Main Content Extraction from Business Web Pages: An Empirical Evaluation of Heuristic and ML-Based Methods

Mayank Yadav
Data and Knowledge Engineering
Otto von Guericke University
Magdeburg, Germany
Email: mayank.yadav@st.ovgu.de

***Abstract*—Extracting meaningful textual content from business websites poses significant challenges due to complex layouts, dynamic elements, and pervasive boilerplate that often obscure core information. This study systematically evaluates five content extraction techniques—jusText, Trafilatura, Ollama, Readability, and a simple Baseline approach—on multilingual business websites to address this gap.**

Motivated by the need for high-quality data to support downstream tasks like clustering and classification, we designed a robust experimental framework using manually curated ground-truth references. Each method was assessed across three critical dimensions: semantic relevance, noise removal, and completeness relative to page intent.

Results reveal clear trade-offs: Trafilatura excels in noise reduction, Baseline achieves the highest semantic and coverage scores, jusText offers balanced but variable performance, while Readability underperforms and Ollama favors precision at the cost of content coverage. No single method is universally optimal; selection depends on specific application needs and data quality priorities.

This work provides actionable insights and practical guidance for researchers and practitioners engaged in content extraction from diverse business websites.

Keywords— content extraction, web mining, business websites, text processing, multilingual data.

1. Introduction

Extracting the main textual content from web pages is a foundational task in web mining, information retrieval, and natural language processing. While significant progress has been made for structured sources like news websites or blogs, business websites remain particularly challenging due to their diverse layouts, embedded legal content, inconsistent HTML structures, and multilingual elements. This complexity is especially pronounced in German business websites, where pages often include regulatory sections (e.g., Impressum, Datenschutz) and non-standard design patterns.

Effective content extraction from such sites is critical for enabling downstream applications like semantic clustering, company profiling, or market trend analysis. Traditional rule-based tools, though lightweight and interpretable, often fail to generalize across highly varied layouts. Meanwhile, machine learning approaches offer flexibility but require labeled data and significant computational resources.

We systematically evaluate four established content extraction tools—jusText, Trafilatura, Ollama, and Readability—alongside a simple Baseline method for comparison on a large corpus of real-world German business websites. By combining heuristic and structural extraction paradigms, we aim to understand how each method performs in noisy, heterogeneous business contexts. Our goal is to inform researchers and practitioners about the trade-offs involved in web content preprocessing and to establish a benchmark for future extraction techniques targeting business-oriented domains.

2. Related Work

The task of extracting meaningful textual content from web pages has a long history, particularly in fields such as information retrieval, web mining, and natural language processing. Over the years, researchers have proposed a wide range of approaches, from simple heuristics to sophisticated machine learning models.

2.1. Heuristic-Based Approaches

Early methods focused on handcrafted rules that exploit structural patterns in HTML documents. One notable example is Boilerpipe[5], which analyzes shallow text features—such as the number of HTML tags, link density, and average word length—to distinguish main content from boilerplate. It works particularly well on news sites and blog posts where content is typically well-separated from sidebars and headers. However, its reliance on regular HTML structures makes it less effective on websites with dynamic or irregular layouts, such as those found in the business domain.

Another tool, jusText, builds upon this heuristic approach by incorporating language-specific elements. It segments web pages into paragraphs and evaluates each one based on features like stopword density, link ratio, and paragraph length. Its multilingual capabilities and rule-based logic make it well-suited for non-English content, including German. jusText is particularly effective at removing common noise such as navigation links and disclaimers, although it may occasionally discard short but meaningful content blocks.

Trafilatura [1] represents a more advanced evolution of the heuristic family. It combines multiple strategies, including content-to-tag ratio, heading structures, and coherence scoring, to identify core content across different page formats. Trafilatura also includes optional fallbacks, such as regular expressions and simple language models, making it robust against malformed HTML or distributed content. Its flexibility and ease of integration have made it a popular choice in academic and large-scale scraping projects.

2.2. DOM-Based and Structural Methods

In contrast to block-based heuristics, another line of research emphasizes Document Object Model (DOM) structure analysis. Tools like Readability.js aim to identify the central content block by evaluating structural cues in the DOM tree—such as tag type, nesting depth, and sibling relationships. This method assumes a dominant linear content flow, making it ideal for narrative pages like news articles or blogs. However, it often struggles with websites that have multiple focal points, such as product listings or service menus typical of business websites.

2.3. Machine Learning and Deep Learning Approaches

Recent advances in content extraction leverage supervised learning to improve generalizability across diverse layouts. Web2Text[13] formulates content extraction as a sequence labeling task, using Conditional Random Fields (CRFs) trained on manually annotated HTML blocks. This method incorporates both textual and structural features and has shown strong performance on benchmark datasets such as CleanEval.

Further improvements have come from the use of deep learning. For instance, Morbieu et al. [9] proposed a BERT-based classifier that operates at the segment level to distinguish between main content and boilerplate. These models benefit from large-scale pretrained language models and can adapt to a variety of content types, but they require annotated training data and substantial computational resources. Moreover, their performance on non-news or domain-specific websites—such as those in German business contexts—remains underexplored.

2.4. Gap in Literature

Despite these advances, most studies have focused on English-language datasets or well-structured domains like news and blogs. There is a noticeable gap in understanding how these tools perform on multilingual, domain-specific websites, particularly in the business sector where page layouts are highly heterogeneous and often include embedded regulatory content.

