MGMT 529 MARKETING CONSULTING PROJECT

Megaputer Natural Language Processing

Submitted by
Gautham Shankar Muthukumar
Bhawishya Juneja
Devanshi Parekh
Shubhi Srivastava
Mansi Swami
Suqi Zhuo

Table of Contents

Introduction **2**

Methodology **4**

A. Competitive Analysis 4

B. Data Scraping 4

C. Topic Modeling 4

D. Tools Used 4

Data Collection and Analysis **5**

Objective A: Blog recommendations 4

Steps Taken 4

Tableau Data Visualization 9

Challenges in Deriving Meaningful Insights 10

Timeline Demonstration 12

Objective B- User Pain Point Analysis 12

Steps Taken 13

Tableau Data Visualizations 16

Challenges in Deriving Meaningful Insights 13

Timeline Demonstration 18

Recommendations **18**

Blog Strategy 14

Final Recommendations 21

Consumer Pain Points 23

Final Recommendations 25

Appendix 29

Introduction

The project aimed to conduct a comprehensive competitor analysis using text analysis techniques for Megaputer, focusing on their flagship product, PolyAnalyst, a single-stack solution for data extraction and cleaning, text analysis, and report building. Initially, the objective was to devise a content strategy to align PolyAnalyst with industry competitors. In addition, Megaputer also wants to benchmark the performance of PolyAnalyst against other open-source tools.

The project commenced with meticulous manual research to identify pertinent competitors, including Google Cloud, AWS, Alteryx, SAAS, Microsoft Azure, and IBM. Subsequently, efforts were directed toward extracting relevant content from competitors' websites, primarily focusing on their text analysis categories. The most efficient approach involved leveraging open-source Python tools for web scraping. To accommodate the diverse website structures of each competitor, a tailored approach was adopted, beginning with scraping HTML links from pertinent blog pages. This was followed by the development of Python scripts, utilizing libraries such as Beautiful Soup, to extract blog content in uniform formats like text and CSV files.

Following data collection, a centralized cleaning process, using Python libraries was implemented to eliminate common words that could skew text analysis results. Approximately 600-700 blogs were processed across all competitors. Topic modeling emerged as the preferred method for data analysis, leading to the selection and implementation of Genesis and Scimitar libraries after iterative comparisons. Furthermore, Chat GPT was utilized to interpret and categorize the top 10-15 topics for each competitor, facilitating the creation of a consolidated report outlining common themes and trends.

The project scope expanded to include understanding consumer pain points among competitors. As Megaputer undergoes a strategic transformation, shifting its focus towards B2C markets while maintaining its presence in the enterprise sector, the client shared that it needs to get an understanding of various customer pain points among its competitors. This will enable them to comprehend key differentiating factors to remarket Polyanalyst to end consumers who are engaging with similar products on a day-to-day basis.

In this context, we did a comprehensive competitor analysis to uncover areas for improvement and innovation opportunities. This objective not only equips us with valuable insights into market trends but also empowers us to strategically position Poly Analyst to better address user needs and preferences. Channels for consumer feedback, such as third-party review websites, social media platforms, and communities, were researched. Due to API constraints, a focus was placed on scraping data from credible sources like G2 and Gartner.

Specific competitors within the text analysis category were identified, and data cleaning was performed to isolate negative reviews. Topic modeling was then applied to analyze feedback data using the Genesis and Scimitar libraries. Chat GPT was employed once again to interpret pain points, resulting in the creation of a centralized report on common feedback and reviews. Despite encountering challenges such as website structure variations and API limitations, the project successfully generated valuable insights for refining PolyAnalyst's content strategy and understanding consumer sentiments within the competitive landscape.

Methodology

A.Competitive Analysis

After gaining a thorough understanding of Polyanalyst's product capabilities, we conducted an extensive competitive analysis to evaluate its competitors. Utilizing sources such as Gartner, G2, and Google Search, we identified major competitors.

B.Data Scraping

We employed a combination of JSON scripts, open-source Python libraries, and browser extensions (discussed in detail in the next section) to scrape content from competitor websites and third-party review platforms like G2. This data formed the foundation for our subsequent analyses.

C.Topic Modeling

To uncover common themes and topics within blog content and user reviews, we applied topic modeling techniques, specifically Latent Dirichlet Allocation (LDA). This involved processing the text data, creating a document-term matrix, and applying LDA to extract meaningful topics.

D.Tools Used

a. Json Script: Used for extracting URLs from various competitor blog pages.

b. Python Libraries:

- NLTK (Natural Language Toolkit): Employed for natural language processing tasks such as tokenization and stop word removal.
- BeautifulSoup: Utilized for parsing HTML and XML documents, facilitating data extraction from web pages.

- Scikit-learn: A powerful machine learning library utilized for text processing and topic modeling.
- Gensim: A topic modeling library enabling the creation of semantic representations of documents using statistical methods.

c. G2 Scraping Extension: Leveraged to gather user reviews and insights from the G2 website.

d. Gen AI: Utilized for interpreting results from the topic model.

This comprehensive methodology enabled us to gain valuable insights into the competitive landscape surrounding Polyanalyst, providing a solid foundation for our subsequent analyses and recommendations.

Data Collection and Analysis

Objective A: Blog Recommendations

This section of the project aimed to develop a blog content strategy for Megaputer's PolyAnalyst by analyzing competitor blogs. By understanding the content focus of Megaputer's direct competitors, we can identify potential gaps and opportunities to tailor PolyAnalyst's blog content for a more targeted and effective reach.

Steps Taken

Step 1 - Competitor Identification

We analyzed Polyanalyst's competitors by going to G2 and searching for text analysis products.

- PolyAnalyst, a single-stack solution for data manipulation and analysis, was chosen as the product of focus.
- Direct competitors, rather than point solutions, were prioritized.

