Comparative Analysis of Paid and Free Articles on Medium.com

Florian Eder, Moritz Enderle

November 10, 2025

Contents

Abstract

This report analyzes differences between paid (member-only) and free articles on Medium.com. To enable this comparison, we developed a custom data acquisition pipeline to scrape and structure a novel dataset of 65,248 articles. After filtering to ensure content comparability, our final analysis dataset comprised 33,510 articles. We employed statistical analysis (two-sample t-tests), embedding-based topic modeling (UMAP, DBSCAN), and NLP-based text analysis to compare engagement, thematic focus, and writing quality (grammatical correctness and AI-generated content) between paid and free content.

Our results demonstrate significant disparities. Paid articles receive substantially higher engagement in terms of "claps" (mean: 407.9 vs. 122.4) and responses (mean: 4.9 vs. 1.2), with both differences being highly statistically significant ($p < 0.001$). Topic modeling revealed a strong thematic divide: technical topics such as "Crypto & Web3" (86.2% free) and "Emerging Tech" (63.5% free) are predominantly free, while personal and lifestyle themes like "Health & Well-being" are largely monetized (16.7% free). Text analysis showed that while paid articles have a statistically significant lower number of grammar errors ($p < 0.001$), the practical effect size is small ($d = 0.1167$). We found no statistically significant difference in the prevalence of AI-generated content ($p = 0.065$).

We conclude that an article's topic is a primary differentiator between paid and free content on Medium, more so than grammatical quality or the use of AI tools. The platform's ecosystem appears structured to favor monetization for personal and lifestyle content, while technical topics are more frequently used to build a free audience.

1 Data Acquisition

There is currently no publicly available dataset of articles on Medium.com we could use for our analysis. The platform also does not provide an official API for data access. 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.

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.

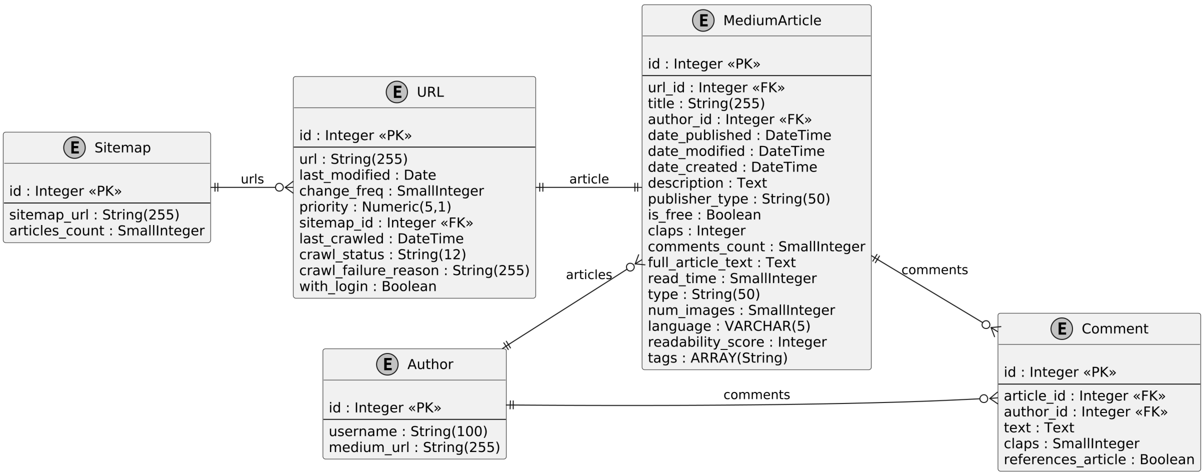


Figure 1: Entity-relationship diagram of the scraping data model.

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.

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. To overcome this issue, we employed playwright to fully render the pages before extracting the relevant data. 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.

For member-only (paid) articles, we implemented a method to access the full content by purchasing the Medium membership and using the associated cookies during scraping.

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. The database schema is shown in Figure 1.

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 $\mathbf{65,248}$ 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 is illustrated in Figure 2, which shows the distribution of articles per month from 2020 to 2025, highlighting the increasing proportion of paid content over time.

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 **66.4**% (**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.

