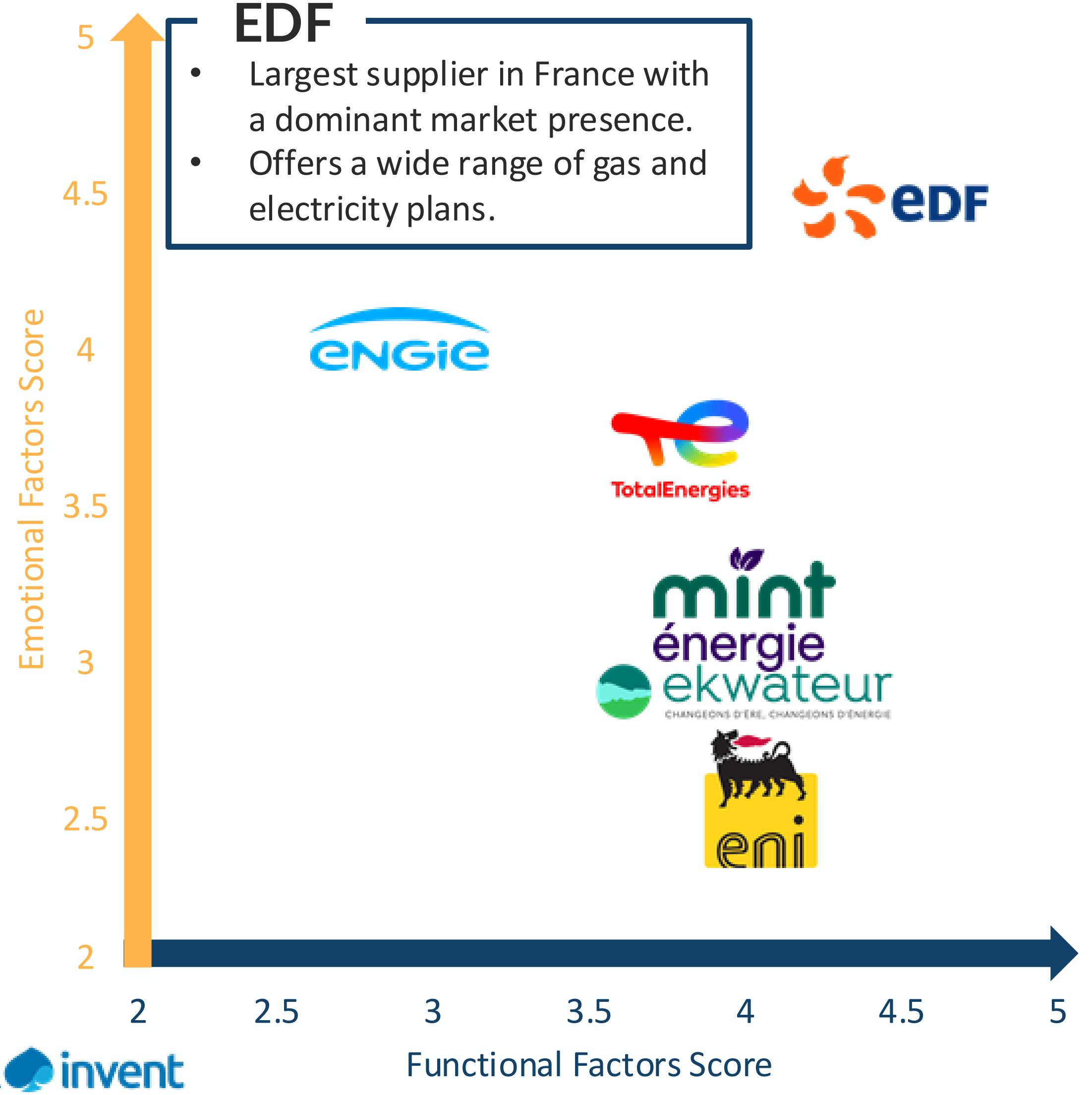
- Strong commitment to renewable energy and competitive pricing.
- Extensive gas and electricity offerings.

Eni

- Low-cost with a focus on carbon-neutral energy.
- Straightforward contracts and growing market presence.
- Appeals to price-conscious and eco-friendly consumers.

EDF

- Largest supplier in France with a dominant market presence.
- Offers a wide range of gas and electricity plans.



Mint Énergie

- Affordable, straightforward energy plans with a digital-first approach.
- Offers budget-friendly green energy options.
- Appeals to cost-conscious, eco-minded consumers.

ekWateur

- 100% renewable energy supplier, transparent pricing.
- Strong sustainability focus.
- Appeals to younger, environmentally-aware demographics.