- Third-party platforms like G2 and Gartner were used for competitor identification.
- Client confirmation solidified the final list of competitors:
    - Alteryx
    - KNIME
    - AWS Comprehend
    - IBM Watson
    - SAS Viya
    - Google Cloud NLP
    - Microsoft Azure NLP

Step 2 - Competitor Blog URL Scraping

- A JavaScript (JS) script was created to efficiently extract blog URLs from competitor websites.
- Expertise in coding ensured accurate URL capture.
- The script generated a filtered JSON file containing the URLs for the first 100 blog entries from each competitor's blog pages.
- **GitHub Access to files**

Step 3 - Blog Text Scraping

- The BeautifulSoup library was utilized to scrape text content from the extracted URLs.
- This involved accessing the HTML structure of the blog pages and isolating the relevant textual data.
- Scraped blog text in ".txt" format was saved in our system's designated directory to enable text analysis.
- **GitHub Access to files**

Step 4 - Text Analysis

- Text cleaning techniques were applied to the scraped blog content.
- NLTK library facilitated the removal of stop words, refining the data for analysis.
- Keyword frequency analysis identified the most and least frequently used words across the blogs.
- Separate CSV/Excel files were generated for each competitor, showcasing the top 100 keywords and their frequency of use.
- To reduce the number of operations, we combined the code for blog text scraping & text analysis
- This data was further used to provide additional insights, such as keyword frequency, etc. However, it failed to provide scrapable and clear data for topic modeling.
- GitHub Access to files

Step 5 - Topic Modeling

- Topic modeling employed Scikit-learn and Gensim libraries to scrape information from the blog text files saved in our directories.
- Gensim generated better results for Microsoft, KNIME, & Alteryx, while SAS VIYA, AWS, IBM Watson, & Google got better results using the Scikit-Learn library.
- These libraries enabled text processing, document-term matrix creation, and application of Latent Dirichlet Allocation (LDA) to uncover thematic clusters within the blog content.
- A total of 15 topics were generated for each competitor.
- GitHub Access to files

Step 6 - Topic Interpretation

- Gen AI tools were used to interpret the topics generated by the LDA model.

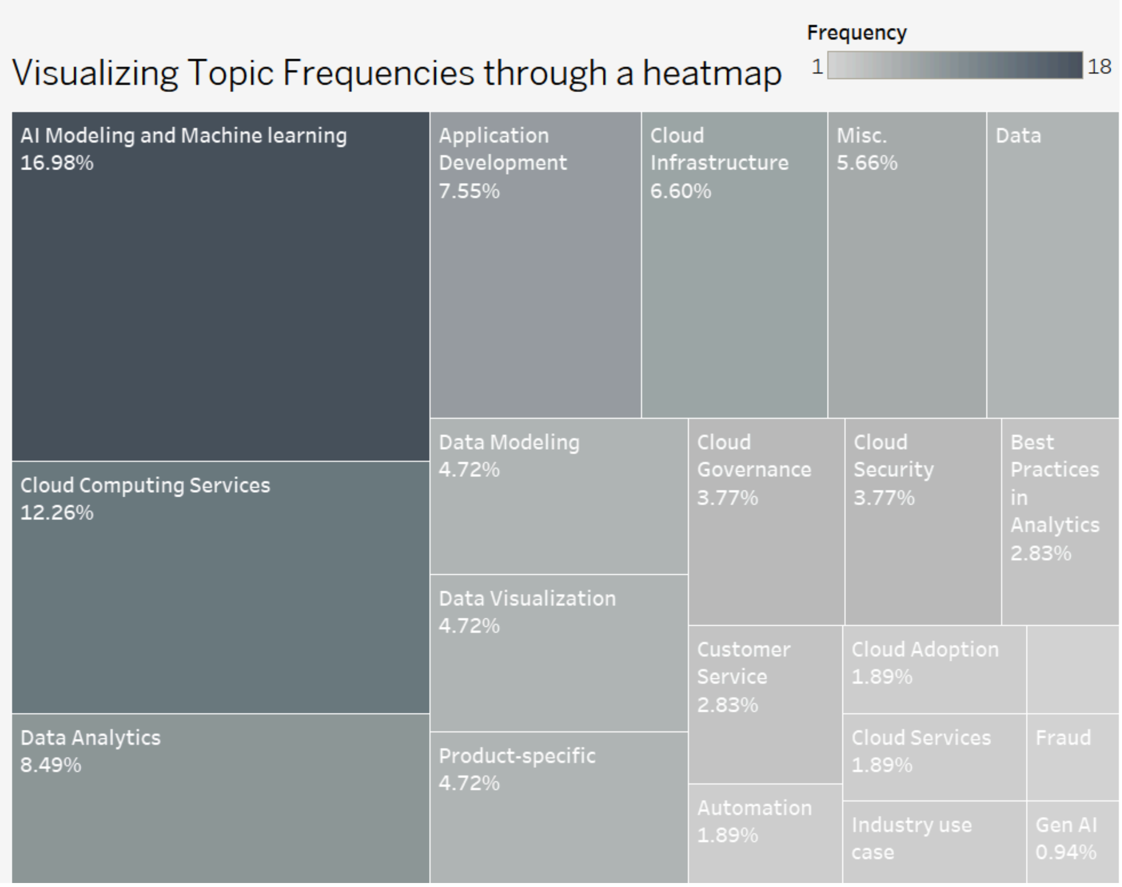
- Manual categorization of topics was conducted based on our understanding of the content and context of the blogs under multiple categories such as industry use cases, informative blogs, testimonials, etc.
- Topic Modeling Results (Excel)

Step 7 - Visualization

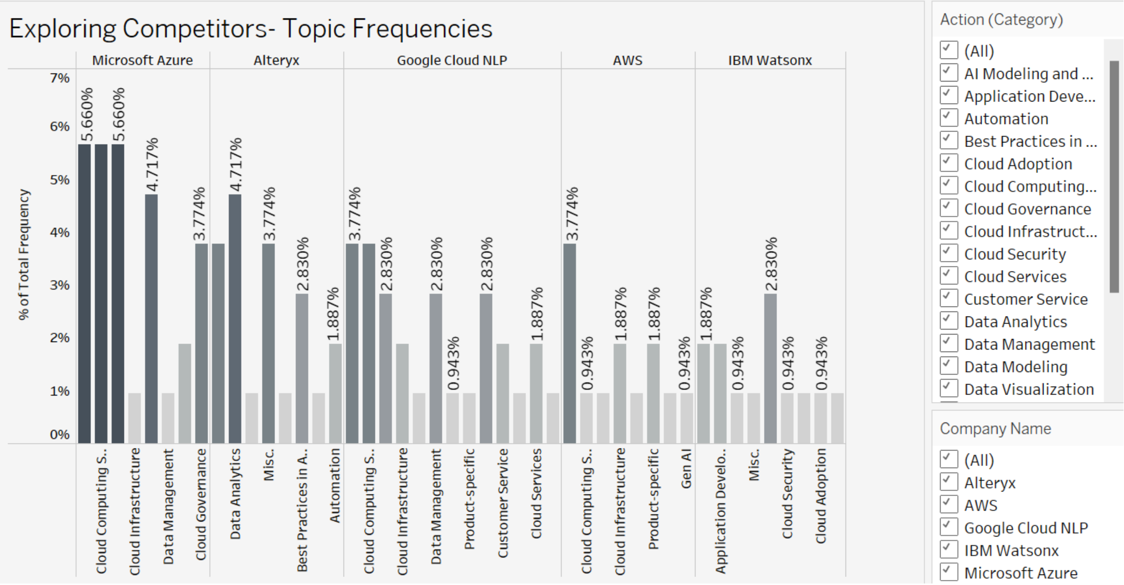
- Tableau and Excel were leveraged to create visualizations that effectively communicate the analysis findings.
- Visualization was done in a two-way format- visualizing all competitor blogs under overarching categories, and second, visualizing the findings within a competitor blog itself to provide a comprehensive overview of the blog content strategy landscape.

