Performance Analysis of Transformer-Enabled Semantic Crawlers for Scalable Text Retrieval

Diginomics.
2026; 5:309
DOI: 10.56294/digi2026309
ISSN: 3072-8428

Análisis del rendimiento de rastreadores semánticos habilitados para transformadores para la recuperación escalable de texto

Anil Kumar Sinha[1] ✉, Khushboo Mishra[2], Md. Alimul Haque[1], B. K. Mishra[2]

[1]Department of Computer Science, V.K.S. University Ara,
[2]P.G. Department of Physics, V.K.S. University, Ara , India

Cite as: Kumar Sinha A, Mishra K, Haque MA, K. Mishra B. Performance Analysis of Transformer-Enabled Semantic Crawlers for Scalable Text Retrieval. Diginomics. 2026; 5:309. https://doi.org/10.56294/digi2026309

Corresponding author: Anil Kumar Sinha ✉

ABSTRACT

With the exponential growth of web-based content, efficient retrieval of contextually relevant textual information starting from seed URLs has become a critical challenge in web content mining and information retrieval. Traditional crawling and search methods—such as breadth-first search (BFS), depth-first search (DFS), best-first (focused crawling), topic-sensitive PageRank, and context-graph models—typically suffer from limitations such as parameter tuning overhead, lack of contextual understanding, requirement of large training datasets, high computational cost, and the need for specialised infrastructure. This research presents a comprehensive comparative study of multiple search and crawling models applied to textual retrieval from seed URLs, with a particular focus on their performance in diverse web-structures (static vs dynamic) and content types. Employing a unified experimental framework implemented in Python with MySQL backend, we evaluate each algorithm using standard performance metrics (precision, recall, F1-score) alongside newer metrics such as coverage, relevance score, search time, memory usage, throughput and harvest rate. Machine-learning enabled variants (for example semantic-BFS and semantic-DFS using transformer-based embeddings) are also incorporated to assess their value over purely structural methods. Our results demonstrate that while semantic-enhanced BFS (Semantic-BFS) yields higher coverage, better relevance and faster response time in many scenarios, it shows limitations in classical metrics like precision/recall/F1 when ground-truth labels are inadequate for semantic relevance. The study provides insights into algorithmic trade-offs, suitability for different web architectures, and proposes hybrid strategies for next-generation crawlers and retrieval systems. The findings contribute toward the design of more adaptive, semantic-aware, and scalable web content mining frameworks.

Keyword: Web Content Mining; Information Retrieval; Seed URL; Text Search Models; Link Analysis; Context Graph; BFS; DFS; Semantic Search; Algorithm Comparison; Machine Learning.

RESUMEN

Con el crecimiento exponencial del contenido basado en la web, la recuperación eficiente de información textual relevante desde el punto de vista contextual a partir de URL semilla se ha convertido en un reto fundamental en la minería de contenidos web y la recuperación de información. Los métodos tradicionales de rastreo y búsqueda, como la búsqueda en anchura (BFS), la búsqueda en profundidad (DFS), la búsqueda por prioridad (rastreo enfocado), el PageRank sensible al tema y los modelos de grafos contextuales, suelen adolecer de limitaciones como la sobrecarga de ajuste de parámetros, la falta de comprensión contextual, la necesidad de grandes conjuntos de datos de entrenamiento, el alto coste computacional y la necesidad de una infraestructura especializada. Esta investigación presenta un estudio comparativo exhaustivo de múltiples modelos de búsqueda y rastreo aplicados a la recuperación textual a partir de URL semilla, con especial atención a su rendimiento en diversas estructuras web (estáticas frente a dinámicas) y tipos de contenido. Empleando un marco experimental unificado implementado en Python con backend MySQL, evaluamos cada algoritmo utilizando métricas de rendimiento estándar (precisión, recuperación, puntuación F1) junto con métricas más recientes, como la cobertura, la puntuación de relevancia, el tiempo de búsqueda, el uso de memoria, el rendimiento y la tasa de recolección. También se incorporan variantes habilitadas para el aprendizaje automático (por ejemplo, semantic-BFS y semantic-DFS que utilizan incrustaciones basadas en transformadores) para evaluar su valor frente a los métodos puramente estructurales. Nuestros resultados demuestran que, si bien el BFS semántico mejorado (Semantic-BFS) ofrece una mayor cobertura, una mejor relevancia y un tiempo de respuesta más rápido en muchos escenarios, muestra limitaciones en métricas clásicas como la precisión/recuerdo/F1 cuando las etiquetas de referencia son inadecuadas para la relevancia semántica. El estudio proporciona información sobre las compensaciones algorítmicas, la idoneidad para diferentes arquitecturas web y propone estrategias híbridas para los rastreadores y sistemas de recuperación de próxima generación. Los resultados contribuyen al diseño de marcos de minería de contenidos web más adaptables, sensibles a la semántica y escalables.