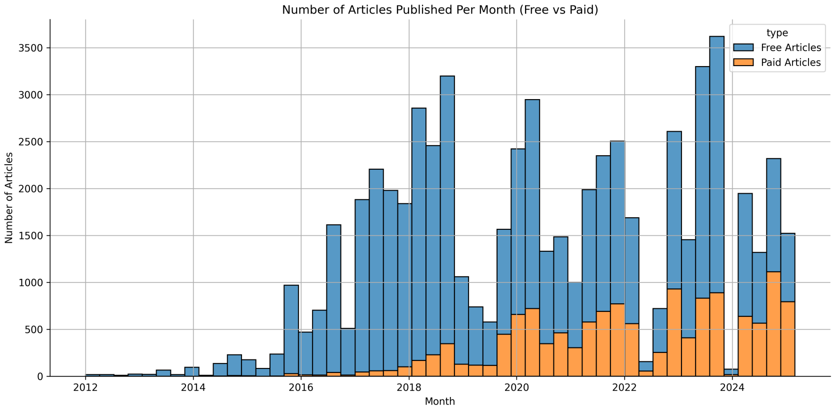


Figure 2: Distribution of scraped articles per month (2020-2025), segmented by premium status. The chart illustrates the growing relative proportion of member-only content in the dataset over time.

2.2 Descriptive Statistics

Table 2.2 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.

Metric	**Label**	**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

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

2.2.1 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 2.2.1, 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. Similarly, for the 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.

Metric	**t-statistic**	**p-value**
Clap Count	54.690392	< 0.0001
Response Count	53.863954	< 0.0001
Reading Time (min)	2.264893	0.023528

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

2.3 Discussion

Paid articles on Medium.com generate far greater reader engagement than free ones, reflecting stronger audience investment in premium content. This gap suggests that the paywall not only filters for higher-quality or more committed writing but also supports deeper interaction between authors and readers. Overall, the premium model appears to drive both greater visibility and meaningful participation within the platforms ecosystem.

3 Topic Modeling

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) [4], embedding-based techniques utilize pre-trained language models to generate contextual embeddings, allowing for context based similaritiy search rather than purely relying on independant words [1].

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

3.1 Data Preprocessing & Generation of Embeddings

To ensure the quality of the text data used for topic modeling, we implemented a preprocessing pipeline which removed 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 [9] in early 2025. This distilled version of the original General Text Embeddings (GTE) model [5] achieved remarkable performance in the MTEB v2 English Benchmark [7], outperforming larger models while ensuring efficient use of memory and computational resources. This was especially suitable for our large dataset of Medium articles, allowing us to generate high-quality embeddings without prohibitive resource consumption.

3.2 Methodology

We conduct topic modeling through density-based clustering on a low-dimensional manifold learned from sentence embeddings. The overall workflow consists of the following steps:

- **Inputs and balanced sampling**: Two embedding matrices are loaded—one for free articles and one for member-only articles—along with a metadata file containing per-article labels and titles. To address class imbalance, stratified sampling is applied, selecting `N_SAMPLES_PER_CLASS = 11000` articles from each class.
- **Manifold learning**: The sampled embeddings are reduced to three dimensions using UMAP [6] with parameters `n_neighbors = 20`, `min_dist = 0`, and `n_components = 3`. The low `min_dist` setting promotes the formation of tight, well-separated clusters to support subsequent density-based clustering, while 3D preserves sufficient structure for analysis and visualization.
- **Density-based clustering**: DBSCAN [3] is applied to the 3D UMAP coordinates with `eps = 0.3` and `min_samples = 150`. This non-parametric method automatically determines the number of clusters and identifies noise (label `-1`), making it well-suited for heterogeneous web text where not all articles align with dominant themes.
- **Topic labeling from representative titles**: For each non-noise cluster, 50 representative article titles are uniformly sampled across the cluster's UMAP extent. These titles are used to manually assign human-readable topic names that capture the cluster's thematic core.

- **2D visualization**: Non-noise 3D UMAP coordinates are further projected into two dimensions using t-SNE (`perplexity = 30`, `learning_rate = 200`, `random_state = 42`). The resulting 2D scatter plot displays clusters with convex hulls, color-coded by the within-cluster free-to-paid article ratio.

