Comparative Analysis of Paid and Free Articles on Medium.com

Florian Eder, Moritz Enderle

October 17, 2025

Contents

Abstract

This report presents a comparative analysis of paid and free articles on Medium.com. Using a custom data acquisition pipeline, we collected a dataset of articles and performed statistical analysis to identify key differences in content characteristics, engagement metrics, and author behaviors between paid and free publications.

1 Data Acquisition

There is currently no publicly available dataset of articles on Medium.com we could use for our analysis. The next best option would have been a Medium.com API, but this does not exist. Therefore, we had to build a custom data acquisition pipeline to scrape articles from Medium.com, focusing on both paid and free content.

1.1 Legal Considerations

As this process involves web scraping, which is strictly prohibited by Medium's Terms of Service, we got in touch with Medium's legal department to clarify the situation and obtain permission for our academic project.

Although we received permission to proceed, we ensured our scraping methodology does not overload or negatively impact Medium's services. This includes:

- Respectful crawling delays between requests
- Compliance with robots.txt directives where applicable
- Limiting the total number of scraped articles to a reasonable amount

All collected data is used exclusively for academic research purposes and will not be redistributed or commercialized.

1.2 Page Discovery

Traditional content discovery methods are mostly based around spiders crawling links from one page to another. However, this approach would introduce a significant bias, as most articles on Medium.com are linked only to other articles within the same field (e.g. technology articles link to other technology articles). This would lead to a dataset that is not representative of the overall article distribution on Medium.com.

This led us to discover Medium's sitemaps as a more suitable approach for page discovery. The main sitemap is located at `https://medium.com/sitemap/sitemap.xml` and contains references to 20440 individual sitemaps, each containing up to 7000 URLs. In total, this results in 32 million URLs [as of March 2025]. We completed the whole process over 6 hours with a Gaussian distributed delay with mean 0.05 seconds. From these URLs, we were able to filter out a portion of non-article URLs (e.g. user profiles, tag pages, etc.). This finally resulted in 14702 individual sitemaps and a total of just over 30 Million URLs.

To impose structure on the crawl schedule we adopted the sitemap-provided priority scores as an initial ordering heuristic. From there we applied a "most-seen-first" policy: recommended-article links observed while rendering a page trigger incremental boosts to their target URLs'

crawl priority. This feedback loop yields a balanced sampling across topical clusters while still emphasizing pages with demonstrably high visibility.