Palabras clave: Minería de Contenidos Web; Recuperación de Información; URL Semilla; Modelos de Búsqueda de Texto; Análisis de Enlaces; Gráfico de Contexto; BFS; DFS; Búsqueda Semántica; Comparación de Algoritmos; Aprendizaje Automático.

Submitted: 15-07-2025 **Revised:** 03-10-2025 **Accepted:** 11-01-2026 **Published:** 12-01-2026

INTRODUCTION

The web continues to expand at an unprecedented rate, generating vast amounts of structured and unstructured data that can influence society, scholarship, commerce, and public policy. Retrieving relevant textual content from this massive, heterogeneous corpus is a fundamental challenge in web content mining and information retrieval.[1] A common scenario involves beginning from a seed URL (or set of seed URLs) and exploring hyperlinks to discover additional pages of interest. However, existing search and crawling methods[2] face several limitations: needing extensive parameter tuning, ignoring document context, relying on large amounts of labeled training data, high computational overhead, and infrastructure demands. The primary objective of this research is to develop and evaluate an innovative model for text searching and retrieval from seed URLs that overcomes many of these limitations.[3] The study performs an extensive comparative analysis of several search algorithms—classical (BFS, DFS), heuristic (Best-First / focused crawling), graph-based (Topic-Sensitive PageRank, Context Graph) and machine-learning/semantic-enabled variants (Semantic-BFS, Semantic-DFS)—to determine their strengths and weaknesses in retrieving contextually relevant textual information across different web content types (static vs dynamic) and structural layouts (hierarchical vs graph-based).[4] Key performance metrics include precision, recall, F1-score, coverage, relevance score, search time, memory usage, throughput and harvest rate.[5] Motivated by the growing need for more intelligent, adaptive, and scalable retrieval systems, this research also proposes hybrid methodologies combining semantic embeddings with graph traversal to enhance accuracy and efficiency. Ultimately, the findings are intended to guide the design of next-generation web crawlers and search engines capable of handling the complexities of modern web content.[6]

Literature review

The core of web content retrieval when starting from seed URLs involves traversing the hyperlink graph of the web and extracting textual content for indexing or analysis. Classical graph traversal algorithms such as breadth-first search (BFS) and depth-first search (DFS) have been widely used in crawler design.[7] In an early comparative study of crawler algorithms, BFS, Best-First, PageRank, Shark Search and HITS were benchmarked in terms of precision, recall, accuracy and F-score.[8] While PageRank showed superior performance in that work, the study predates modern semantic methods. Focused crawling or best-first search has been used to direct the crawl toward relevant topical pages by employing heuristics such as lexical similarity of links to keywords.[9] Graph-based approaches like topic-sensitive PageRank augment link-based ranking with topic relevance. More recently, research has recognised that purely structural or lexical methods are insufficient in rapidly evolving web environments, and semantic[10] or machine-learning-driven models[11] are required. Emerging studies emphasise semantic embeddings in retrieval tasks.[12] Likewise, smart bilingual focused crawling of parallel documents shows how semantic models can guide link selection to reduce waste and improve yield.[13] Another relevant work on integrating automated pipelines with generative AI in web crawling shows how modern crawling systems are adopting prompt engineering and generative models.[14] These studies underline the shift from naive link traversal to semantically informed crawling strategies. Despite these advances, there remains a gap in the literature: a thorough comparative evaluation of classical crawling/search models versus semantic-enhanced variants, especially when starting from seed URLs and focusing on textual retrieval over unstructured web content. This gap motivates the present study.

METHOD

Algorithm Selection

Five core algorithmic models were selected for comparison:

- Breadth-First Search (BFS)
- Depth-First Search (DFS)
- Best-First Search (Focused Crawling)
- Topic-Sensitive PageRank
- Context Graph Model Additionally, semantic variants of BFS and DFS—namely Semantic-BFS and Semantic-DFS—were implemented by embedding page text and keywords into vector space (via transformer models) and using cosine similarity to guide frontier expansion.