- **Hypothesis tests within clusters**: Let $Y_i \in \{0, 1\}$ indicate whether article i is free (1) or paid (0), with $p_0 = 0.5$ representing the overall free proportion in the full embedding population. For each non-noise cluster c, a one-sample t-test evaluates $H_0 : \mathbb{E}[Y \mid c] = p_0$ against the two-sided alternative. The complete results can be viewed in Table A.

3.3 Results

Figure 3 presents the 2D t-SNE visualization of the clustered articles, with color coding to indicate the proportion of free articles within each cluster. A clear gradient emerges from left to right, with clusters on the left predominantly free (green) and those on the right predominantly paid (red). The thematic labels assigned to each cluster reveal, that this gradient corresponds to a shift from tech-based topics (e.g., software engineering, cloud computing, AI) to more personal and lifestyle-oriented themes (e.g., relationships, personal growth, health).

We can further group the identified clusters into higher-level themes based on their labels. Table 3.3 summarizes these themes, along with their sizes, average share of free articles, and variances. In this table, the previously observed gradient is reinforced: technical themes such as "Emerging Tech & Engineering" exhibit higher average ratio (0.635), while personal themes like "Health & Well-being" show a lower ratio (0.167).

A special mention goes to the "Crypto & Web3" cluster, which stands out with the highest mean free proportion of 0.862. This suggests that articles in this domain are mostly free.

Statistical significance was assessed for each cluster using one-sample t-tests comparing the proportion of free articles within the cluster to the overall population proportion of 0.5. Close to all clusters exhibited highly significant deviations ($p < 0.05$), indicating that the distribution of free versus paid articles is not uniform across topics. This further supports the notion that the topic of an article is strongly associated with its premium status on Medium.com.