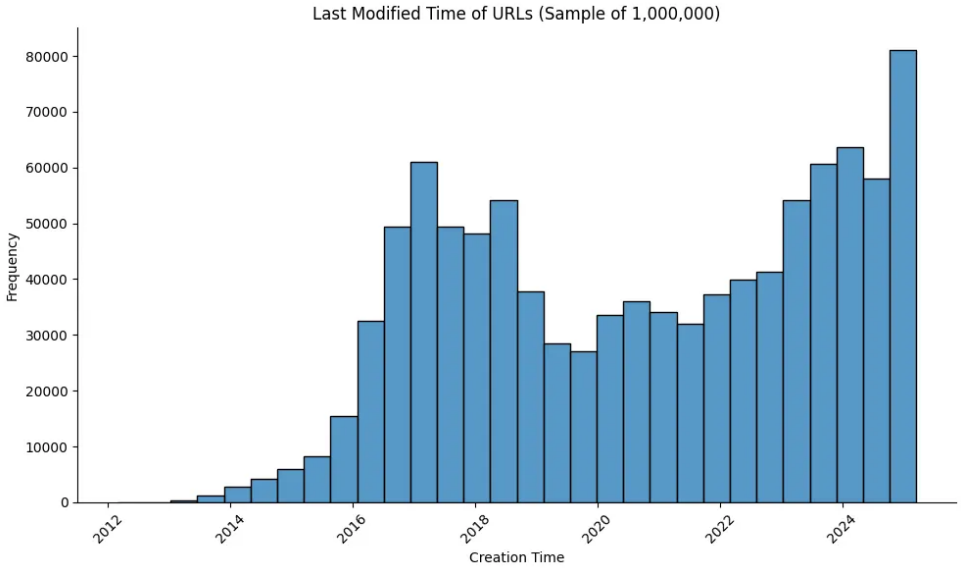

Figure 1: Distribution of Medium article last modification dates from sitemap data (2012-2024). The graph shows a peak in article creation around 2017, followed by a decline during the COVID-19 pandemic, and a notable resurgence since 2022, reflecting Medium's evolving content landscape and user engagement patterns.

1.3 Scraping the Articles

The articles on Medium.com are dynamically loaded using JavaScript, which means a simple HTTP request to fetch the HTML content is not sufficient. This is done mainly to optimize loading times and improve user experience. To overcome this issue, we used a headless browser to fully render the page before extracting the content. We employed Playwright, which provides a high-level API to control headless browsers. We integrated it into Python using the `playwright-python` package.

Due to the high number of articles to be scraped, we parallelized the process using multiple worker processes. Each worker operates independently, fetching URLs from the database, rendering the pages, and extracting the relevant data.

We extracted most data directly from the rendered HTML using CSS selectors. However, some data points are embedded in JSON-LD structured data within the page, which we parsed to extract additional metadata. In order to capture the full comment thread, we programmed the scraper to simulate user interactions such as scrolling and clicking "see all responses" buttons.

TODO: Diagram of the scraping pipeline

Premium Content Access

We fetched premium-only pages by logging in with a stored cookie as a JSON file. This allowed us to access member-only content while not having to deal with username and password authentication. During premium content scraping, we lowered the rates to avoid overloading the servers.

1.4 Database Storage

We stored data in DuckDB, a columnar database optimized for analytical queries. We designed the schema to include tables for sitemaps, URLs, articles, authors, and comments, with relationships maintained through foreign keys. This enables efficient querying for our comparative analysis.

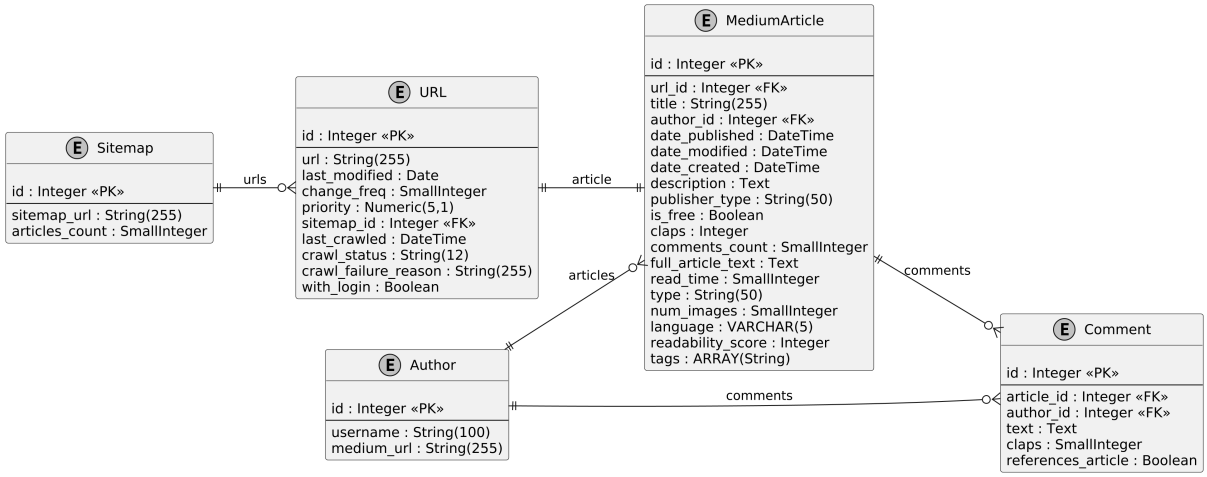

Figure 2: Entity-relationship diagram of the scraping data model.