Tableau Data Visualization

Blog Recommendation: Topic Frequencies, [Link to Dashboard]



Competitor-wise - Topic Frequencies, [Link to Dashboard]



Action (Category)

☑ (All)
☑ AI Modeling and ...
☑ Application Deve...
☑ Automation
☑ Best Practices in ...
☑ Cloud Adoption
☑ Cloud Computing...
☑ Cloud Governance
☑ Cloud Infrastruct...
☑ Cloud Security
☑ Cloud Services
☑ Customer Service
☑ Data Analytics
☑ Data Management
☑ Data Modeling
☑ Data Visualization

Company Name

☑ (All)
☑ Alteryx
☑ AWS
☑ Google Cloud NLP
☑ IBM Watsonx
☑ Microsoft Azure

By analyzing competitor blogs, we gained valuable insights into their content focus. This data will be instrumental in developing a targeted blog content strategy for Megaputer's PolyAnalyst. The strategy will address potential gaps in competitor content and position PolyAnalyst as a leader in the market through its unique value proposition.

Challenges in Deriving Meaningful Insights

Keyword Analysis: Despite our efforts to analyze keywords extracted from competitor blog URLs, aiming to identify prevalent and rare terms after eliminating common stop-words, the results yielded excessively generic terms like 'data' and 'analytics.' These findings lacked the depth of insight we were seeking, leading us to discontinue this approach in favor of focusing more on topic modeling.

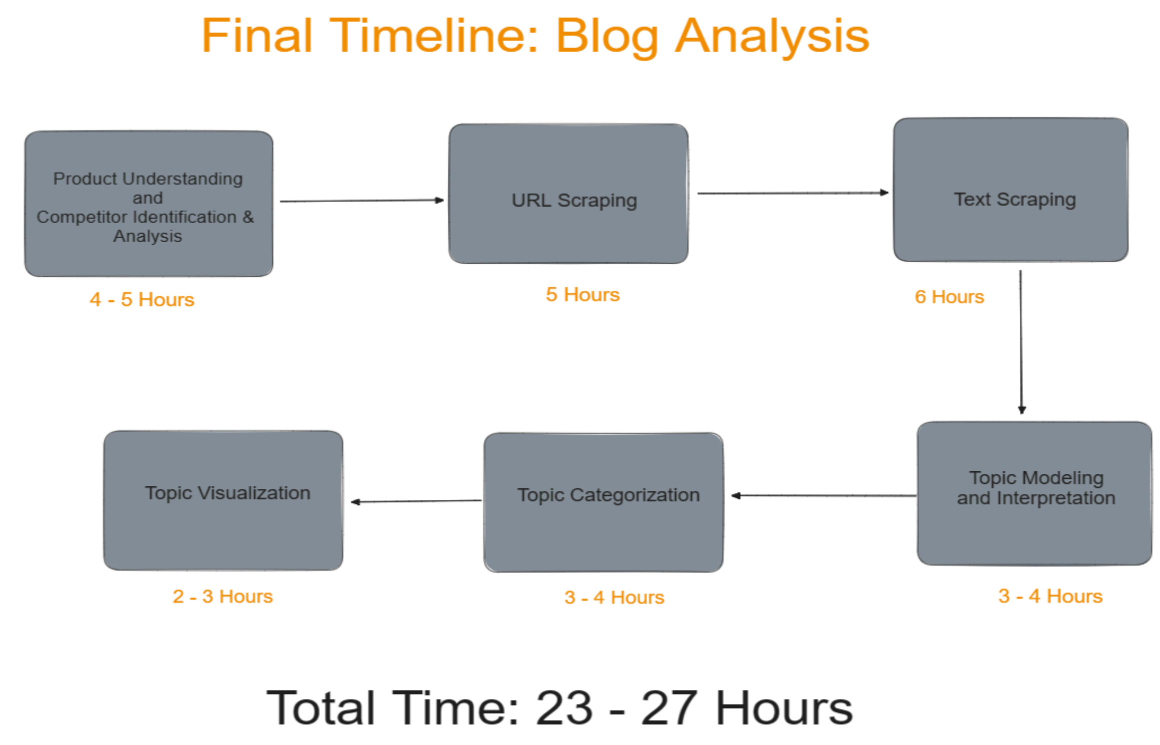
Resolvable Challenges

- URL Scraping: With each competitor having a different HTML structure, the JSON skeleton code needed hours of work and expertise from computer science professionals.
- Text Scraping: The dynamic nature of competitors' websites required ongoing development efforts to ensure the Python codes remained effective.
- Topic Modeling: For some competitors, the Scikit library generated better results, while for others the Gensim library worked better. Hence, two sets of code were required to be run for each competitor to get the best results possible.

Competitor-Wise Challenges

- Microsoft Azure: Some codes struggled to scrape keywords from all 100 URL links. For example, the code for Microsoft only scraped 77 out of 100 extracted URL links. To compensate, we increased the number of extracted links to 150.
- SAS Viya: Selecting specific categories for blog analysis, such as "Advanced Analytics" for SAS, posed challenges due to the lack of language identification in SAS syntax or blog posts. We addressed this by utilizing the website's blog library to extract relevant URLs.
- Google Cloud: Google was the competitor 0, on which we began the scraping. Our initial roadblock was getting the URLs for blogs in our directories, which Python alone did not achieve. After more trials, we were able to scrape the URLs through the JS console code and eventually build more on blog scraping using Python, and this is how a skeletal code was built for all competitors to follow.
- Alteryx: Permission errors and irrelevant keywords were encountered when scraping Alteryx blogs.
- Amazon AWS: Our initial attempt at using JavaScript to scrape blog URLs failed. We suspected that the website's structure may have been the cause. To work around this issue, we developed a Python code to extract the top 100 URLs and saved them in a text file.
- IBM Watson: For blog scraping, WatsonX did not have a sufficient number of blogs for scraping. To ensure an adequate dataset, we broadened our scope to include all IBM blogs, not just those specifically related to Watson.