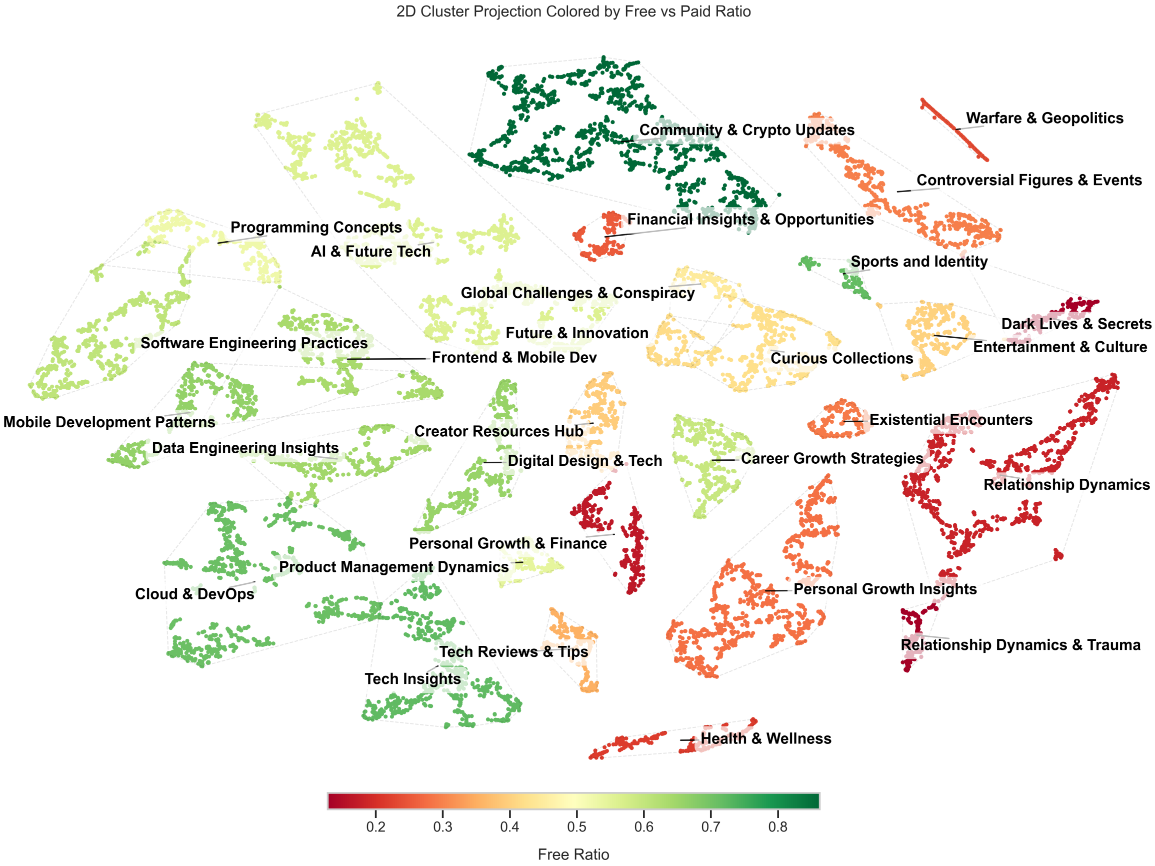


Figure 3: 2D Visualization of Topic Clusters illustrating the Ratio of Free to Paid Articles. The Color Gradient from Green to Red Indicates Increasing Proportion of Paid Articles within Each Cluster.

Theme	Size	Mean Free	Variance
Crypto & Web3	1,489	0.862	0.119
Emerging Tech & Engineering	5,746	0.635	0.225
Product, Business & Growth	1,973	0.521	0.239
Society, Culture & Global	1,487	0.331	0.217
Personal & Emotional Life	2,515	0.240	0.183
Health & Well-being	492	0.167	0.144
Creativity & Curiosity	1,358	0.520	0.216

Table 3: Summary of Higher-Level Topic Clusters with Mean Free Proportions

3.4 Discussion

Our embedding-based topic modeling reveals a pronounced thematic divide on Medium.com, with technical domains like "Crypto & Web3" and "Emerging Tech & Engineering" predominantly free (e.g., 86.2% free in crypto), potentially to foster community engagement and attract a broad readership. Conversely, personal and lifestyle themes such as "Health & Well-being" and "Relationship Dynamics" lean heavily toward paid content (e.g., 16.7% free in health), suggesting authors monetize intimate or specialized narratives. The t-SNE visualization's left-to-right gradient underscores this shift from collaborative tech discourse to introspective storytelling, supported by highly significant t-test results ($p < 0.001$ for most clusters), indicating topic choice strongly predicts premium status.

Beyond engagement disparities, this pattern may reflect Medium's ecosystem dynamics: free tech articles could serve as lead magnets for building author followings, while paid personal content caters to readers seeking depth or exclusivity. However, limitations include potential biases in embedding models toward mainstream topics and the subjective nature of cluster labeling.

4 Text analysis

4.1 AI Generated Content

Since the public accessability of Large Language Models, the question arises, whether blog posts are human-written or at least partially AI generated. Detecting AI-content is not an easy task, since it's relying on AI as well. We've tested several ai-detection models from the RAID benchmark [2] on a custom text, which we additionally rewrote using popular Large Language Models. While most models on the benchmark exhibited high false positives rates, SuperAnnotates "ai-detector-low-fpr" showed comparatively improved performance. It's modelcard can be found on Huggingface [10].

4.1.1 Methology

Given that most of the articles in our dataset exceed the maximum input length of AI detection models, we adopted a three-stage approach. In the first stage we preprocessed the texts to remove markdown elements.

In the second stage, we split each article into sentences that were individually analyzed for AI-generated text:

$$S_{\text{sentence}}(w_k) = \frac{S_{c(w_k)}}{|K|} \tag{1}$$

- $S_{\text{sentence}}(w_k)$: Final score for the k-th word.
- $c(w_k)$: The single chunk (sentence) that contains word w_k.
- $S_{c(w_k)}$: The raw model score for that specific chunk.
- $|K|$: Number of words in the sentence K

In the last stage, we applied a sliding-window approach. A chunk of words holds approximately 7 words:

$$S_{window}(w_k) = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{|c_i|-1} S_i \cdot \delta(i+j,k)}{\sum_{i=0}^{m-1} \sum_{j=0}^{|c_i|-1} \delta(i+j,k)} \tag{2}$$