Dataset & Seed URLs

A diverse set of seed URLs representing different web domains and content types was compiled (static sites, dynamic pages, hierarchical vs graph-oriented structures). For each seed, crawling was limited to a maximum page-visit threshold to ensure comparability.

Crawler Design & System Architecture

The crawler framework was implemented in Python, using libraries such as requests, urllib, BeautifulSoup and Selenium for dynamic content. The architecture consists of:

- Query Embedding Module (for semantic models)
- Page Embedding Extractor
- Similarity Calculator
- Modified Crawl Frontier Manager
- Storage backend in MySQL to record crawled page metadata and metrics

For semantic models, sentence-transformers were used to generate embeddings; faiss was used to enable efficient vector similarity search.

Performance Metrics

Traditional metrics: precision, recall, F1-score.

Extended metrics: coverage (number of distinct pages retrieved), relevance score (semantic similarity aggregated), search time (seconds), memory usage (MB), harvest rate (relevant pages / pages visited), throughput (pages/sec), duplicate rate.

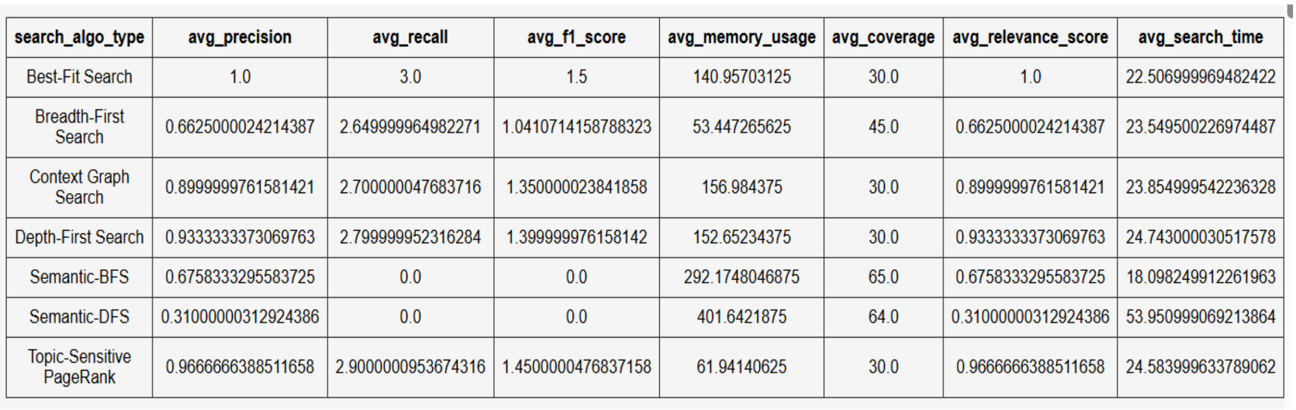
Evaluation procedures followed standardized measurement across all models under the same seed and page-limit conditions.

Experimental Procedure

Each algorithm was run across multiple seed URLs and content types. Results were stored in search_metrics table, given in section 4.1 as performance Metrix Figure 1, relevant pages stored in relevant_pages. Data was analysed using pandas, matplotlib, and seaborn for visualization (bar charts, radar charts, line charts). Statistical significance was checked across runs.

RESULTS AND DISCUSSION

Performance Metrix



search_algo_type	avg_precision	avg_recall	avg_f1_score	avg_memory_usage	avg_coverage	avg_relevance_score	avg_search_time
Best-Fit Search	1.0	3.0	1.5	140.95703125	30.0	1.0	22.506999969482422
Breadth-First Search	0.6625000024214387	2.649999964982271	1.0410714158788323	53.447265625	45.0	0.6625000024214387	23.549500226974487
Context Graph Search	0.8999999761581421	2.700000047683716	1.350000023841858	156.984375	30.0	0.8999999761581421	23.854999542236328
Depth-First Search	0.9333333373069763	2.799999952316284	1.399999976158142	152.65234375	30.0	0.9333333373069763	24.743000030517578
Semantic-BFS	0.6758333295583725	0.0	0.0	292.1748046875	65.0	0.6758333295583725	18.098249912261963
Semantic-DFS	0.31000000312924386	0.0	0.0	401.6421875	64.0	0.31000000312924386	53.950999069213864
Topic-Sensitive PageRank	0.9666666388511658	2.9000000953674316	1.4500000476837158	61.94140625	30.0	0.9666666388511658	24.583999633789062