Timeline Demonstration


Final Timeline: Blog Analysis
Product Understanding and Competitor Identification & Analysis
4 - 5 Hours
URL Scraping
5 Hours
Text Scraping
6 Hours
Topic Modeling and Interpretation
3 - 4 Hours
Topic Categorization
3 - 4 Hours
Topic Visualization
2 - 3 Hours
Total Time: 23 - 27 Hours

Objective B- User Pain Point Analysis

As Megaputer undergoes a strategic transformation, shifting its focus towards B2C markets while maintaining its presence in the enterprise sector, the client shared that it needs to get an understanding of various customer pain points among its competitors. This will enable them to comprehend key differentiating factors to remarket Polyanalyst to the end consumers who are engaging with similar products on a day-to-day basis.

In this context, we did a comprehensive competitor analysis to uncover areas for improvement and innovation opportunities. This objective not only equips us with valuable insights into market trends but also empowers us to strategically position Poly Analyst to better address user needs and preferences.

Steps Taken

Step 1- Data Collection

We identified three primary sources for collecting customer feedback:

- **Social Listening:** This involved monitoring discussions, reviews, and mentions of Poly Analyst and its competitors on platforms such as Reddit and Twitter. We also explored the option of capturing YouTube reviews by popular influencers/ users in the analytics arena to get personalized opinions on multiple competitors.
- **Third-Party Analysts:** We identified that leveraging insights from reputable sources like Gartner and G2 to understand market trends and customer perceptions could be the best way to understand the needs of end users.
- **Developers Community:** Engaging with developer communities on platforms such as Google and AWS would help us gather technical feedback and insights.

Our approaches for gathering data from these sources are as follows:

- **Social Listening:** Deploying Python scripts using PRAW and Tweepy libraries (coupled with Twitter APIs) to scrape data from Reddit and Twitter to extract discussions, reviews, and sentiments related to Poly Analyst and competing tools. For YT, we explored transcription using AI tools such as Notta to convert the audio into transcribed files for further text analysis.

- **Third-Party Analysts:** Exploring APIs (Scrapefly, DataShake) or browser extensions provided by Gartner and G2 to extract reviews and ratings. If APIs are not available, we utilize web scraping techniques with BeautifulSoup or Scrapy.
- **Developers Community:** Participating in discussions and forums on Google and AWS platforms to gather insights from developer communities. Later, Python could be utilized for web scraping to collect valuable data.

Step 2- Text Analysis

a. Reviews Scraping

- We explored all methodologies and identified G2 as the main source for reviews, based on clarity of customer dislikes and the precision of the scraped data.
- For extraction methodology, we identified that the Chrome extension- “Scrape G2 Reviews & Download to Excel” chrome extension from extractg2.com was able to create downloadable CSV files of the reviews for multiple competitors on G2 to our working directories.
- We had to manually scroll through the 2-3 pages of reviews, download the separate CSV files for each page, and then later create one file combining the 2-3 pages of reviews downloaded using the extension.
- The number of reviews averaged about 100 per competitor, and we utilized the same for further analysis.
- GitHub Access to Scraped Reviews Files

b. We further used Python for topic modeling through pandas and spacy libraries.

Step 3- Topic Modeling

a. A combination of 15 topics was generated using the LDA models in Python.

   - GitHub Access to Files

b. Later, we used GenAI tools such as ChatGPT to interpret the topics.

   - GitHub Access to Files

c. The topics were manually bucketed into different categories based on our understanding of the content and context of the customer pain points.

   - Topic Modeling Results (Excel)