1.5 Monitoring

We implemented telemetry to track scraping health via structured worker logs aggregated into daily roll-ups for latency, error, and retry trends. We also configured the pipeline to push batch metrics to Weights & Biases such as success ratios, premium hit rates, and fed into dashboards with alert thresholds.

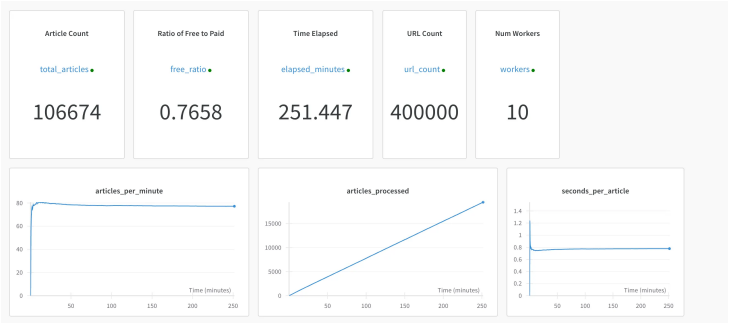

Figure 3: Weights & Biases dashboard summarizing worker throughput, error rates, and premium access metrics collected during scraping.

2 Data Analysis

2.1 Dataset Overview

Our initial scraping process, initiated from the 32 million URLs derived from the sitemaps, culminated in a raw dataset of **65, 248** fully scraped articles.

Ensuring a Fair Comparison

Medium.com introduced the ability for paid memberships in early 2017. As we want to ensure comparability between free and paid articles, taking into account any articles published before this date would introduce a significant bias. Furthermore, to allow for sufficient adoption of the commercial plan by authors and stabilize publication patterns, we only consider articles published after the first of January 2020.

This filtering step resulted in a final analysis dataset of **33, 510** articles published between January 2020 and May 2025, contributed by **24, 639** unique authors, and containing a total of **83, 064** responses.

Of this final, cleaned dataset, **33.6**% (**11, 262** articles) were classified as member-only (paid), and **56.5**% (**22, 248** articles) were free. This composition provides a strong basis for comparative statistical analysis. The primary data points successfully extracted for this analysis include the article's text, estimated reading time, clap count, response count, author follower count, and premium status.

In Figure 4, the adoption and proportion of paid articles within our sample over time is clearly visible. The chart shows the monthly distribution of our scraped articles, where the consistent and increasing proportion of paid articles (relative to free articles) confirms that the post-2020 filter successfully captured the period of active growth in Medium's monetization strategy.

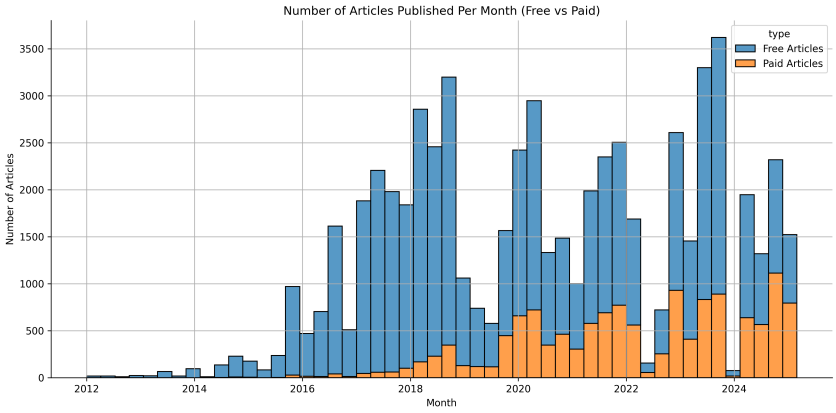

Figure 4: Distribution of scraped articles per month (2020-2025), segmented by premium status. The chart illustrates the growing relative proportion of member-only content in the dataset over time.