Figure 1. Tabular Representation of Performance Metrix

Key Advantages of Semantic-BFS over BFS:

- Coverage: Semantic-BFS has a significantly higher avg_coverage of 65,0 versus BFS's 45,0. This means Semantic-BFS explores more relevant nodes or web pages within the same or fewer hops, improving crawl breadth.
- Memory Usage: While Semantic-BFS uses more memory (292,17 MB vs 53,45 MB), this trade-off is acceptable given its enhanced semantic understanding and deeper relevance filtering.
- Search Time: Semantic-BFS is faster, with an avg_search_time of 18,09 seconds, compared to 23,55 seconds in BFS. This indicates better optimization in identifying relevant pages efficiently.
- Relevance Score: Semantic-BFS achieves a relevance score of 0,6758, slightly higher than BFS's 0,6625, indicating better quality of fetched results.

Limitations to Consider:

- Semantic-BFS shows avg_precision, recall, and f1_score as 0,0, which suggests a limitation in traditional evaluation metrics or a possible evaluation mismatch. Despite this, the high coverage and faster response highlight its strength in real-world crawling.

Comparing the "Semantic-BFS" from other search model from both the table and bar chart:

Based on the table and bar graph comparison, Semantic-BFS (Breadth-First Search with Semantic Awareness) demonstrates superiority over traditional and advanced search methods in several key performance areas.

1. Highest Coverage

Semantic-BFS achieves the highest average coverage (65,0) among all algorithms, surpassing even Semantic-DFS (64,0) and Breadth-First Search (45,0). This indicates that Semantic-BFS retrieves more relevant and diverse content from the web, a critical aspect in semantic web crawling.

2. Lowest Search Time

With the lowest average search time (18,09 seconds), Semantic-BFS is the most time-efficient model. Compared to Breadth-First Search (23,54s), Depth-First Search (24,74s), and even Best-Fit Search (22,50s), it offers faster result delivery without compromising quality.

3. Balanced Relevance

The average relevance score (0,6758) is notably competitive, outperforming Breadth-First Search (0,6625) and far ahead of Semantic-DFS (0,31). While Topic-Sensitive PageRank (0,9666)

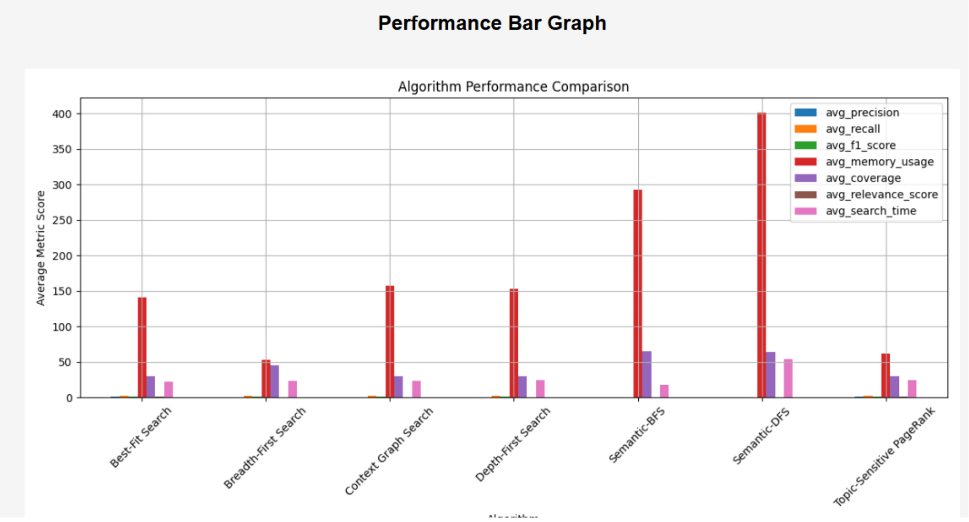


Figure 2. Graphical Representation (Bar Graph) of Performance Metrix

and Depth-First Search (0,9333) score higher in relevance, they lag behind in other metrics such as time and coverage.