INSTITUT POLYTECHNIQUE DE PARIS Capgemini invent

Each competitor was scored using a weighted average over the previously identified criteria, highlighting EDF as the strongest player in the French B2C energy market

Company	Score	Cost	Contract Terms	Brand Recognition	Commitment to Environmental Practices	Reviews (Trustpilot)	Sign-up Simplicity	Digital Tools & Features
TotalEnergies	3.7	4 661 €/y	3.5	4	3 Promotes sustainable development and energy efficiency	3.3	4	3 Website for billing and consumption tracking
EDF	4.4	5 645 €/y	3 1-year contract, cancel anytime	5	3 Net-zero by 2050, CO2 reduction, circular economy initiatives	4.6	3	4 Mobile app to track energy usage and manage bills
mint énergie	3.7	4 663 €/y	4 Flexible contracts, terminable anytime	2	5 100% renewable energy, carbon offset and monitoring	4.6	5	3 Website for billing and consumption tracking
ekwateur	3.6	4 667 €/y	4 Indefinite contract, cancel anytime	2	5 100% renewable energy, carbon monitoring	4.0	5	3 Website for green energy management and tracking
ENGIE	3.5	3 692 €/y	3 1-year contract, cancel anytime	4	4 Net-zero by 2045, biodiversity preservation, sustainable resources	3.8	4	4 Mobile app to track energy usage and manage bills
eni	3.3	4	4 Flexible contracts, terminable anytime	3	3 ISO 14001 certified, advanced HSE systems, environmental focus	1.1	3	4 Mobile app to track energy usage and manage bills

*Data collected through simulating the process of asking an electricity supply contract and comparing available options on platforms like energie.selectra.info. Data based on an average house for 1 or 2 people in the 1st arrondissement, postal code 75001. Average consumption: 2500 kWh/year. Sources:
-energie.selectra.info/
-mint-energie.com/Pages/Informations/nos-engagements
-totalenergies.com/sustainability/our-approach/esg-documentation
-eni.com/en-IT/sustainability/
-edf.fr/en/the-edf-group/taking-action-as-a-responsible-company/corporate-social-responsibility/
-ekwateur.fr/nos-engagements/
-engie.com/sites/default/files/assets/documents/2022-07/Environmental%20policy.pdf

Web Scraping I

```python
for url in urls:
    proxy_host, proxy_port, username, password = random.choice(proxy_list)
    print(f"\n Using Proxy: {proxy_host}:{proxy_port} for {url}")

    driver = get_chrome_driver(proxy_host, proxy_port, username, password)
    driver.get(url)
    # time.sleep(random.uniform(0.2, 0.8))
    cookie_button = driver.find_element(By.XPATH, "//*[@id='onetrust-accept-btn-handler']")
    cookie_button.click()
    total_pages = int(driver.find_element(By.XPATH, "//*[@id='__next']/div/div/main/div/div[4]/section/div[26]/nav/a[4]/span").text)

    for page in range(1, total_pages + 1):
        print(f"Scraping page {page} of {total_pages}")

        WebDriverWait(driver, 10).until(
            EC.presence_of_element_located((By.XPATH, "//*[@id='__next']/div/div/main/div/div[4]/section/div/article/div/section"))
        )

        # time.sleep(random.uniform(0.2, 0.8))

        comments = driver.find_elements(By.XPATH, "//*[@id='__next']/div/div/main/div/div[4]/section/div/article/div/section")
```

Web Scraping II

```python
for i, comment in enumerate(comments, start=1):
    try:
        title = comment.find_element(By.XPATH, "./div[2]/a/h2").text if comment.find_elements(By.XPATH, "./div[2]/a/h2") else "No title"
        body = comment.find_element(By.XPATH, "./div[2]/p[1]").text if comment.find_elements(By.XPATH, "./div[2]/p[1]") else "No body"
        date = comment.find_element(By.XPATH, "./div[1]/div[2]/time").text if comment.find_elements(By.XPATH, "./div[1]/div[2]/time") else "No date"
        note_element = comment.find_elements(By.XPATH, "./div[1]/div[1]/img")

        note = note_element[0].get_attribute("alt")[5] if note_element else "No rating"

        reviews.append({
            "url": url,
            "proxy": proxy_host,
            "comment": f"comment {i} (page {page})",
            "title": title,
            "body": body,
            "date": date,
            "note": note
        })

    except Exception as e:
        print(f"Error processing comment {i} on page {page}: {e}")
```