2.2 Comparative Statistics

To quantify the differences in engagement and content length between member-only (paid) and free articles, we conducted a comparative statistical analysis. We focused on key metrics such as **TODO:** .

2.2.1 Descriptive Statistics

Table 1 presents the core descriptive statistics for the engagement and length metrics, segmented by the article's premium status. These statistics highlight a clear difference in average engagement, particularly in clap count, where paid articles appear to significantly outperform free articles.

Table 1: Descriptive Statistics for Key Metrics (Post-Jan 2020)

Metric	Status	N	Mean	Median	Std. Dev.
Clap Count	Paid	11,262	407.9	207.0	519.0
	Free	22,248	122.4	40.0	272.1
Response Count	Paid	11,262	4.9	2.0	7.0
	Free	22,248	1.2	0.0	3.1
Reading Time (min)	Paid	11,262	6.0	5.0	2.9
	Free	22,248	5.9	5.0	3.2

2.2.2 Hypothesis Testing (Two-Sample t-tests)

Due to the observed differences in mean engagement metrics and reading time, we performed independent two-sample t-tests to formally assess the statistical significance of these differences. Given the large sample sizes, the Central Limit Theorem allows us to proceed with t-tests despite the non-normality and heteroscedasticity of the underlying populations, focusing on differences in the sample means.

The null hypothesis (H_0) for each test is that there is no difference between the means of paid and free articles for a given metric ($\mu_{paid} = \mu_{free}$).

The results of the t-tests, presented in Table 2, reveal statistically significant differences between paid and free articles for all examined metrics. For clap count and response count, the t-statistics exceed 50, with p-values effectively at zero, indicating extremely strong evidence against the null hypothesis of no difference in means. This suggests that paid articles have substantially higher levels of engagement compared to free articles. Conversely, for reading time, the t-statistic of 2.26 yields a p-value of 0.0235, which, while below the conventional 0.05 threshold, indicates a marginally significant difference, implying that paid articles are, on average, slightly longer in reading time than free ones. These findings underscore the potential impact of premium status on content engagement and length within the Medium.com ecosystem.

Table 2: Two-Sample T-Test Results for Key Metrics

Metric	t-statistic	p-value
Clap Count	54.690392	< 0.001
Response Count	53.863954	< 0.001
Reading Time (min)	2.264893	0.023528

TODO: maybe extend this part?

3 Topic Modeling

3.0.1 Introduction

Topic modeling using embeddings represents a modern approach to uncovering thematic structures in text data, leveraging dense vector representations to capture semantic similarities. Unlike traditional methods like Latent Dirichlet Allocation (LDA), embedding-based techniques utilize pre-trained language models to generate contextual embeddings, allowing for more nuanced topic discovery that accounts for word polysemy and context.

In our analysis of Medium.com articles, we use embedding-based topic modeling to compare thematic distributions between paid and free content. This method enables us to identify clusters of semantically similar articles and assess whether certain topics offer a higher propensity for paid content.

3.0.2 Data Preprocessing & Generation of Embeddings

Preprocessing for embedding-based models is not as intensive as for traditional models, as embeddings inherently capture semantic relationships. However, the scraped content was saved in a markdown format, which included various non-textual elements such as images, code snippets, and formatting syntax. To ensure the quality of the text data used for topic modeling, we implemented a preprocessing pipeline which removed these non-textual elements, retaining only the core textual content of each article. This step was crucial to prevent noise from affecting the embedding generation and subsequent topic modeling.