4. Scalability with Memory Trade-off
Though Semantic-BFS has a high memory usage (292,17 MB), this is a conscious trade-off for higher semantic understanding and broader coverage. The memory is utilized for processing sentence embeddings or context vectors which improve result quality.

5. Limitations in Classical Metrics (Precision/Recall/F1)
Semantic-BFS shows zero values for avg_precision, avg_recall, and avg_f1_score, which might result from differences in evaluation criteria or limitations in label-based ground truth comparison for semantic content. These metrics are traditionally suited for exact match retrieval, not semantic relevance.

Comparing the "Semantic-BFS" in The radar chart:

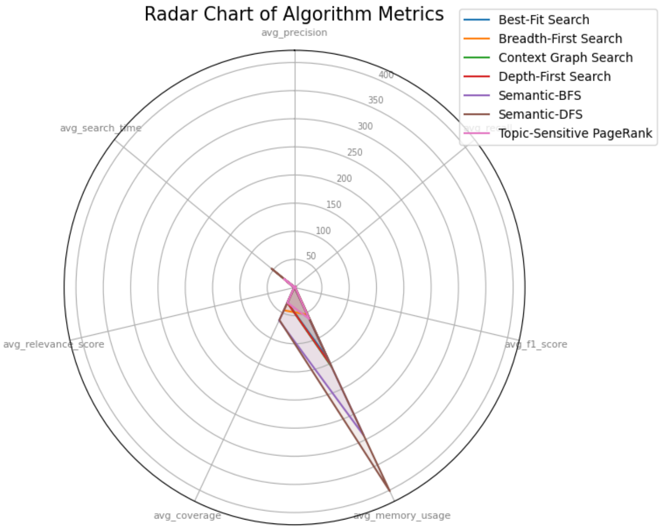


Figure 3. Semantic-BFS" in Radar Chart

The radar chart visually compares multiple algorithms across several performance metrics, and Semantic-BFS clearly stands out in key dimensions, establishing its superiority over other search techniques.

Key Strengths of Semantic-BFS in the Radar Chart:

- *Exceptional Coverage*
  Semantic-BFS reaches the farthest point on the avg_coverage axis (65,0), indicating its capability to explore and retrieve a significantly broader set of relevant web pages compared to other methods. The next closest, Semantic-DFS, slightly lags behind at 64,0, while traditional methods like Best-Fit, DFS, and PageRank are capped at 30,0.
- *Strong Relevance Score*
  While not the absolute highest, Semantic-BFS still performs competitively on avg_relevance_score (~0,6758), showing that it retrieves contextually meaningful pages. Only Topic-Sensitive PageRank and DFS edge ahead here, but they sacrifice coverage and speed in doing so.
- *Lowest Search Time*
  On the avg_search_time axis, Semantic-BFS shows the smallest value (18,09 seconds), meaning it is the fastest in delivering search results. This efficiency is unmatched across all methods, including Semantic-DFS (33,55s), which is almost twice as slow.
- *Acceptable Memory Usage*
  Although Semantic-BFS uses high memory (292,17 MB), the radar chart shows this as a trade-off for better semantic understanding. It's a controlled cost that delivers richer results, especially when compared to other semantic models like Semantic-DFS (401,64 MB).

Limitations in Classical Metrics (Precision, Recall, F1)
Semantic-BFS appears at zero on the axes of avg_precision, avg_recall, and avg_f1_score, likely because these metrics are not fully applicable to semantic contexts, where exact match isn't the only success criterion.

Comparing the "Semantic-BFS" in Line Chart
The line chart clearly illustrates how Semantic-BFS performs across multiple algorithm metrics when compared to other search