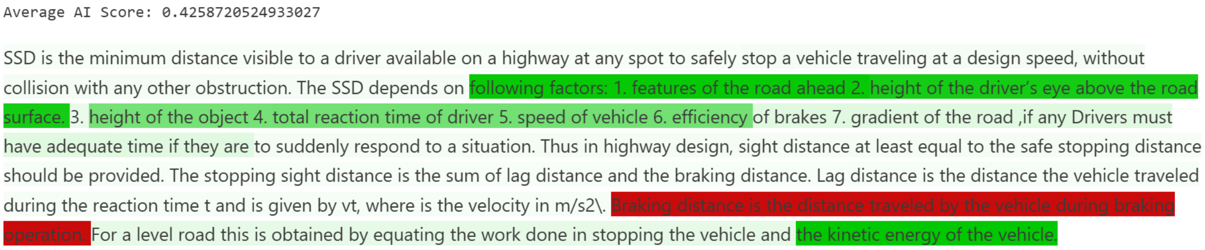
- $S_{\text{window}}(w_k)$: Final score for the k-th word in the text.
- k: Index of the target word in the full text.
- i: Index of chunk.

- j: Index of a word within chunk c_i.
- m: Total number of chunks.
- $|c_i|$: Number of words in chunk c_i.
- S_i: Model score for the entire chunk c_i.
- $\delta(i+j,k)$: Kronecker delta (1 if $i+j=k$, else 0).

The resulting scores from both approaches then are aggregated and normalized:

$$S_{final} = \frac{S_{avg-window} + S_{avg-sentence}}{2} \tag{3}$$

We visualize on a per-word basis in Figure 4. For article comparision, we computed the mean of the per-word scores, denoted as *average AI score*.



```
Average AI Score: 0.4258720524933027
```

SSD is the minimum distance visible to a driver available on a highway at any spot to safely stop a vehicle traveling at a design speed, without collision with any other obstruction. The SSD depends on following factors: 1. features of the road ahead 2. height of the driver's eye above the road surface. 3. height of the object 4. total reaction time of driver 5. speed of vehicle 6. efficiency of brakes 7. gradient of the road ,if any Drivers must have adequate time if they are to suddenly respond to a situation. Thus in highway design, sight distance at least equal to the safe stopping distance should be provided. The stopping sight distance is the sum of lag distance and the braking distance. Lag distance is the distance the vehicle traveled during the reaction time t and is given by vt, where is the velocity in m/s2\. Braking distance is the distance traveled by the vehicle during braking operation. For a level road this is obtained by equating the work done in stopping the vehicle and the kinetic energy of the vehicle.

Figure 4: Visualization of AI scores on a per-word basis. Green indicates low AI likelihood (low score), black represents uncertainty (score ≈ 0.5), and red denotes high AI likelihood (high score).

4.1.2 Descriptive Statistics and Hypothesis Testing

For a fair comparison between free and paid articles, we balanced the dataset by sampling from the free articles to match the number of paid ones.

The descriptive statistics for the AI-generated content are summarized in Table 4:

Article Type	**Mean**	**Median**	**Standard Deviation**
Paid	0.392	0.383	0.119
Free	0.389	0.381	0.130

Table 4: Summary of AI-generated content scores by article type.

Additionally, a two-sample t-test was conducted to assess whether the observed differences in average AI scores between free and paid articles were statistically significant. The t-test results indicate a t-statistic of **-1.844** and a p-value of **0.065**. While the mean AI-generated score for paid articles is slightly higher than for free articles, this difference is not statistically significant at the 95% level.

4.1.3 Distribution Over Time

To explore temporal trends in AI-generated content, articles were grouped into half-year periods from 2017 onwards. The analysis reveals a slight upwards trend from 2017 to 2025. Between 2017 and 2022, average AI scores remained stable around 0.33-0.38, suggesting human-written texts and a low, text-inherent false positive rate. The outliers may be explained by the closed beta access to ChatGPT and other language models. A marked rise appears after late 2022, when ChatGPT became publicly available, and further after March 2023, when the ChatGPT API (gpt-3.5-turbo) was released to the wider public. Figure 5 presents the distribution of AI scores for both article types across half-year intervals.