For generating embeddings, we utilized the `prdev/mini-gte` model developed by QTACK [1] in

3.0.3 Methodology

We utilize BERTopic, a topic modeling framework that combines transformer-based embeddings with clustering algorithms. The process involves:

1. Generating sentence-level or document-level embeddings using a pre-trained model
2. Reducing dimensionality with techniques like UMAP
3. Clustering embeddings using HDBSCAN
4. Extracting representative topics using class-based TF-IDF

TODO: Describe hyperparameters (e.g., embedding model, clustering parameters), training process, and evaluation metrics used for topic quality assessment.

3.0.4 Results

The embedding-based topic modeling using BERTopic identified 26 distinct topics across the dataset. We analyzed the proportion of paid articles within each topic and performed two-sample t-tests to assess if these proportions differ significantly from the overall paid proportion (33.6%).

Table 3 summarizes the mean proportion of paid articles per topic, along with t-statistics and p-values. Significant differences ($p < 0.05$) indicate over- or under-representation of paid content.

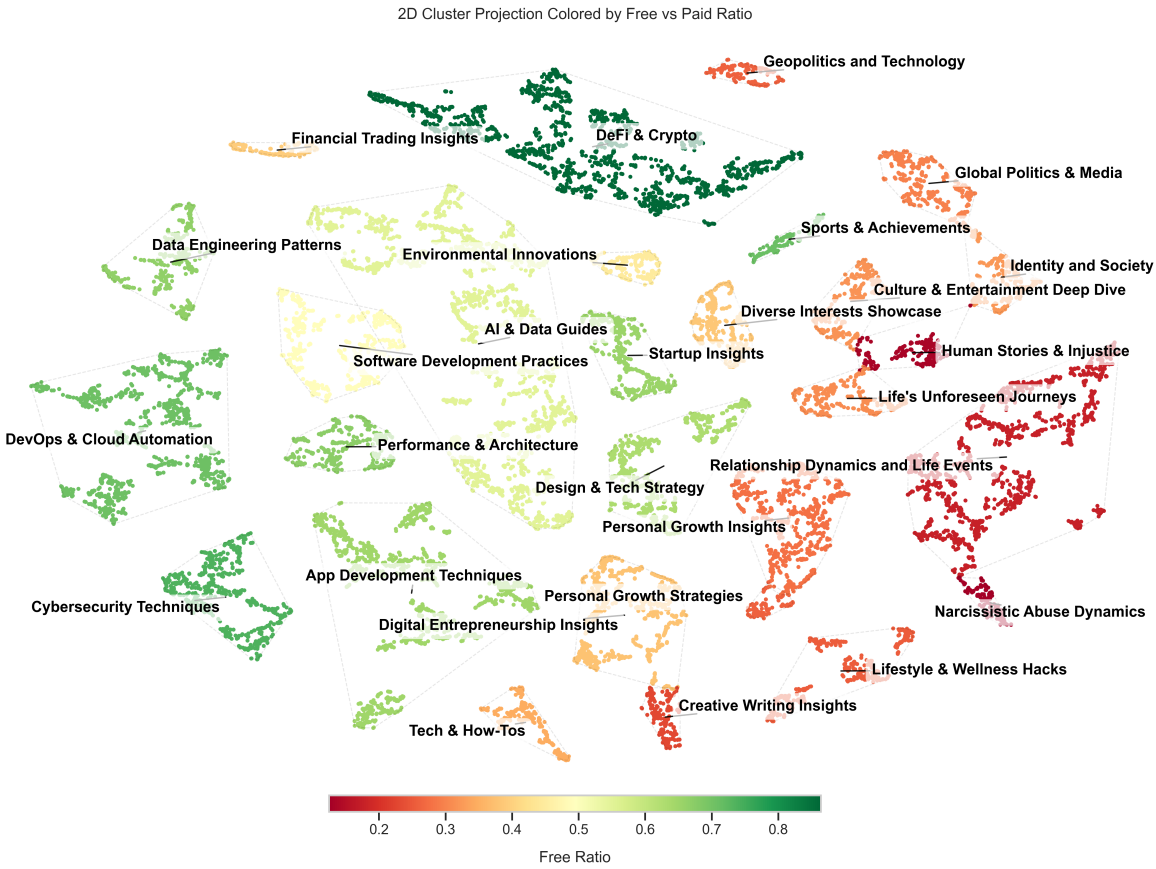

Figure 5: 2D visualization of topic clusters showing free-to-paid article ratios. Color gradient (red to green) indicates free article proportion: red for lower free ratios (more paid), green for higher.