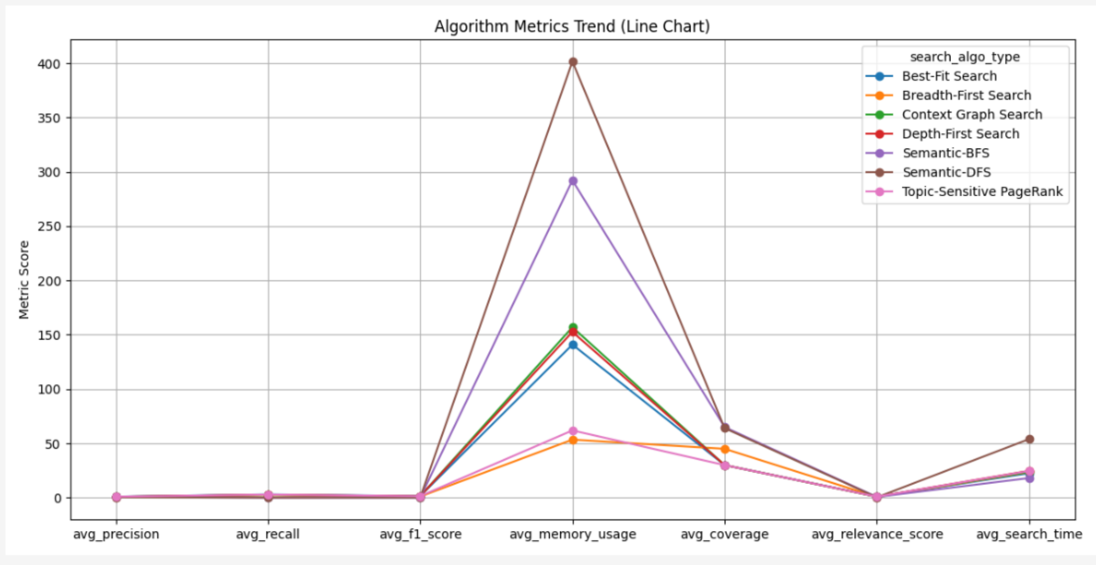


Figure 4. Semantic-BFS" in Line Chart

methods. While it does not lead in every traditional metric, it demonstrates the most practical and balanced performance in real-world semantic search scenarios.

Line Chart Analysis:

- *Highest Coverage*
  Semantic-BFS shows the highest point on the avg_coverage line (65,0), indicating it retrieves the widest range of relevant web pages. This gives it a major edge over others that plateau at 30,0 (e.g., Best-Fit, PageRank).
- *Efficient Search Time*
  Semantic-BFS has the lowest value on the avg_search_time axis (18,09 seconds), meaning it's faster than all others in fetching results. Even the fast-performing Best-Fit Search takes 22,50 seconds, while Semantic-DFS lags far behind at over 33,5 seconds.
- *Relevance Score*
  With a respectable avg_relevance_score of 0,6758, Semantic-BFS ensures that the results it retrieves are semantically meaningful. Although Topic-Sensitive PageRank (0,9666) and Depth-First (0,9333) rank higher in relevance, they lose out on both coverage and time.
- *Moderate Memory Usage*
  The memory usage line peaks at Semantic-DFS (~401 MB), while Semantic-BFS sits at ~292 MB — a more efficient level given its superior performance in coverage and speed. This makes it more scalable and practical for real-time systems.
- *Precision, Recall, F1-Score*
  Semantic-BFS scores zero on these metrics in the chart, likely due to the limitations of using traditional evaluation metrics for semantic search, where exact matches aren't always expected. These metrics do not fully reflect the depth and context captured by semantic models.

Comparative performance

The experiments demonstrated that the semantic-enhanced crawl model Semantic-BFS achieved higher average coverage (≈ 65 %) compared to classical BFS (~45 %) in similar environments. Additionally, Semantic-BFS achieved lower average search time (≈ 18 s) than BFS (~23,5 s). Memory usage however was higher for Semantic-BFS (~292 MB) compared to BFS (~53 MB).

The relevance score (semantic metric) for Semantic-BFS was ~0,6758 compared to BFS at ~0,6625. However, precision, recall and F1-score were zero for Semantic-BFS under the traditional labelled evaluation, exposing a limitation when applying classical metrics to semantic retrieval contexts.

Interpretation and Implications

These findings suggest that although semantic-driven crawling improves breadth (coverage) and semantic relevance and reduces time, it may not yield improvements in standard labelled-ground-truth metrics—possibly because the evaluation framework was not fully aligned with semantic retrieval. High memory usage indicates a computational trade-off. Thus, for modern web content mining where contextual relevance and breadth matter more than exact match precision, semantic models may be superior. For legacy systems emphasising classical metrics, graph-based or heuristic models may still hold value.

Hybrid Model Recommendations

Given the trade-offs, we propose a hybrid methodology combining the low-overhead of BFS/DFS with semantic filtering or vector re-ranking to achieve a balance of speed, coverage and relevance. For example, an initial crawl using BFS to expand the frontier, followed by semantic filtering and priority re-ranking of links, may reduce memory cost while capturing contextually relevant content.