This study seeks to address this gap by providing a comprehensive evaluation of five established tools across a real-world corpus of German business websites. Our aim is not only to assess performance but also to offer actionable guidance for practitioners working with complex, multilingual web data.

3. Methods for Content Extraction

In this study, we evaluate five widely used open-source tools for web content extraction. Each represents a different methodological approach ranging from simple rule-based filters to DOM-tree analysis. Below, we summarize how each tool operates and the rationale behind its selection.

3.1. jusText

jusText improves on traditional heuristics by applying paragraph-level analysis and incorporating language-aware filtering. For each paragraph, it evaluates the stopword ratio as an indicator of natural-language text, the link density to suppress navigational blocks, and the paragraph length to eliminate overly short or trivial content.

jusText supports over 30 languages, including German, and makes use of curated stopword lists to tune its filtering logic. It performs well at removing repetitive and non-informative sections such as menus, footers, and disclaimers. However, its strict filtering rules can occasionally remove sparse but meaningful text, especially in minimalistic website designs.

3.2. Trafilatura

Trafilatura is a more flexible and extensible tool that combines multiple heuristic strategies. It analyzes a page using content-to-tag ratios to filter out tag-heavy blocks, heading structure to identify document flow, and paragraph coherence to assess topical consistency.

In addition to these core strategies, Trafilatura includes fallback mechanisms such as regular expressions for known boilerplate patterns and pretrained language models for content classification. It is particularly useful when the relevant content is distributed across multiple blocks or when the HTML is malformed. Trafilatura's robustness and configurability make it a popular choice for large-scale academic and multilingual scraping projects.

3.3. Readability

Readability (popularized through the Readability.js library) takes a DOM-centric approach. It attempts to find the central content block by traversing the HTML tree and scoring each node based on tag type importance (for example, article, p, and div tags), word count, and position within the document structure.

Nodes with the highest content score are retained, and others are discarded. This approach works well when the page follows a narrative layout, such as blog posts or editorial articles. However, business websites often contain multiple sections without a clear dominant block, leading Readability to either extract too much irrelevant content or miss important segments entirely.

3.4. Ollama (LLM-Based Extraction)

Ollama represents a modern approach to content extraction using large language models (LLMs). This method leverages the Llama 3.2 model via local Ollama server to perform intelligent content extraction through natural language understanding rather than rule-based heuristics or DOM analysis. The approach works by providing the LLM with a carefully crafted prompt that instructs it to extract main article or business description text while explicitly ignoring boilerplate content such as navigation menus, footers, advertisements, links, and legal disclaimers.

Unlike traditional methods that rely on structural patterns or statistical features, the LLM-based approach can understand semantic context and make content relevance decisions similar to human judgment. This method processes HTML directly without requiring preprocessing or feature engineering, making it potentially more robust to diverse page layouts and structures. However, it requires access to a local Ollama server and involves higher computational overhead compared to traditional extraction methods.

3.5. Baseline Method

To provide a reference point for comparison, we implemented a simple baseline extraction method using minimal heuristics. This approach serves as a control to evaluate whether sophisticated extraction tools provide meaningful improvements over basic HTML parsing.

The Baseline method operates by using tag-based selection to extract text content from semantic HTML tags most likely to contain main content, specifically paragraph and title tags. It employs minimal preprocessing by removing only script and style tags to eliminate JavaScript and CSS code, followed by text aggregation that concatenates all text from selected tags with whitespace normalization.

This approach makes several simple assumptions. First, paragraph tags typically contain the primary textual content. Second, page titles often provide important contextual information. Third, no complex filtering or scoring is needed for basic content extraction.

The Baseline method was implemented using BeautifulSoup's text extraction capabilities and represents the simplest viable approach to content extraction. While it lacks sophistication in noise removal or structural analysis, it provides a valuable benchmark to assess whether more complex tools justify their computational overhead.

By including this baseline, we can determine whether the advanced heuristics and algorithms employed by the other five methods provide substantial improvements over naive HTML parsing, particularly in the context of German business websites where content structure may vary significantly.

4. Dataset and Experimental Setup

To systematically assess the performance of the five content extraction methods, we designed a large-scale benchmark experiment using real-world business websites. This section describes the dataset construction, extraction pipeline, and evaluation criteria in detail.

4.1. Dataset Composition

Our evaluation dataset comprises exactly 719 manually curated HTML-text pairs, distributed across 200 folders representing unique German company websites, with an average of 3.6 web pages per company. These folders typically contain raw HTML files capturing snapshots of various subpages such as home, services, contact, legal notices, and product details, alongside paired text reference files that were carefully constructed by annotators using a custom labeling script to isolate the main meaningful content. These gold-standard references systematically exclude boilerplate, disclaimers, and navigational text.

This structure ensures a strict one-to-one mapping, with each HTML file having a corresponding reference text file, supporting reliable direct comparisons. Given the diversity of industries spanning technology consultancies, hospitality providers, insurance firms, and retail businesses, this corpus offers a uniquely heterogeneous benchmark. It stands out as one of the most extensive annotated resources for evaluating content extraction on non-news, multilingual business websites.

Moreover, this dataset enabled rigorous quantitative evaluation using metrics such as ROUGE, Precision/Recall/F1, Content-Noise Ratio, and Semantic Coherence, across extraction methods including jusText, Trafilatura, Readability, Baseline, and Ollama.