Table 3: Proportion of Paid Articles per Topic with Cluster Size and T-Test Results

Topic	Size	Mean	t-statistic	p-value
DeFi & Crypto	**1523**	**0.864**	**24.065**	**0.000**
Cybersecurity Techniques	**579**	**0.737**	**4.633**	**0.000**
Sports & Achievements	162	0.710	1.599	0.112
DevOps & Cloud Automation	**1014**	**0.708**	**3.878**	**0.000**
Performance & Architecture	379	0.673	0.834	0.405
Data Engineering Patterns	424	0.665	0.540	0.589
Startup Insights	393	0.654	0.052	0.958
App Development Techniques	1087	0.649	-0.221	0.825
Tech Strategy	603	0.632	-1.061	0.289
AI & Data Guides	**1971**	**0.551**	**-9.075**	**0.000**
Software Development Practices	**611**	**0.489**	**-8.070**	**0.000**
Environmental Innovations	**204**	**0.446**	**-5.922**	**0.000**
Financial Trading Insights	**153**	**0.392**	**-6.579**	**0.000**
Diverse Interests Showcase	**337**	**0.383**	**-10.178**	**0.000**
Digital Entrepreneurship Insights	**631**	**0.379**	**-14.174**	**0.000**
Tech & How-Tos	**257**	**0.346**	**-10.303**	**0.000**
Identity and Society	**284**	**0.324**	**-11.818**	**0.000**
Culture & Entertainment Deep Dive	**311**	**0.318**	**-12.638**	**0.000**
Life's Unforeseen Journeys	**304**	**0.309**	**-12.937**	**0.000**
Global Politics & Media	**348**	**0.296**	**-14.557**	**0.000**
Personal Growth Insights	**691**	**0.276**	**-22.101**	**0.000**
Personal Growth Strategies	**100**	**0.260**	**-8.908**	**0.000**
Geopolitics and Technology	**170**	**0.259**	**-11.690**	**0.000**
Lifestyle & Wellness Hacks	**384**	**0.253**	**-18.020**	**0.000**
Creative Writing Insights	**201**	**0.229**	**-14.268**	**0.000**
Relationship Dynamics and Life Events	**1266**	**0.177**	**-44.341**	**0.000**
Human Stories & Injustice	**195**	**0.133**	**-21.280**	**0.000**
Narcissistic Abuse Dynamics	**152**	**0.125**	**-19.607**	**0.000**

TODO: Include visualizations like topic embeddings in 2D space, topic-word clouds, and comparative topic prevalence charts between paid and free content.

The embedding-based model identified **TODO:** insert number topics across the dataset. Topic distributions revealed **TODO:** key findings, such as prevalent topics in paid vs. free articles and semantic similarities.

TODO: Include visualizations like topic embeddings in 2D space, topic-word clouds, and comparative topic prevalence charts between paid and free content.

3.0.5 Discussion

The embedding-based topic modeling highlights **TODO:** insights into thematic differences between paid and free articles. For example, paid articles may cluster around **TODO:** specific semantic themes, while free content shows diversity in **TODO:** other areas.

TODO: Discuss the advantages of embedding-based approaches over traditional methods, implications for content monetization, and limitations such as computational requirements or interpretability challenges.

4 AI Generated Content

5 Conclusion

TODO:

References

[1] QTACK. mini-gte: A compact text embedding model. Hugging Face, 2024. Accessed: 2025-10-17.