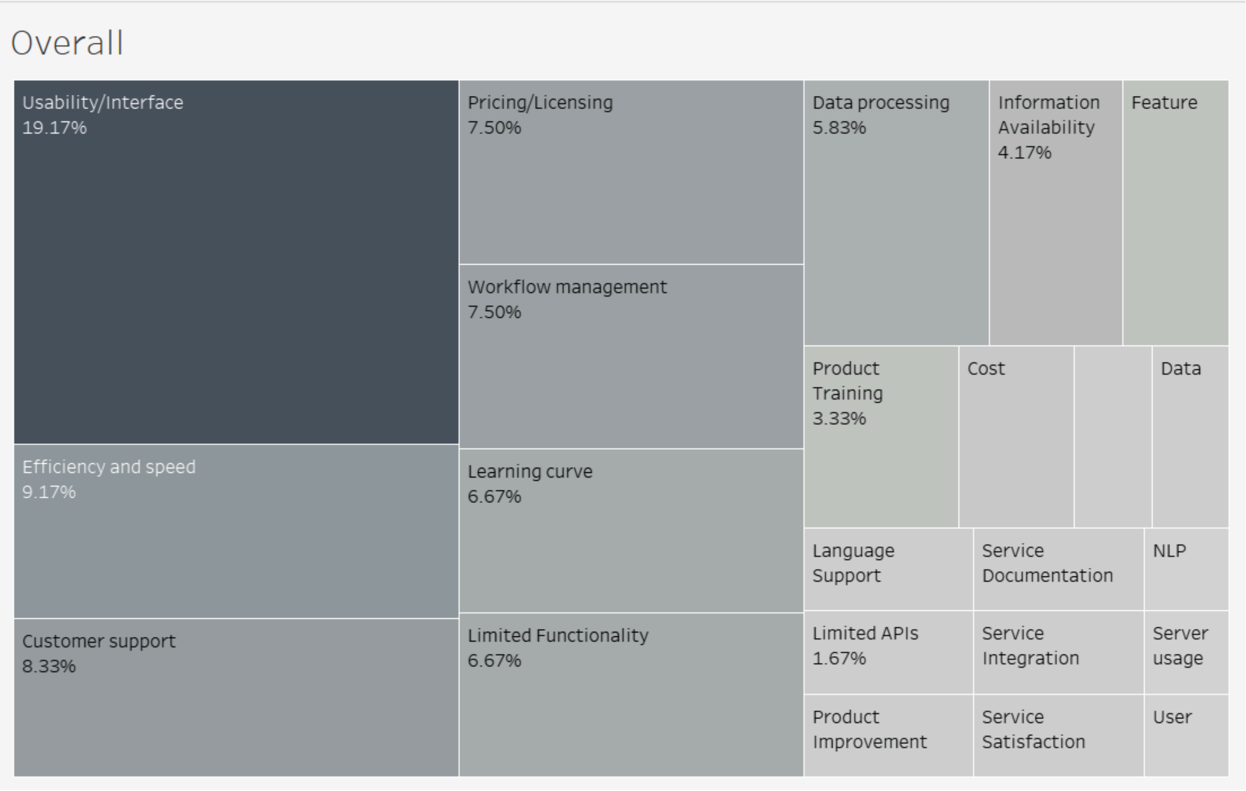
Step 4- Visualization

a. **Customer Pain Points Analysis on Excel:** The identified and interpreted 15 topics were divided into separate categories in Excel, and redundant or repetitive categories were removed.

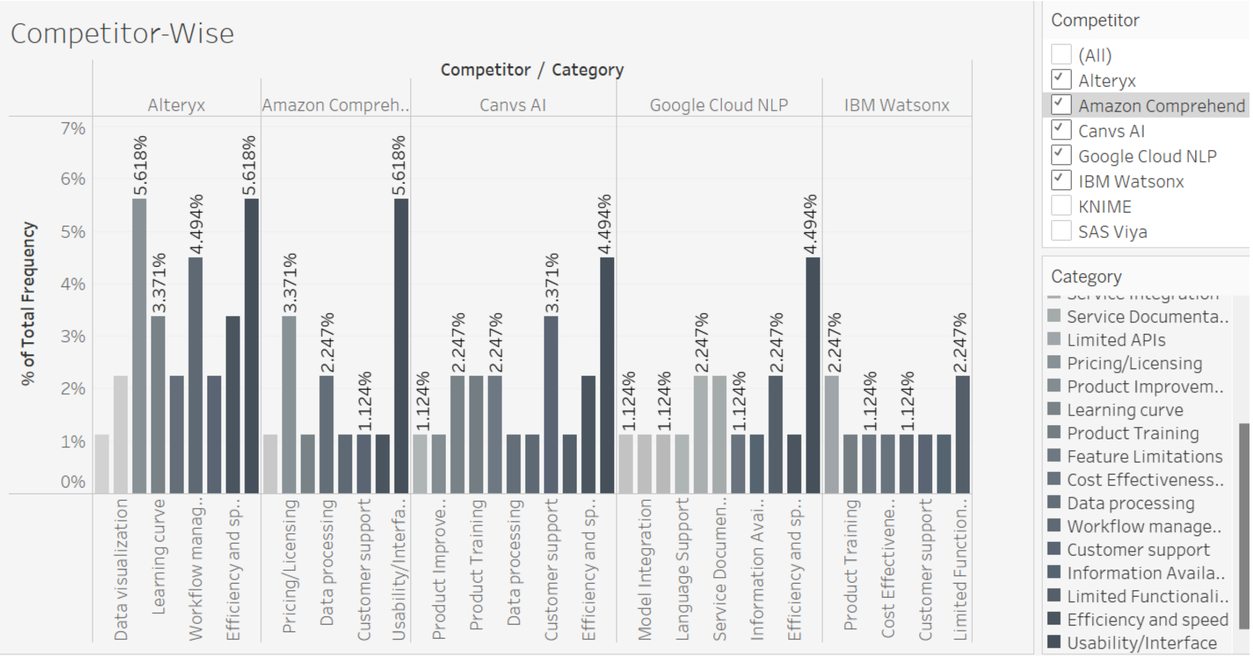
b. **Bar Graphs/Flow Charts from Tableau/Excel:** We later input the topics extracted in multiple categories through Excel (pivot tables) and visualize the results of competitors as well as overall categories across competitors to give our recommendations.

Tableau Data Visualizations

Consumer Pain Points: Topic Frequencies: [Link to Dashboard]



Consumer Pain Points: Competitor-wise - Topic Frequencies - [Link to Dashboard]



Competitor

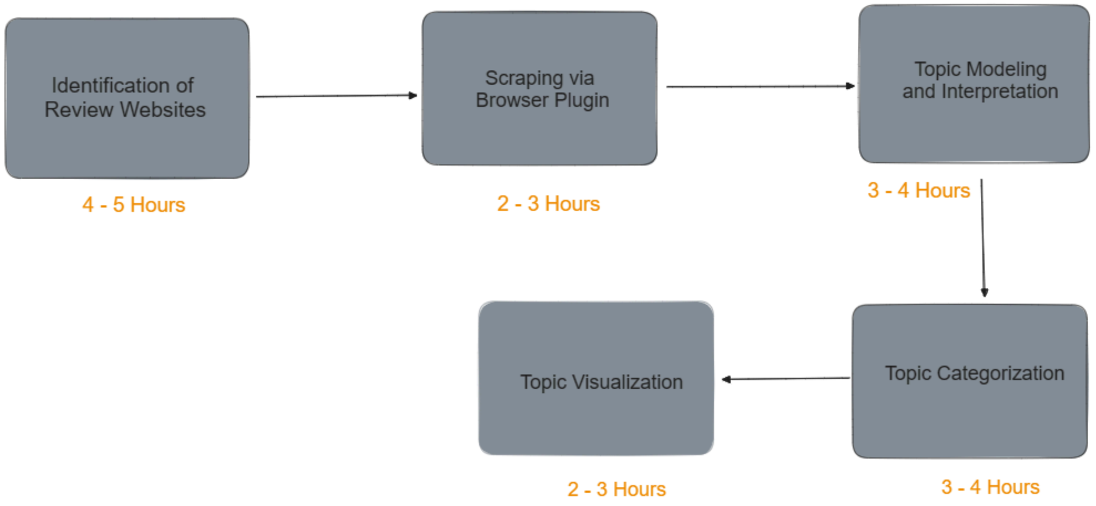
- ☐ (All)
- ☑ Alteryx
- ☑ Amazon Comprehend
- ☑ Canvs AI
- ☑ Google Cloud NLP
- ☑ IBM Watsonx
- ☐ KNIME
- ☐ SAS Viya