Web Scraping III

```python
        if page < total_pages:
            try:
                next_page_button = WebDriverWait(driver, 10).until(
                    EC.element_to_be_clickable((By.XPATH, "//*[@id='__next']/div/div/main/div/div[4]/section/div[26]/nav/a[5]"))
                )

                driver.execute_script("arguments[0].scrollIntoView();", next_page_button)
                driver.execute_script("arguments[0].click();", next_page_button)
                # time.sleep(random.uniform(0.2, 0.8))

            except Exception as e:
                print(f"No more pages or error: {e}")
                break

    driver.quit()

print(f"\nTotal Reviews Extracted: {len(reviews)}")
```

Data Preprocessing

```
# Afficher un exemple avant et après tokenisation et lemmatisation
example_before = df_2["text"].iloc[0]  # Prend le premier commentaire
example_after = tokenize_lemm_func(example_before)  # Applique la fonction pour voir la différence

print("Exemple avant nettoyage :\n", example_before)
print("\nExemple après tokenisation et lemmatisation :\n", example_after)
```

```
Exemple avant nettoyage :
 Merci Octopus pour vos tarifs et pour vos récompenses lors des éco-sessions. Pas déçu, depuis plus d'un an maintenant. Société sérieuse.

Exemple après tokenisation et lemmatisation :
 Octopus tarif récompense déçu an Société sérieux
```

TF-IDF Weight Calculation

```
vectorizer = TfidfVectorizer()
tfidf_matrix = vectorizer.fit_transform(corpus)
# On converti la matrice TF-IDF en tableau dense
tfidf_array = tfidf_matrix.toarray()

# On calcule la moyenne des scores TF-IDF pour chaque mot
average_scores = np.mean(tfidf_array, axis=0)

# On obtient les noms des mots
feature_names = vectorizer.get_feature_names_out()

# On crée un DataFrame associant les mots et leurs scores moyens
df_tfidf_scores = pd.DataFrame({
    'Mot': feature_names,
    'Score_TF_IDF_Moyen': average_scores
})

# Nous trions les mots par score décroissant et on affiche les 20 premiers
top_20_words = df_tfidf_scores.sort_values(by='Score_TF_IDF_Moyen', ascending=False).head(20)

print("Les 20 mots les plus importants selon les scores TF-IDF :")
print(top_20_words)
```

```
Les 20 mots les plus importants selon les scores TF-IDF :
                Mot  Score_TF_IDF_Moyen
9858        service            0.030168
4828     expérience            0.029031
2767         client            0.028972
2213            bon            0.028608
2144          bien            0.027002
3532           date            0.026318
5170    fournisseur            0.019729
11417          être            0.019663
3252        contrat            0.019176
4882        facture            0.018703
10406         tarif            0.017864
11304       énergie            0.017721
2738          clair            0.017076
8355           prix            0.017060
8808         rapide            0.016948
3165   consommation            0.016179
7532        octopus            0.016039
4898          faire            0.015317
4703           euro            0.015184
3121     conseiller            0.014733
```

Word Embedding with World2Vec

```
import gensim
from gensim.models import Word2Vec

# We repare the data for Word2Vec
sentences = df_cleaned['text'].apply(lambda x: x.split(' ')).values

# We training the Word2Vec model
model = Word2Vec(sentences, vector_size=256, window=5, min_count=1, sg=1)

# Example of how to use the model
word_vectors = model.wv
print(word_vectors.similar_by_word('mauvais'))
```

```
[('faite', 0.8243527412414551), ('apprendre', 0.8202271461486816), ('vue', 0.820162832736969), ('incompréhensible',
0.819408655166626), ('quasi', 0.8088821172714233), ('déception', 0.807542622089386), ('.....', 0.8068920373916626),
('valable', 0.8057442903518677), ('fiable', 0.805336058139801), ('complètement', 0.8047224879264832)]
```

```
import gensim
from gensim.models import Word2Vec

# We repare the data for Word2Vec
sentences = df_cleaned['text'].apply(lambda x: x.split(' ')).values

# We training the Word2Vec model
model = Word2Vec(sentences, vector_size=256, window=5, min_count=1, sg=1)

# Example of how to use the model
word_vectors = model.wv
print(word_vectors.similar_by_word('positif'))
```

```
[('octopus', 0.9348611831665039), ('partager', 0.9283393621444702), ('souligner', 0.9206451773643494), ('satisfaisa
nt', 0.9197130799293518), ('Bon', 0.9190029501914978), ('recommande', 0.9188801050186157), ('adhésion', 0.916264891
6244507), ('trer', 0.9112939834594727), ('concevoir', 0.910823404788971), ('hyper', 0.9105439782142639)]
```

END OF APPENDIX