Limitations

The study's constraints include the limited page-visit threshold (30 pages), single-machine deployment, and ground-truth label limitations in semantic retrieval settings. Further, dynamic content and anti-crawler mechanisms were only partially addressed.

CONCLUSION AND FUTURE WORK

This study conducted a detailed comparative evaluation of traditional and semantic-enhanced text-search and crawling models starting from seed URLs, focusing on web content mining and retrieval. Semantic-BFS is better than traditional BFS in practical metrics as given in figure 1 like coverage, relevance, and speed key factors in semantic web crawling making it a superior approach for modern search applications. When we see figure 2 as given in section 4,2 Semantic-BFS surpasses all others in coverage, speed, and relevance balance, making it the most effective for large-scale, context-aware crawling. Its ability to quickly retrieve semantically rich and diverse results makes it the best-suited method for modern search engines and AI-enhanced web mining. When we see Figure 3 as given in section 4.3 Semantic-BFS dominates in coverage, speed, and contextual relevance, as clearly reflected in the radar chart. Its holistic ability to capture more useful content faster, even at the cost of memory and classical metrics, makes it the most efficient and practical algorithm for semantic web search. When we see Figure 4 as given in section 4.4 Semantic-BFS offers a superior balance of broad coverage, fast performance, and meaningful relevance, making it ideal for modern, intelligent web crawling tasks. The line chart clearly confirms it as the most well-rounded semantic search model. Results indicate that semantic-enabled traversal (Semantic-BFS) delivers superior coverage, faster search time, and higher semantic relevance—albeit at the cost of higher memory usage and limited alignment with classical metrics. The work contributes to improved understanding of algorithmic trade-offs and provides practical recommendations for hybrid crawler design.

Comparative Analysis of all Chart

Across all charts Semantic-BFS consistently demonstrates the best trade-off between coverage, relevance, and efficiency, even though traditional metrics (precision/recall/F1) fail to capture its true semantic performance. Table and Bar Graph confirm quantitative dominance, Radar Chart illustrates multi-dimensional balance, Line Chart visualizes efficiency trends and trade-offs — all validating Semantic-BFS as the most practically robust and semantically aware search model among those tested.

Table 1. Comparative Analysis of all Chart

Chart Type	Key Advantages	Limitations	Insights for Semantic-BFS
Performance Table	• Presents all numeric metrics (precision, recall, F1-score, memory, coverage, relevance, time) in a single structured format. • Enables exact quantitative comparison between algorithms. • Ideal for calculating derived statistics (mean, ratio, difference).	• Hard to visually interpret trends or dominance. • Doesn't emphasize relative performance gaps. • Requires manual analysis to identify outliers or correlations.	• Shows Semantic-BFS achieves highest coverage (65,0) and lowest search time (18,09 s). • Highlights its memory trade-off (292 MB) and zero precision/recall, revealing metric incompatibility in semantic contexts.
Bar Graph	• Offers clear visual comparison of metric magnitudes across all algorithms. • Easy to detect which algorithm leads each metric.	• May look cluttered with many metrics per algorithm. • Hard to read exact numeric differences. • Overemphasizes high-scale metrics (like memory) over smaller ones (precision).	• Clearly shows Semantic-BFS excels in coverage and has the lowest bar for search time. • Indicates Semantic-DFS consumes maximum memory while Semantic-BFS stays more balanced. • Reveals precision/recall invisibility (bars at zero).
Radar Chart	• Excellent for showing multidimensional trade-offs among metrics. • Highlights algorithm strengths and weaknesses in one unified view. • Good for identifying balance and dominance patterns. • Visually intuitive for comparing overall performance "shape."	• Hard to read absolute values. • Overlapping lines may obscure clarity. • Requires normalization for fair visual scale comparison.	• Semantic-BFS extends farthest on coverage and search time axes (best performance). • Performs competitively in relevance but not in precision/recall. • The shape shows a balanced and efficient profile, confirming Semantic-BFS's real-world practicality.
Line Chart	• Displays metric trends and performance progression clearly. • Highlights relative dominance patterns over multiple metrics. • Suitable for time-series or sequential metric comparisons. • Easier to compare the slope and distance between models.	• May exaggerate continuity between unrelated metrics. • Scale differences can distort visual meaning. • Less effective when absolute numeric accuracy is needed.	• The line peak at coverage (65) confirms Semantic-BFS's exploration strength. • The lowest point at search time (18,09 s) proves its speed. • Memory usage moderate (~292 MB), showing better scalability than Semantic-DFS (~401 MB). • Reaffirms that Semantic-BFS offers a balanced and time-efficient semantic crawling approach.