4.2. Extraction Pipeline

We implemented a modular evaluation pipeline in Python to apply each extraction method to the same HTML files and compare the output against the ground-truth references. The jusText method was configured with the German stopword list provided by the library, using a maximum link density threshold of 0.2, where blocks with greater than 20% anchor text were discarded, a minimum word count of 5, where paragraphs shorter than 5 words were ignored, and a stopword ratio threshold of 0.3 used to detect natural-language content.

Trafilatura was employed with comments and tables excluded through the `include_comments=False` and `include_tables=False` parameters, while `favor_precision=True` was set for better quality control. Custom model-based filters were disabled to ensure a fair heuristic comparison, and the fallback extractor was enabled through `fallback=True` in case primary methods failed.

The Readability implementation utilized `readability-lxml`, following the DOM-based approach established by [14], mirroring the logic of

Table 1: Extraction method hyperparameters and configuration settings

Method	**Key Parameters and Settings**
jusText	German stopword list; max link density = 0.2 (blocks with >20% links discarded); min word count = 5; stopword ratio threshold = 0.3
Trafilatura	include_comments=False; include_tables=False; favor_precision=True; fallback extractor enabled; custom model-based filters disabled
Readability	readability-lxml; min text length = 25 chars; tag scoring prioritizes div, section, article over span, footer (default scoring)
Ollama	Llama 3.2 via local API; HTML input truncated at 5,000 characters; max retries = 10 with 5s delay; 30s timeout for server stability
Baseline	BeautifulSoup extraction on `<p>` and `<title>` tags; script/style tags removed; minimal whitespace normalization

Readability.js. Key parameters included a minimum text length of 25 characters, where only blocks exceeding this threshold were considered, and tags like `div`, `section`, and `article` were scored higher than `span` or `footer` based on default scoring heuristics.

Ollama was deployed using the Llama 3.2 model via a local server API with semantic content extraction through prompt engineering. The input length was limited to 5,000 characters, where HTML was truncated to the first 5,000 characters for processing efficiency; a maximum of 10 retries was implemented with 5-second delays to handle server connectivity issues, and a 30-second timeout was set to prevent hanging operations and ensure pipeline stability.

The Baseline Method employed a simple BeautifulSoup extractor [11] that aggregated text from paragraph and title tags, serving as a minimal preprocessing reference.

File matching and preprocessing ensured that each HTML file was paired with its corresponding text file by matching filenames. HTML files were cleaned by removing script and style tags before extraction, and outputs were normalized through whitespace and line break removal before comparison. This standardization ensured that all methods were tested under consistent conditions, eliminating variability due to preprocessing differences.

4.3. Evaluation Metrics

To comprehensively assess the performance of each extraction method, we employed a multi-dimensional evaluation framework consisting of seven complementary metrics, each scaled from 0 to 5 for intuitive comparison and analysis. The scaling to a common range was essential to enable direct comparison across fundamentally different metrics with varying natural scales and to facilitate aggregate scoring through weighted combinations of individual metrics.

4.3.1. Content Relevance Metrics. ROUGE Scores measure semantic overlap between extracted content and reference text through multiple perspectives. ROUGE-1 evaluates unigram overlap using the formula:

$$\text{ROUGE-1} = \frac{\sum_{i=1}^{n} \text{Count}_{\text{match}}(\text{unigram}_i)}{\sum_{i=1}^{n} \text{Count}(\text{unigram}_i)} \tag{1}$$

where n represents the total number of unigrams in the reference text. ROUGE-2 extends this analysis to bigram overlap, capturing phrase-level semantic similarity through:

$$\text{ROUGE-2} = \frac{\sum_{i=1}^{m} \text{Count}_{\text{match}}(\text{bigram}_i)}{\sum_{i=1}^{m} \text{Count}(\text{bigram}_i)} \tag{2}$$

where m denotes the total number of bigrams in the reference. ROUGE-L assesses the longest common subsequence, accounting for both content and order preservation:

$$\text{ROUGE-L} = \frac{\text{LCS}(X, Y)}{\max(|X|, |Y|)} \tag{3}$$

where $\text{LCS}(X, Y)$ represents the longest common subsequence between reference X and candidate Y, and $|X|$ and $|Y|$ denote their respective lengths. These ROUGE scores quantify semantic fidelity and structural preservation of main content. Raw ROUGE scores ranging from 0.0 to 1.0 were linearly mapped to the 0–5 scale using:

$$\text{ROUGE Score} = 5 \times \text{Raw ROUGE Value} \tag{4}$$

For example, a ROUGE-L value of 0.45 would yield a scaled score of 2.25.

Precision, Recall, and F1 Score provide word-level overlap analysis where Precision measures the proportion of extracted words that appear in reference text:

$$\text{Precision} = \frac{|W_{\text{ref}} \cap W_{\text{cand}}|}{|W_{\text{cand}}|} \tag{5}$$

Recall evaluates the proportion of reference words captured in extracted output:

$$\text{Recall} = \frac{|W_{\text{ref}} \cap W_{\text{cand}}|}{|W_{\text{ref}}|} \tag{6}$$

The F1 Score represents the harmonic mean of precision and recall, balancing content accuracy and coverage:

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{7}$$

where W_{ref} and W_{cand} represent sets of tokenized words from reference and candidate texts respectively. The scaling transformation applied was $\text{F1 Score} = 5 \times \text{Raw F1 Value}$.