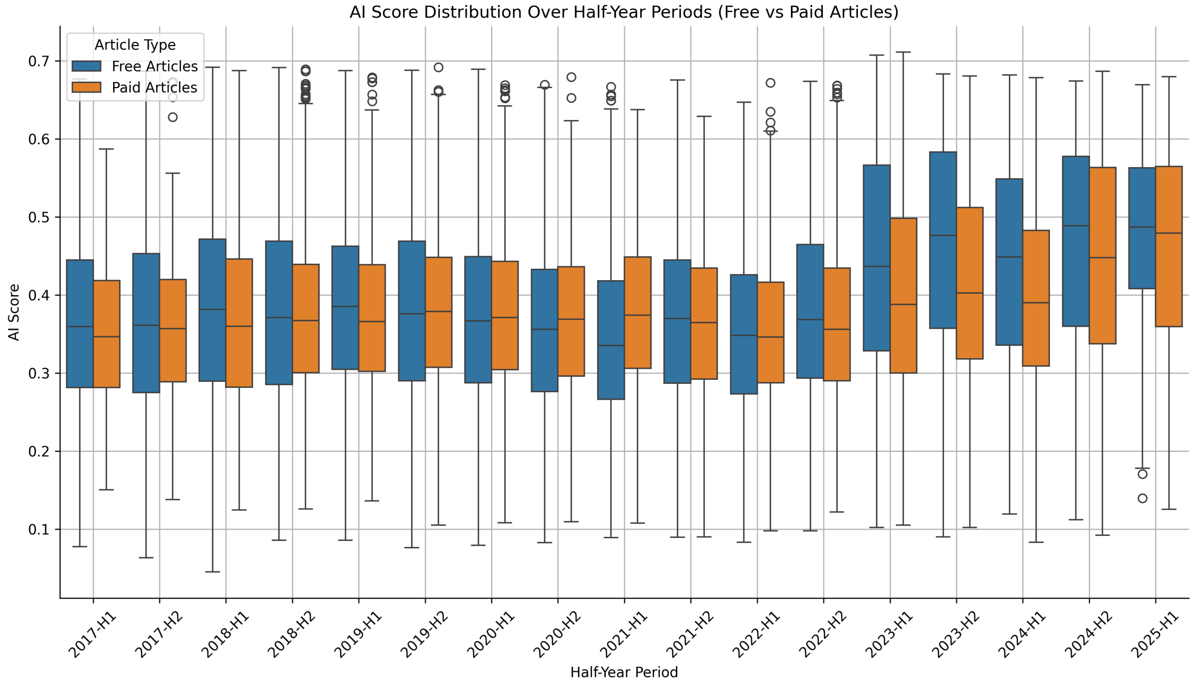


Figure 5: Distribution of AI-generated content scores across half-year periods for free and paid articles.

4.1.4 Discussion

Our analysis indicates that AI-generated content is present in both free and paid articles, but the difference in prevalence between the two types is small and not statistically significant. The stable trend over time suggests that, despite the growing availability of AI text-generation tools, there has been a small but not dramatic increase in AI-generated content within the dataset period.

The methodology of combining sentence-level detection with a sliding-window approach ensures that even long articles exceeding model input limits can be analyzed effectively. However, the inherent limitations of current AI detectors - including sensitivity to adversarial patterns and sampling strategies - mean that AI-score estimates should be interpreted with caution.

4.2 Grammar Analysis

While content quality can be influenced by topic, author expertise, and stylistic choices, grammatical correctness remains a central indicator of writing quality and editorial oversight. Especially in the case of paid articles, we expect them to be subject to more rigorous editorial and review processes.

4.2.1 Methodology

To analyze grammar errors, we use `LanguageTool` for python [8], an open-source grammar and style checker capable of identifying a wide range of linguistic issues. The articles were preprocessed to remove markdown elements, code snippets, and other non-linguistic components, before being processed in full. We exclude certain linguistic rules from the detector, as they frequently produce false positives. Additionally, we omit spelling errors, since domain-specific terms (e.g., `fastapi` in the programming domain) are often incorrectly flagged as misspellings. We call the total number of detected issues per article *grammar_error_count*

4.2.2 Descriptive Statistics and Hypothesis Testing

The descriptive analysis indicates a higher mean grammar error count in free articles ($\mu_{\text{free}} = 16.85$) compared to paid articles ($\mu_{\text{paid}} = 12.84$), as well as a higher standard deviation among free articles.

To formally assess this difference, we conducted a two-sample Welch's t-test, using a sample size equal to that of the smaller group (paid articles) to ensure a fair comparison. Our null hypothesis (H_0) posits no difference in mean grammar counts between free and paid articles ($\mu_{\text{free}} = \mu_{\text{paid}}$). The test yielded a large absolute t-value (t = 9.9961) and a very small p-value ($p = 1.15 \times 10^{-23}$), leading us to reject H_0 and conclude that the difference is statistically significant.