Challenges in Deriving Meaningful Insights

- Social Listening: Reddit Scraping: Obtaining relevant reviews from Reddit posed a challenge due to the platform's cluttered nature and the need for approval for the paid API, which would take 8-12 weeks, exceeding our project timeline.
- Twitter Scraping: While the free developer plan for Twitter allows scraping up to 1500 posts per month, initial difficulties in obtaining and using the API key resulted in authorization issues, preventing successful data extraction.
- YouTube Videos: Most videos found on YouTube were tutorials demonstrating specific use cases rather than providing insights into user pain points. Except for Alteryx and IBM WatsonX, which each had one review video, YouTube did not offer a reliable source for user pain points.
- Third-Party Analysts: Efforts to scrape reviews from G2 and Gartner using third-party API providers like Scrapefly and DataShake were unsuccessful, due to paid services or firewall restrictions. This led us to resort to using only browser plugins for scraping.
- Developer Communities: Communities owned by companies typically contained FAQs or solutions to user questions, which did not provide meaningful insights into user pain points.

Timeline Demonstration

Final Timeline: Competitor Pain-Points Identification



Identification of Review Websites

4 - 5 Hours

Scraping via Browser Plugin

2 - 3 Hours

Topic Modeling and Interpretation

3 - 4 Hours

Topic Categorization

3 - 4 Hours

Topic Visualization

2 - 3 Hours

Total Time: 14 - 19 Hours

Recommendations

Objective 1 - Blog Strategy

Current Topics Strengths of Megaputer:

- Specialized Expertise: Your topics emphasize Megaputer's expertise in data analysis, language processing, and legal insights, showcasing your company's specialized knowledge.
- Tool-Centric Approach: Topics like "Text Analysis and PolyAnalyst" and "Data Analysis and Language Processing Tools" highlight specific tools offered by Megaputer, helping potential customers understand your product offerings.

Proposing a comprehensive blog content strategy after a competitor analysis of 6 different companies that have the same capabilities as Ployanalyst.

Part 1: Targeting Users of PolyAnalyst

Audience: Analysts, developers, data scientists, researchers, and other users of PolyAnalyst.

Objectives:

- Product education and knowledge dissemination
- Customer support and engagement
- Community building and user collaboration

Content Focus:

- Product-focused articles highlighting PolyAnalyst and its unique features
- How-to guides, tutorials, and best practices for effective use of PolyAnalyst

Part 2: Targeting Decision Makers and Executives

Audience: decision-makers, VPs, and executives (both current and potential customers).

Objectives:

- Lead generation and customer acquisition
- New business creation and expansion
- Upselling and cross-selling opportunities
- Increasing market visibility and brand awareness

Content Focus:

- Product advocacy emphasizing the business value of PolyAnalyst
- Thought leadership articles showcasing Megaputer's industry insights and expertise

- Reviews and testimonials highlighting customer success stories
- PR announcements and updates about Megaputer's achievements and milestones
- Event summaries and coverage, including webinars, conferences, and workshops

Common Relevant Themes Across Competitors:

1. AI modeling and machine learning
2. Data analytics
3. Data management
4. Data visualization
5. Product specific

Common topics across competitors:

- Cloud-based AI Solutions: Exploring the advancements and benefits of AI-powered solutions offered by cloud providers like Google Cloud, AWS, and Azure.
- Data Analytics and Insights: Understanding the role of data analytics in extracting valuable insights, improving decision-making, and driving business growth.
- Machine Learning Models: Delving into the development, deployment, and optimization of machine learning models using platforms like SageMaker and Knime.
- AI in Security and Fraud Detection: Exploring the application of AI in enhancing security measures, detecting fraudulent activities, and ensuring data protection.
- Generative Models and AI Accelerators: Understanding the capabilities and applications of generative models, AI accelerators, and AI-driven solutions for various industries.
- Data Management and Storage Solutions: Discussing the importance of efficient data management, storage solutions, and data lakes in handling large volumes of data.

- AI-driven Customer Engagement: Exploring how AI technologies are revolutionizing customer engagement strategies, personalizing user experiences, and improving customer satisfaction.
- Edge Analytics and Intelligent Devices: Investigating the role of edge analytics and intelligent devices in processing data closer to its source, enabling real-time insights and actions.
- Data Security and Compliance in the Cloud: Addressing the Challenges and Best Practices for Ensuring Data Security, Compliance, and Privacy in Cloud-Based Environments.
- Future Trends in AI and Data Science: Analyzing the latest trends, innovations, and advancements in AI, machine learning, and data science that are shaping the future of industries.

Content Frequency:

In line with our competitive analysis, we recommend that Megaputer publish 3-4 blogs per week. This frequency will help maintain a consistent presence and engagement with our audience, ensuring the timely delivery of valuable insights and updates.

Final recommendations

Industry-Specific Applications: While your current topics focus on tools and techniques, integrating industry-specific applications can make the content more relatable and valuable for potential customers in various sectors.

Recommended Change: Integrate topics like "AI in Healthcare: Medical Text Analysis Solutions" or "AI in Finance: Fraud Detection and Data Analysis" to showcase how Megaputer's solutions cater to specific industries.

Trends and Innovations: Including topics on emerging trends and innovations in AI, analysis, and data can position Megaputer as a thought leader in the field.
Recommended Change: Add topics such as "Future Trends in AI and Data Science" or "Advancements in Natural Language Processing" to highlight Megaputer's forward-thinking approach.

Customer Success Stories and Case Studies: Sharing success stories and case studies can build credibility and demonstrate the real-world impact of Megaputer's solutions.
Recommended Change: Incorporate topics like "Customer Success Stories: How Megaputer Transformed Data Analysis for Businesses" or "Case Study: Enhancing Legal Compliance with Megaputer's Language Processing Tools."

Ethical and Legal Considerations: Given the increasing importance of data ethics and compliance, addressing these aspects can help build trust with potential customers.
Recommended Change: Introduce topics like "Ethical Considerations in AI and Data Analysis" or "Navigating Data Privacy Laws with Megaputer's Solutions" to address these concerns proactively.

Educational Content: Providing educational content can help potential customers understand the value of AI and data analysis better.

Recommended Change: Include topics such as "Beginner's Guide to Data Analysis and Language Processing" or "Understanding AI: Basics and Applications" to educate your audience.