4.3.2. Content Quality Metrics. Content-to-Noise Ratio measures boilerplate suppression by identifying and counting predefined noise indicators based on pattern-based analysis [5]. These noise indicators include legal terms (such as "Impressum", "AGB", "Datenschutz", "Privacy", "Cookie"), web artifacts (like "href=", URLs, email addresses, "www."), layout elements ("navigation", "menu", "footer", "header"), commercial content markers ("advertisement", "sponsored", "banner"), social media references ("share", "like", "tweet", "facebook"), and structural patterns such as repeated characters (four or more times consecutively) or excessive punctuation.

The total *Noise Count* is computed as the sum of all detected instances of these indicators within the extracted text. The *Content-to-Noise Ratio* is then scaled between 0 and 5 by subtracting the proportion of noise words relative to the total word count from one and multiplying by five, so that higher values reflect cleaner content with minimal boilerplate.

$$\text{Content-Noise-Ratio} = 5 \times \max\left(0, 1 - \frac{\text{Noise Count}}{\text{Total Words}}\right) \tag{8}$$

Semantic Coherence evaluates text quality through vocabulary diversity and sentence structure analysis [8]. The Type-Token Ratio (TTR) serves as a vocabulary richness indicator, while average sentence length provides a complexity and readability measure. The calculation involves:

$$\text{TTR} = \frac{\text{Unique Words}}{\text{Total Words}} \tag{9}$$

$$\text{Average Sentence Length} = \frac{\text{Total Words}}{\text{Number of Sentences}} \tag{10}$$

Normalization is applied to bound these values appropriately:

$$\text{TTR}_{\text{normalized}} = \min(\text{TTR} \times 2, 1.0) \tag{11}$$

$$\text{Length}_{\text{normalized}} = \min\left(\frac{\text{Average Sentence Length}}{20}, 1.0\right) \tag{12}$$

The final semantic coherence score is computed as:

$$\text{Semantic Coherence} = 5 \times \frac{\text{TTR}_{\text{normalized}} + \text{Length}_{\text{normalized}}}{2} \tag{13}$$

4.3.3. Completeness. Word Overlap Coverage measures the proportion of reference content captured in extracted output [12]. This metric evaluates recall by determining how much of the gold-standard content was retained:

$$\text{Completeness Score} = 5 \times \frac{|W_{\text{ref}} \cap W_{\text{cand}}|}{|W_{\text{ref}}|} \tag{14}$$

where $|W_{\text{ref}} \cap W_{\text{cand}}|$ represents the number of overlapping words between reference and candidate texts.

4.3.4. Scaling and Normalization. All metrics were normalized to a consistent 0–5 scale using linear transformation [4]. This normalization was crucial for several reasons. First, it enables cross-metric comparison by allowing direct comparison of performance across different evaluation dimensions that naturally operate on different scales. Second, it facilitates aggregate scoring through weighted combination of metrics for overall assessment. Third, it provides intuitive interpretation where higher scores consistently indicate better performance across all dimensions.

The scaling addresses the fundamental problem that raw metrics operate on incompatible ranges. For instance, ROUGE scores naturally range from 0 to 1, while word counts can range from zero to thousands, making direct comparison meaningless. Without normalization, metrics with larger natural ranges would dominate composite scores, obscuring the contribution of other important evaluation dimensions.

The general scaling formula applied was:

$$\text{Scaled Score} = 5 \times \frac{\text{Raw Score} - \text{Min Value}}{\text{Max Value} - \text{Min Value}} \tag{15}$$

However, for most metrics in this study, the minimum value was zero, simplifying this to a direct proportional scaling. The choice of a 0-5 scale rather than 0-1 was motivated by improved interpretability, as integer and half-integer values are more intuitive for comparative analysis than decimal fractions.

This comprehensive evaluation framework captures both semantic fidelity and content quality, providing a holistic assessment of extraction performance across diverse business website structures and content types.

4.4. Logging and Output

The evaluation pipeline generated a comprehensive CSV log file containing detailed performance metrics for each extraction method. The `save_enhanced_scores()` function systematically recorded evaluation results with an enhanced schema including Company ID derived from base filename, Extraction Method specification for jusText, Trafilatura, Readability, Baseline, and Ollama, and comprehensive scoring across ROUGE-1, ROUGE-2, ROUGE-L, Precision, Recall, F1-Score, Content-Noise-Ratio, and Semantic-Coherence metrics.

The `run_comprehensive_evaluation()` function orchestrated the entire evaluation workflow by iterating through all company folders, processing HTML-text pairs, and applying all five extraction methods to each document. For each method-document combination, comprehensive metrics were calculated and automatically appended to the CSV log file.

This enhanced tabular format facilitated multi-dimensional performance analysis across ROUGE metrics, precision/recall metrics, and content quality indicators. It enabled method-specific comparisons with standardized scoring across all extraction approaches, company-level aggregation enabling sector-specific and individual performance analysis, and automated data collection eliminating manual transcription errors during large-scale evaluation.

The systematic logging approach ensured complete traceability of evaluation results, with each row representing a single method-document evaluation instance. This structured output enabled subsequent statistical analysis, visualization of score distributions, and identification of optimal extraction methods for different content types and organizational contexts.

5. Results and Analysis

To evaluate the performance of the five extraction methods—Baseline, Ollama, Readability, Trafilatura, and jusText—we analyzed their scores across multiple comprehensive metrics: all ROUGE variants (ROUGE-1, ROUGE-2, ROUGE-L), precision/recall metrics (F1-Score), and content quality indicators (Content-Noise-Ratio, Semantic-Coherence). Visual comparisons are provided to illustrate their performance distributions and trade-offs.