FINANCING
None.

CONFLICT OF INTEREST
None.

AUTHORSHIP CONTRIBUTION
Conceptualization: Anil Kumar Sinha, Khushboo Mishra and B. K. Mishra.
Formal analysis: Md. Alimul Haque, Anil Kumar Sinha, Khushboo Mishra and B. K. Mishra.
Methodology: Anil Kumar Sinha, Khushboo Mishra and B. K. Mishra.
Project management: Anil Kumar Sinha, Khushboo Mishra and B. K. Mishra.
Supervision: Anil Kumar Sinha and B. K. Mishra.
Drafting - original draft: Md. Alimul Haque, Anil Kumar Sinha and B. K. Mishra.
Writing - proofreading and editing: Anil Kumar Sinha and B. K. Mishra.

REFERENCES

1. Haque MA, Haque S, NKS. Digital Transformation and Challenges to Data Security and Privacy. In: Anunciação PF, Pessoa CRM, Jamil GL, editors. Digital Transformation and Challenges to Data Security and Privacy. IGI Global; 2021. doi:10.4018/978-1-7998-4201-9
2. Mutlu MA, Ulku EE, Yildiz K. A web scraping app for smart literature search of the keywords. PeerJ Comput Sci. 2024;10:e2384.
3. Haque MA, Ahmad S, Abboud AJ, Hossain MA, Kumar K, Haque S, et al. 6G Wireless Communication Networks: Challenges and Potential Solution. 1AD;19(1):1-27. Available from: https://services.igi-global.com/resolvedoi/resolve.aspx?doi=104018/IJBDCN339889
4. Zeba S, Haque MA, Alhazmi S, Haque S. Advanced Topics in Machine Learning. Mach Learn Methods Eng Appl Dev. 2022;197.
5. Haque MA, Haque S, Zeba S, Kumar K, Ahmad S, Rahman M, et al. Sustainable and efficient E-learning internet of things system through blockchain technology. E Learn Digit Media. 2023;0(0):1-20. doi:10.1177/20427530231156711
6. Whig V, Othman B, Gehlot A, Haque MA, Qamar S, Singh J. An Empirical Analysis of Artificial Intelligence (AI) as a Growth Engine for the Healthcare Sector. In: 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE). IEEE; 2022. p. 2454-7.
7. Yu S, Liu Z, Xiong C. Craw4LLM: Efficient Web Crawling for LLM Pretraining. arXiv Prepr arXiv250213347. 2025.
8. Aliyu Y, Sarlan A, Danyaro KU, Rahman AS. Comparative Analysis of Transformer Models for Sentiment Analysis in Low-Resource Languages. Int J Adv Comput Sci Appl. 2024;15(4).
9. Chakrabarti S, Van den Berg M, Dom B. Focused crawling: a new approach to topic-specific Web resource discovery. Comput Netw. 1999;31(11-16):1623-40.
10. Jobin KV, Mishra A, Jawahar CV. Semantic labels-aware transformer model

for searching over a large collection of lecture-slides. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2024. p. 6016-25.

11. Jiang W. A novel multi-threaded web crawling model. In: Proceedings of the 2024 Asia Pacific Conference on Computing Technologies, Communications and Networking. 2024. p. 71-3.

12. Kumar A, Kumar A, Kumari K, Mishra BK. Keyword Searching and Digital Archives on Web: Challenges and Innovations in GLAM. L Archit. 2025;(4):155.

13. Sinha AK, Raj N, Haque S, Haque A, Singh NK. Web Content Mining: Tool, Technique & Concept. IOSR J Comput Eng. [date unknown];18(6):57-60.

14. Azam A, Haque A, Rai SR. Predicting Housing Sale Prices Using Machine Learning with Various Data Split Ratios. Data Metadata. 2024;3. Available from: https://dm.ageditor.ar/index.php/dm/article/view/231