We further evaluated the effect size using Cohen's d and Hedges' g, which indicated a small difference of **0.1167** standard deviations, suggesting that the practical magnitude of this difference is limited.

4.2.3 Distribution of Grammar Errors

To further understand the nature of the differences identified in the t-test, we analyzed the distribution of errors across specific grammatical categories. The accompanying bar chart (Figure 6) illustrates the average number of errors per word in an article for both free (`is_free = True`) and paid (`is_free = False`) content, with the y-axis on a logarithmic scale to account for wide variations in error frequency.

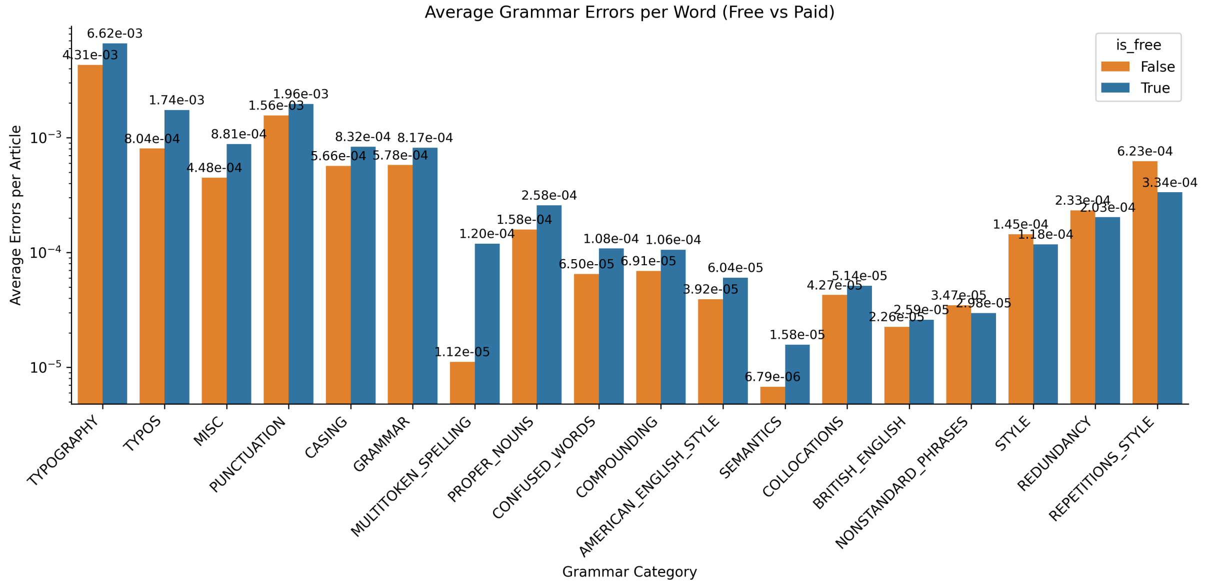


Figure 6: Average Grammar Errors per Article (Free vs Paid)

The visualization reinforces the overall findings: free articles exhibit a higher mean error count in the vast majority of categories. This gap is particularly evident in high-frequency error types such as typography errors (`TYPOGRAPHY`: 7.08 vs. 5.45) and american/british english errors (`TYPOS`: 2.33 vs. 1.12).

A notable exception to this pattern emerges in style-related categories. Paid articles show a higher average error count for repetitions in the text (`REPETITIONS_STYLE`: 0.88 vs. 0.48). A similar, though less pronounced, trend is observed for redundancy (`REDUNDANCY`: 0.34 vs. 0.29) and general style errors (`STYLE`: 0.21 vs. 0.16). The complete results can be viewed in Appendix B

4.2.4 Discussion

The results of the Welch's t-test confirm our initial hypothesis that paid articles do contain a statistically significant lower number of mean grammar errors compared to free articles, which may be explained by a more effective editorial or review process. However, the effect size is small, which suggests that while the diffenence is real and consistent across the sample, the practical distinction in quality, as percieved by the human reader, may be minimal. Paid status is a significant predictor of lower error counts, but its not a strong one.