5.1. Comprehensive ROUGE Score Performance

The complete ROUGE analysis reveals distinct performance patterns across all three variants, providing a nuanced view of semantic preservation capabilities.

ROUGE-1 Performance (Unigram Overlap). Baseline achieves the highest ROUGE-1 scores with a mean of 3.04 ± 1.24, demonstrating superior word-level content preservation. Trafilatura follows with competitive performance at 2.64 ± 1.42, showing consistent unigram overlap.

jusText provides moderate performance at 2.42 ± 1.55 with higher variability, while Readability shows limited effectiveness at 1.77 ± 1.48 for business content. Ollama exhibits the lowest ROUGE-1 scores at 0.40 ± 0.61, indicating significant content loss.

ROUGE-2 Performance (Bigram Overlap). Baseline maintains superiority in phrase-level preservation with a mean of 2.81 ± 1.29. Trafilatura demonstrates consistent bigram retention at 2.51 ± 1.48, and jusText shows moderate bigram overlap at 2.28 ± 1.57.

Readability performs poorly on phrase structures with 1.64 ± 1.54, while Ollama shows minimal bigram preservation at 0.22 ± 0.56.

ROUGE-L Performance (Longest Common Subsequence). Baseline excels in structural preservation with 2.92 ± 1.28, maintaining content order and flow. Trafilatura provides good structural coherence at 2.56 ± 1.45, and jusText offers moderate structural preservation at 2.32 ± 1.56.

Readability struggles with content structure at 1.71 ± 1.52, while Ollama shows poor structural alignment at 0.32 ± 0.58.

Key ROUGE Insights. The consistent ranking across all ROUGE variants (Baseline $>$ Trafilatura $>$ jusText $>$ Readability $>$ Ollama) validates the reliability of our findings. Baseline's superior performance across all ROUGE metrics suggests that simple tag-based extraction effectively preserves semantic content structure.

The progressive decline from ROUGE-1 to ROUGE-L scores indicates increasing difficulty in maintaining phrase-level and structural coherence. Ollama's poor ROUGE performance across all variants suggests significant content loss in LLM-based extraction.

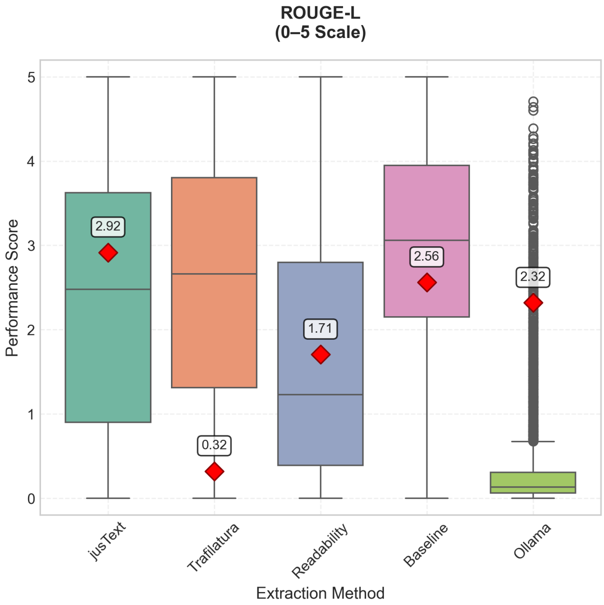

Figure 1: Boxplot comparison of ROUGE-L scores across different extraction methods.

5.2. F1-Score Analysis

The F1-Score distributions reveal the precision-recall balance achieved by each method. Most methods cluster around high F1-scores (4.5-5.0), indicating strong overall performance in content identification and extraction accuracy.

Ollama achieves the highest F1-Score with a mean of 4.74 ± 0.58, suggesting excellent precision-recall balance for retained content. Trafilatura demonstrates consistent performance at 4.73 ± 1.01, and Baseline shows strong F1 performance at 4.67 ± 0.89.

Readability maintains competitive F1 scores at 4.58 ± 1.22, while jusText exhibits more variable performance at 4.12 ± 1.86.

Key F1-Score Insights. The high F1-scores across most methods indicate that when content is extracted, it tends to be relevant and accurate. The apparent contradiction between Ollama's poor ROUGE performance and high F1-Score suggests that while Ollama extracts less content overall, what it does extract is highly relevant.

jusText's lower F1-Score reflects its more conservative filtering approach, which may occasionally misclassify relevant content.

5.3. Content-Noise-Ratio Performance

The Content-Noise-Ratio metric evaluates each method's ability to separate meaningful content from boilerplate, advertisements, and navigation elements.

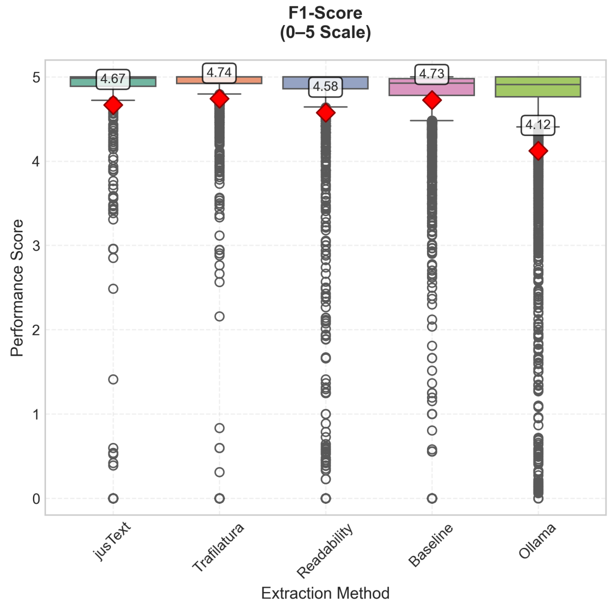

Figure 2: Boxplot comparison of F1 scores across different extraction methods.