Additional Recommendations

Content Generation and Distribution Plan:

To gain a competitive edge, Megaputer will initiate a focused content generation strategy aimed at creating high-quality blogs, whitepapers, and customer case studies over the next six months.

Distribution Channels:

Once the content is created, we will strategically distribute it across various channels, including social media platforms and email newsletters. This multi-channel approach aims to maximize visibility, enhance SEO rankings, and drive lead generation by attracting a wider audience.

Objectives:

- Increase Cross-Channel Traffic: By leveraging multiple distribution channels, we aim to increase traffic across various platforms, enhancing brand visibility and engagement.
- Boost SEO Performance: Regular and relevant content publication will improve SEO rankings, making Megaputer more discoverable to potential customers searching for related topics.
- Lead Generation: Through informative and engaging content, we aim to attract and convert potential leads into loyal customers, driving business growth and expansion.

Consumer Pain Points

Based on the analysis of competitors' negative customer reviews, it's evident that there are several key pain points in the text analysis software market that Megaputer Intelligence can address to enhance its competitiveness.

Apart from the blog scraping through NLP, we also went through the dislikes mentioned in all of the scraped reviews (excels consolidated here) manually and we see below the biggest end-user pain points:

- High Licensing Costs: The cost of licensing is a prevalent concern among users, especially for platforms like Alteryx, SAS Viya, and potentially Google Cloud Natural Language API for high-volume usage. Users often find the licensing expensive, particularly for smaller companies or individual developers, which may hinder wider adoption or require a significant investment to achieve ROI.
- Steep Learning Curve: Across various platforms like Alteryx, Google Cloud Natural Language API, KNIME, and SAS Viya, users consistently mention the steep learning curve as a significant challenge. Transitioning from familiar tools like Excel or adapting to new analytics platforms can be difficult due to complexity, lack of training, and limited documentation.
- Performance Issues: Performance-related issues, such as slow processing speed, interface lags, and memory consumption, are frequently mentioned across platforms like Alteryx, KNIME, and SAS Viya. These issues impact user experience, particularly when dealing with large datasets or complex workflows, and can lead to frustration and inefficiency.
- Documentation and Learning Resources: Users consistently express concerns about the lack of comprehensive documentation and learning resources across various platforms, including Alteryx, Google Cloud Natural Language API, KNIME, and SAS Viya. Limited

resources make it challenging for users to troubleshoot issues, learn new features, or optimize their workflows effectively.

- Integration Challenges: Integration challenges are noted across platforms like Alteryx, KNIME, and SAS Viya, where users face difficulties integrating these tools into existing analytics ecosystems or workflows. This challenge adds complexity to adoption and may require additional time and effort to ensure smooth integration with other tools or systems.
- Limited Features or Functionality: Some users mention limited features or functionality compared to other alternatives in the market, particularly in terms of visualization tools, customization options, or support for specific use cases. This limitation may impact the suitability of these platforms for certain tasks or workflows and influence users' adoption decisions

Based on our topic modeling results as well as our manual analysis, below are a few recommendations for where Megaputer can differentiate from its competitors and create a marketing strategy that addresses these issues and attracts more trials from end users of these tools:

Final Recommendations

Primary Focus (Part 1: Usability/Interface & Pricing/Licensing)

<u>Usability and Interface:</u>

This topic emerged as the biggest pain point through both topic modeling, and manual sweeping and should be the first focus for the remarketing strategy:

- Recommendation:

    - Allocate resources to overhaul the user interface (UI) and user experience (UX) designs.
    - Conduct usability testing to identify pain points and streamline navigation.
    - Implement intuitive features and workflows to reduce the learning curve for new users.
    - Enhance visualizations and feedback mechanisms to improve user interaction.
    - Make blog content focusing on the new features/ usability enhancements on the website accessible to end consumers.

<u>Pricing or Licensing:</u>

Although pricing fared lower on the topic modeling results, we saw many users expressing their pain points as pricing models of the multiple competitors, which is why we recommend putting it in the primary focus (please note that we do not have pricing information available for all competitors and Polyanalyst, so our recommendations are not detailed)

- Recommendation:
    - Review pricing and licensing models to ensure competitiveness and transparency.
    - Introduce flexible pricing options (e.g., subscription-based, pay-as-you-go) to accommodate varying customer needs.
    - Provide clear communication regarding pricing structures and value propositions.

Secondary Focuses (Part 2: Efficiency and Speed, Customer Support)

<u>Customer Support:</u>

We saw that many customers talked about inefficient or non-systemized customer support systems, where on-one support was not available, the process of filing for support requests was difficult, or simply, there was a language barrier.

- Recommendation:

    - Introduce a Polyanalyst users community channel on the website itself, where multiple users of Polyanalyst can upload their roadblocks, and other members can help them navigate the roadblocks or help them elevate the problems by liking/reposting the complaints on the channel to promote the visibility of the issues directly to the developers of Polyanalyst
    - Strengthen customer support services with additional resources and training for support staff. Implement a multi-channel support system (e.g., live chat, email, phone) for faster issue resolution.
    - A bot could be introduced for the documentation of the different feedback/problems a user is facing, this also helps maintain records for Polyanalyst developers about bugs, which will aid in constant improvements.

Enhance knowledge bases and documentation to empower users to troubleshoot common issues independently.

<u>Efficiency and Speed:</u>

Many customers included comments such as “high data processing time” or “crashes due to the input of heavy datasets," and it could be addressed in the following ways:

- Recommendation:
    - Invest in optimizing algorithms and data processing pipelines to improve speed.
    - Utilize parallel processing techniques and leverage cloud computing resources for scalability.
    - Prioritize performance enhancements in software updates based on user feedback.

Other Considerations (Part 3: Data Processing, Information Availability, Feature Limitations, and Product Training)

Data Processing:

- Recommendation:
    - Continue to invest in optimizing data processing capabilities for efficiency and scalability.
    - Explore opportunities for integrating advanced data processing techniques (e.g., machine learning) into the software.

Information Availability:

- Recommendation:
    - Improve data accessibility and availability within the software to empower users to make informed decisions.
    - Enhance data visualization features to facilitate insight discovery and interpretation.

Feature Limitations:

- Recommendation:
    - Address identified feature limitations through regular software updates and feature enhancements.
    - Prioritize the development of high-demand features based on user feedback and market trends.