The categorical error distribution gives a better understanding and indicates that the nature of errors differs between the two content types. Free articles show higher error rates in foundational grammar, whereas paid articles often use repetitions and redundancy.

In summary, while our analysis confirms that paid articles exhibit fewer mechanical grammar errors, the effect size and the differences in style-based errors suggest a more complex relationship. The editorial process for paid articles may be primarily focused on correcting foundational grammatical errors, while being more prone to stylistic choices that automated tools flag as problematic.

5 Conclusion

A Hypothesis tests within clusters

Topic	Size	Mean Ratio	t-statistic	p-value
Community & Crypto Updates	**1489**	**0.138**	**40.563**	**0.000**
Tech Insights	**765**	**0.278**	**13.663**	**0.000**
Sports and Identity	**155**	**0.284**	**5.949**	**0.000**
Cloud & DevOps	**1004**	**0.291**	**14.586**	**0.000**
Mobile Development Patterns	**477**	**0.338**	**7.496**	**0.000**
Digital Design & Tech	**528**	**0.343**	**7.603**	**0.000**
Data Engineering Insights	**489**	**0.354**	**6.755**	**0.000**
Frontend & Mobile Dev	**564**	**0.363**	**6.735**	**0.000**
Software Engineering Practices	**694**	**0.393**	**5.746**	**0.000**
Career Growth Strategies	**457**	**0.407**	**4.042**	**0.000**
AI & Future Tech	**1720**	**0.441**	**4.952**	**0.000**
Product Management Dynamics	216	0.458	1.226	0.221
Programming Concepts	354	0.475	0.957	0.339
Global Challenges & Conspiracy	165	0.552	-1.326	0.187
Curious Collections	**539**	**0.579**	**-3.704**	**0.000**
Future & Innovation	**180**	**0.583**	**-2.261**	**0.025**
Entertainment & Culture	**344**	**0.599**	**-3.735**	**0.000**
Creator Resources Hub	**383**	**0.606**	**-4.229**	**0.000**
Tech Reviews & Tips	**264**	**0.648**	**-5.015**	**0.000**
Controversial Figures & Events	**603**	**0.703**	**-10.910**	**0.000**
Existential Encounters	**210**	**0.710**	**-6.672**	**0.000**
Personal Growth Insights	**1021**	**0.721**	**-15.725**	**0.000**
Financial Insights & Opportunities	**197**	**0.746**	**-7.920**	**0.000**
Warfare & Geopolitics	**153**	**0.765**	**-7.694**	**0.000**
Health & Wellness	**292**	**0.784**	**-11.788**	**0.000**
Relationship Dynamics	**1097**	**0.820**	**-27.503**	**0.000**
Personal Growth & Finance	**370**	**0.832**	**-17.098**	**0.000**
Dark Lives & Secrets	**200**	**0.870**	**-15.520**	**0.000**
Relationship Dynamics & Trauma	**187**	**0.872**	**-15.155**	**0.000**

Table 5: Proportion of Free Articles per Topic with Cluster Size and T-Test Results

B Language Tools Error IDs

A list with all the errors language_tool detects can be found on our github page: https://github.com/M-Enderle/Medium-Mining/blob/main/report/language_tool_rule_ids_en.txt

References

[1] Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453, 07 2020.

[2] Liam Dugan, Alyssa Hwang, Filip Trhlik, Josh Magnus Ludan, Andrew Zhu, Hainiu Xu, Daphne Ippolito, and Chris Callison-Burch. Raid: A shared benchmark for robust evaluation of machine-generated text detectors, 2024.

[3] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press, 1996.

[4] Hamed Jelodar, Yongli Wang, Chi Yuan, Xia Feng, Xiahui Jiang, Yanchao Li, and Liang Zhao. Latent dirichlet allocation (lda) and topic modeling: models, applications, a survey, 2018.

[5] Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning, 2023.

[6] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020.

[7] Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*, 2022.

[8] Daniel Naber. *A Rule-Based Style and Grammar Checker*. PhD thesis, Technische Fakultät, Universität Bielefeld, 08 2003.

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

[10] SuperAnnotate. SuperAnnotate/ai-detector-low-fpr. Hugging Face, 2024. [Accessed: 2025-10-24].