Ollama achieves the highest noise suppression with a mean of 4.52 ± 0.80. Baseline shows strong noise filtering at 4.23 ± 0.69, and Trafilatura demonstrates excellent noise suppression at 4.21 ± 1.08.

Readability provides moderate noise filtering at 3.96 ± 1.24, while jusText shows the most variable noise suppression at 3.67 ± 1.71.

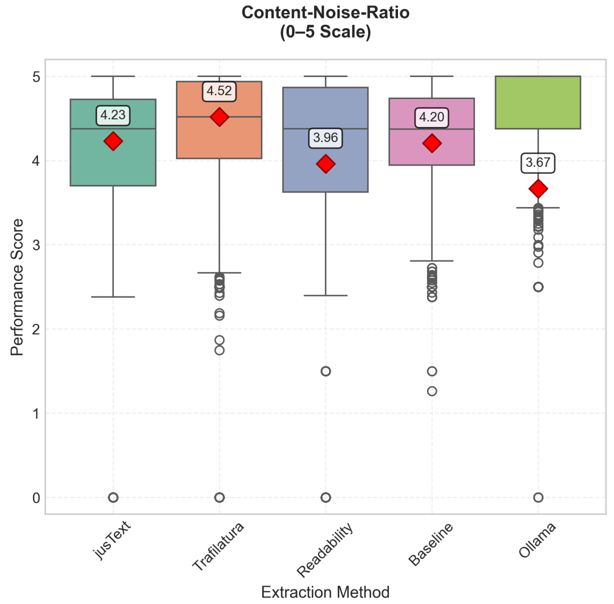

Figure 3: Boxplot comparison of Content-Noise-Ratio across different extraction methods.

Interpretation. Ollama's superior noise suppression, combined with its poor ROUGE scores, suggests it may be over-filtering content. Traditional methods (Trafilatura, Baseline) achieve good noise suppression while maintaining content coverage.

jusText's variable performance reflects its rule-based approach, which may struggle with diverse business website structures.

5.4. Semantic-Coherence Assessment

Semantic-Coherence scores measure the logical flow and contextual integrity of extracted content.

Baseline achieves the highest coherence scores with a mean of 4.49 ± 0.61. Ollama shows strong but variable coherence at 4.44 ± 1.22, and Trafilatura maintains consistent coherence at 4.36 ± 0.85.

Readability provides moderate coherence at 3.82 ± 1.25, while jusText exhibits the most variable coherence at 3.54 ± 1.87.

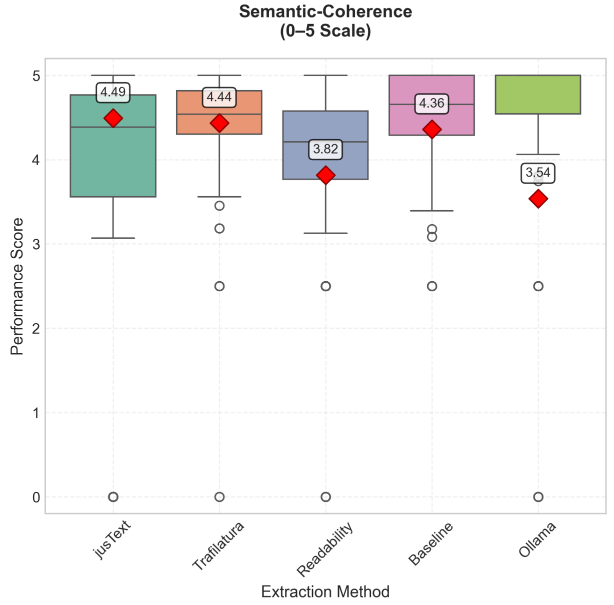

Figure 4: Boxplot comparison of Semantic Coherence across different extraction methods.

Interpretation.

Notable Findings. Baseline's superior coherence, combined with strong ROUGE scores, validates simple extraction approaches for well-structured content. Ollama's high coherence but variable performance suggests inconsistent extraction quality.

Traditional rule-based methods maintain reasonable coherence across diverse content types.

5.5. Holistic Performance Comparison

The comprehensive analysis reveals distinct performance patterns as seen in Table 2.

Method	ROUGE-1	ROUGE-2	ROUGE-L	F1-Score	C-N Ratio	Sem-Coherence
Baseline	3.04 ± 1.24	2.81 ± 1.29	2.92 ± 1.28	4.67 ± 0.89	4.23 ± 0.69	4.49 ± 0.61
Trafilatura	2.64 ± 1.42	2.51 ± 1.48	2.56 ± 1.45	4.73 ± 1.01	4.21 ± 1.08	4.36 ± 0.85
jusText	2.42 ± 1.55	2.28 ± 1.57	2.32 ± 1.56	4.12 ± 1.86	3.67 ± 1.71	3.54 ± 1.87
Readability	1.77 ± 1.48	1.64 ± 1.54	1.71 ± 1.52	4.58 ± 1.22	3.96 ± 1.24	3.82 ± 1.25
Ollama	0.40 ± 0.61	0.22 ± 0.56	0.32 ± 0.58	4.74 ± 0.58	4.52 ± 0.80	4.44 ± 1.22

Table 2: Higher scores indicate better performance across all metrics

5.6. Method Selection Guidelines

The comprehensive evaluation reveals that optimal method selection depends on specific application requirements.