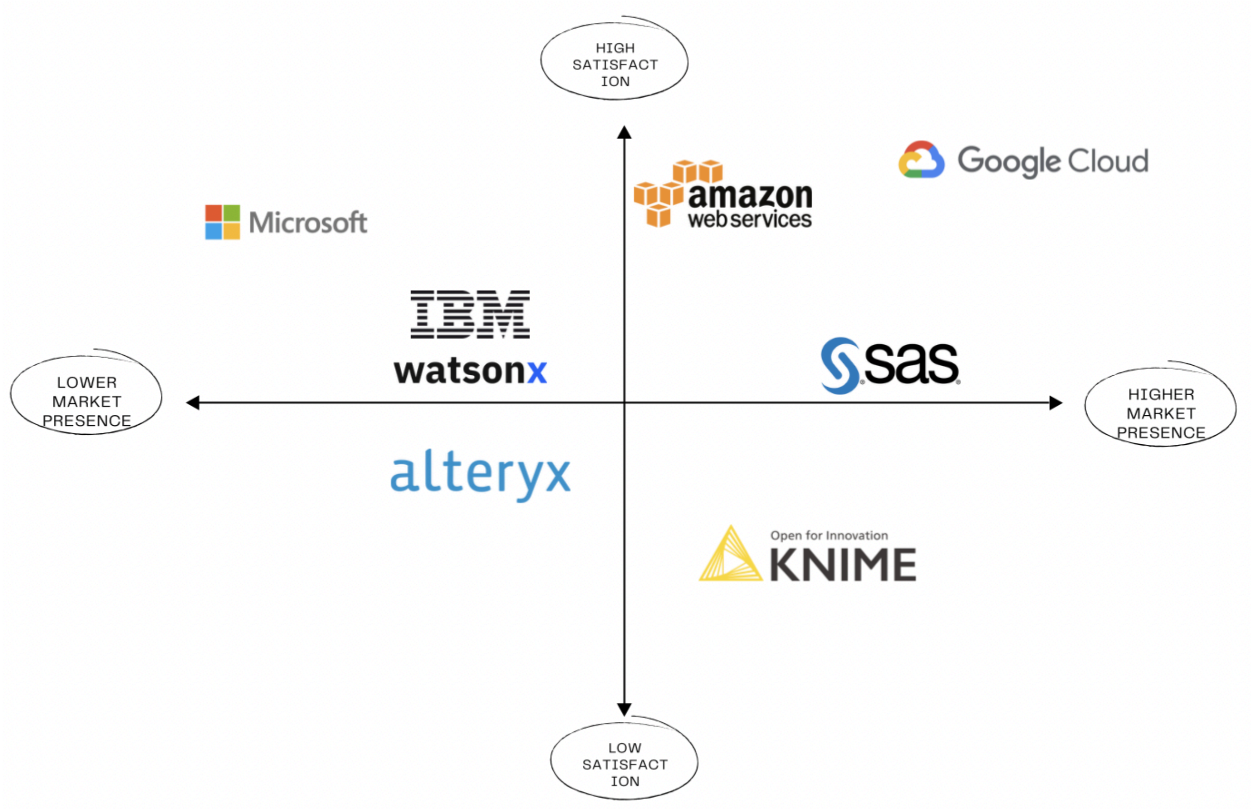
Least Focus (Factors with less than 2% significance)

Exclude from immediate focus: cost-effectiveness and planning, complexity of the platform, language support, limited APIs, product improvement, service documentation, service integration, service satisfaction, NLP capabilities, server usage, and user expectations.

Therefore, focusing on usability improvements through UI/UX overhaul, reevaluating pricing models, and implementing a multi-channel customer support system directly addresses user concerns regarding Polyanalyst's ease of use, efficiency, and value proposition. This comprehensive approach can significantly enhance the user experience and position the poly-analyst-term success.

Appendix

COMPETITOR ANALYSIS



Source: https://www.g2.com/categories/text-analysis

CODES AND OUTPUTS

Inserting the instructions to navigate all the files in GitHub

GitHub Link: https://github.com/Mansiswami/Megaputer/tree/main

Steps for BlogAnalysis

Step 1: Scrape URLs

GitHub Folder: Megaputer Project -> BlogAnalysis -> Step1_ScrapeURLs

JSON Code: Code_URL_(competitor’s name).js

Output File: Output_URL_(competitor’s name).json

Step 2: Scrape Blogs

GitHub Folder: Megaputer Project -> BlogAnalysis -> Step2_ScrapeBlogs

Python Code: Code_BlogScrape_(competitor’s name).py

Output Files: Text files under output folder

Step 3: Word Analysis

GitHub Folder: Megaputer Project -> BlogAnalysis -> Step3_WordAnalysis

Output File: Output_Words_(competitor’s name).xlsx

Step 4: Topic Modeling

GitHub Folder: Megaputer Project -> BlogAnalysis -> Step4_TopicModelling

Python Code: Code_TopicModel_(competitor’s name).py

Final Output: Topic_Interpretation_(competitor’s name).txt

Final results (excel): Final_Output_BlogAnalysis

Tableau Dashboard for Topic Frequencies:

https://public.tableau.com/views/BlogRecommendation_TopicFrequencies/Dashboard2?:language=en-US&:sid=&:display_count=n&:origin=viz_share_link

Tableau Dashboard for Competitor Wise Topic Frequencies:

https://public.tableau.com/views/BlogRecommendation_Competitor-TopicFrequencies/Dashboard1?:language=en-US&:sid=&:display_count=n&:origin=viz_share_link

Steps for Consumer Pain Points

Step 1: Scrape Reviews

GitHub Folder: Megaputer Project -> ConsumerPainPoints -> Step1_ScrapeReviews

Output File: Reviews_(competitor’s name).xlsx

Step 2: Topic Modeling

GitHub Folder: Megaputer Project -> ConsumerPainPoints -> Step2_TopicModelling

Python Code: Code_TopicModel_(competitor’s name).py

Step 3: Output

GitHub Folder: Megaputer Project -> ConsumerPainPoints -> Step3_Output

Final Output: Review_Interpretation_(competitor’s name).txt

Final Results (Excel): Final_Output_CompetitorPainpoints

Tableau Dashboard Competitor Wise:

https://public.tableau.com/app/profile/gautham.shankar.muthukumar/viz/MegaputerCustomerPainPoints/Dashboard1

Tableau Dashboard Overall Heatmap:

https://public.tableau.com/app/profile/gautham.shankar.muthukumar/viz/MegaputerCustomerPainPointsOverall/Overall