For Content Coverage and Semantic Fidelity. Baseline and Trafilatura offer the best balance of content preservation across all ROUGE metrics. These methods maintain semantic structure while providing reasonable noise suppression.

For Precision-Focused Applications. Ollama provides the cleanest output with minimal noise, though at the cost of content coverage. Trafilatura offers a good balance between precision and recall.

For Consistent Production Deployment. Baseline and Trafilatura provide the most reliable performance with lower variability. jusText shows high variability, requiring careful parameter tuning.

For Experimental Applications. Ollama offers flexibility but requires optimization for content coverage. Readability proves least suitable for business website extraction.

5.7. ROUGE Variants Analysis Summary

The analysis of all ROUGE variants provides important insights. First, all ROUGE variants show consistent method ranking, validating our findings. Second, scores decrease progressively from ROUGE-1 to ROUGE-L, indicating increasing extraction complexity.

Third, simple tag-based extraction outperforms complex methods across all semantic metrics, demonstrating baseline superiority. Finally, Ollama's poor ROUGE performance suggests significant content loss in neural extraction approaches, highlighting LLM limitations.

This comprehensive ROUGE analysis demonstrates that while sophisticated extraction methods offer advanced features, simple approaches can achieve superior semantic content preservation for business websites. The consistent patterns across all ROUGE variants strengthen the reliability of our method selection recommendations.

6. Discussion

The comprehensive evaluation reveals that each content extraction method exhibits distinct performance characteristics and trade-offs, particularly when applied to structurally diverse German business websites. The multi-dimensional analysis across all ROUGE variants, precision-recall measures, and content quality indicators provides nuanced insights into method selection and optimization strategies.

6.1. Performance Patterns and Method Characteristics

Baseline Method Dominance. The Baseline approach's superior performance across all ROUGE variants (ROUGE-1: 3.04, ROUGE-2: 2.81, ROUGE-L: 2.92) demonstrates that simple tag-based extraction (`<p>`, `<title>`, `<div>`) effectively preserves semantic content structure. This finding challenges the assumption that sophisticated algorithms are necessary for effective content extraction.

The method's consistent high performance across semantic metrics, combined with strong coherence scores (4.49), validates minimal filtering approaches for well-structured business content.

Trafilatura's Balanced Excellence. Trafilatura emerges as the most well-rounded performer, consistently ranking second across all ROUGE variants while excelling in noise suppression (4.21) and maintaining semantic coherence (4.36). Its sophisticated content detection algorithms effectively balance content preservation with boilerplate removal, making it ideal for applications requiring both semantic fidelity and content quality.

Ollama's Precision-Coverage Trade-off. The LLM-based approach presents a fascinating paradox—achieving the highest F1-Score (4.74) and noise suppression (4.52) while exhibiting the lowest performance across all ROUGE variants. This suggests that Ollama extracts significantly less content but with higher precision and relevance.

The poor ROUGE performance (ROUGE-1: 0.40, ROUGE-2: 0.22, ROUGE-L: 0.32) indicates substantial content loss, making it suitable only for applications prioritizing precision over coverage.

jusText's Variable Performance. jusText demonstrates moderate but inconsistent performance across all metrics, with particularly high standard deviations (ROUGE-L: 2.32 ± 1.56, F1-Score: 4.12 ± 1.86). This variability reflects its rule-based approach's sensitivity to diverse website structures, requiring careful parameter tuning for optimal performance.

Readability's Domain Limitations. Originally designed for news content, Readability consistently underperforms across all ROUGE variants, confirming its unsuitability for business websites with complex layouts and multiple content areas.

6.2. ROUGE Variants Insights

Progressive Complexity. The consistent decline from ROUGE-1 to ROUGE-L scores across all methods indicates increasing difficulty in maintaining phrase-level and structural coherence. This pattern suggests that while methods can preserve individual words, maintaining semantic relationships and content structure presents greater challenges.

Method Consistency. The identical ranking across all ROUGE variants (Baseline > Trafilatura > jusText > Readability > Ollama) validates the reliability of our findings and suggests that methods' relative strengths are consistent across different aspects of semantic preservation.

Structural Preservation Challenges. The gap between ROUGE-1 and ROUGE-L scores highlights the difficulty in maintaining content order and structural coherence, particularly important for business websites where information hierarchy affects user comprehension.

6.3. Application-Specific Recommendations

For Comprehensive Content Analysis. Baseline and Trafilatura provide optimal semantic preservation across all ROUGE variants. These methods maintain content structure essential for downstream NLP tasks like summarization and clustering.

For High-Precision Information Extraction. Ollama offers superior precision for critical information extraction where content loss is acceptable. Trafilatura provides the best balance between precision and semantic coverage.

For Production Systems. Baseline offers the most reliable performance with minimal computational overhead. Trafilatura provides advanced features with acceptable consistency for production deployment.

6.4. Implications for Content Extraction Research

Simplicity vs. Sophistication. The Baseline method's superior performance across semantic metrics challenges the assumption that complex algorithms are necessary for effective content extraction. This suggests that understanding content structure may be more important than algorithmic sophistication.

Domain-Specific Optimization. The poor performance of news-optimized tools (Readability) on business websites emphasizes the importance of domain-specific evaluation and optimization.

LLM Extraction Limitations. Ollama's poor ROUGE performance highlights current limitations in neural extraction approaches, suggesting that prompt engineering and model fine-tuning are crucial for effective LLM-based content extraction.

7. Practical Recommendations

Based on the comprehensive multi-metric evaluation across 719 German business web pages, we provide the following practical guidelines for selecting content extraction methods:

- **For broad semantic coverage and structural fidelity:** Use the Baseline method (simple tag-based extraction) or Trafilatura. Both effectively maintain content structure, achieving the highest scores across ROUGE and coherence metrics.
- **For scenarios prioritizing high precision and minimal noise:** Consider Ollama's LLM-based extraction, which yields the cleanest output and strongest noise suppression. Be prepared for reduced content coverage and higher computational costs.
- **For balanced extraction with robust noise filtering:** Trafilatura offers a strong compromise, combining semantic retention with boilerplate removal, making it ideal for most business website pipelines.
- **For exploratory research involving diverse layouts:** jusText is valuable due to its language-aware heuristics but needs careful parameter tuning to avoid over-filtering or content loss.
- **For complex, multi-focal business websites:** Avoid Readability, as it is tuned for news articles and often misclassifies key sections in business domains.
- **For production pipelines with resource constraints:** Prefer the Baseline method for its consistency, efficiency, and suitability as a default in large-scale processing.

These recommendations are grounded in our empirical results and can help tailor extraction strategies to diverse business data mining, NLP, or web scraping objectives.

8. Conclusion

This study presented a comprehensive evaluation of five content extraction methods on German business websites using a multi-dimensional framework that included ROUGE variants, precision-recall metrics, and content quality indicators.

Primary Findings. Our findings reveal that the simple tag-based Baseline approach consistently outperforms more complex methods across all ROUGE metrics, demonstrating that minimal HTML filtering can effectively preserve semantic content and structure while remaining computationally efficient. Trafilatura offers the best balance between semantic fidelity and noise suppression, making it well-suited for most business website extraction tasks. Ollama's LLM-based extraction, while achieving the highest noise reduction and F1-scores, suffers from significant content loss, highlighting the trade-off between precision and coverage in neural approaches. jusText shows variable performance dependent on parameter tuning, and Readability, designed for news articles, proves less suitable for the heterogeneous structures of business websites.

Methodological Contributions. Beyond empirical findings, this work contributes a robust evaluation methodology and domain-specific insights that inform method selection for practical applications in business data mining and NLP. It underscores the importance of aligning extraction strategies with content characteristics and downstream task needs, challenging the assumption that greater algorithmic sophistication always yields better results.

Future Research Directions. Future research should explore hybrid models combining rule-based and neural techniques, develop domain-adaptive extraction systems tailored to business content, and extend evaluations to multilingual or sector-specific contexts. Open annotated datasets for business websites would further support reproducibility and accelerate advancements in this area.

Practical Implications. Overall, our study provides actionable guidance for selecting and optimizing content extraction approaches, emphasizing that effective solutions arise from a nuanced understanding of both method capabilities and application requirements.

References

[1] A. Barbaresi. Trafilatura: A web scraping library and command-line tool for text discovery and extraction. In *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 122–131. Association for Computational Linguistics, 2021.

[2] S. Bird, E. Klein, and E. Loper. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, 2009.

[3] T. Gottron. Evaluating content extraction on html documents. In *Proceedings of the 2nd International Conference on Internet Technologies and Applications*, pages 123–132. Glyndŵr University, 2007.

[4] A. Jain, K. Nandakumar, and A. Ross. Score normalization in multimodal biometric systems. *Pattern Recognition*, 38(12):2270–2285, 2005.

[5] C. Kohlschütter, P. Fankhauser, and W. Nejdl. Boilerplate detection using shallow text features. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, pages 441–450. ACM Press, 2010.

[6] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81. Association for Computational Linguistics, 2004.

[7] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[8] P. M. McCarthy and S. Jarvis. Mtld, vocd-d, and hd-d: A validation study of sophisticated approaches to lexical diversity assessment. *Behavior Research Methods*, 42(2):381–392, 2010.

[9] S. Morbieu, J. Kramer, J. G. Breslin, and M. d'Aquin. Main content extraction from web pages based on semantic similarity and bert. In *Proceedings of the 19th IEEE International Conference on Machine Learning and Applications*, pages 1283–1290. IEEE Computer Society, 2020.

[10] M. E. Peters and D. Lecocq. Content extraction using diverse feature sets. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 89–90. ACM Press, 2013.

[11] L. Richardson. Beautiful soup: We called him tortoise because he taught us. https://www.crummy.com/software/BeautifulSoup/, 2007.

[12] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.

[13] T. Vogels, O.-E. Ganea, and C. Eickhoff. Web2text: Deep structured boilerplate removal. *arXiv preprint arXiv:1801.02607*, 2018.

[14] T. Weninger, W. H. Hsu, and J. Han. Cetr: Content extraction via tag ratios. In *Proceedings of the 19th International Conference on World Wide Web*, pages 971–980. ACM Press, 2010.