CONTEXT-AWARE AUTOMATIC TERM EXTRACTION USING USER BEHAVIOR AND TEXTUAL FEATURES

Vineetha Singa[1], Vigneshwar[2], Ashwin[2]

ABSTRACT

Automatic Term Extraction (ATE) is a pivotal task in Natural Language Processing (NLP) that seeks to identify semantically significant words and multi-word expressions from domain-specific corpora. Classical ATE methods, grounded in statistical heuristics such as Term Frequency–Inverse Document Frequency (TF-IDF) or purely linguistic pipelines, often fail to reflect the genuine importance of terms in interactive, user-driven environments. This paper proposes a context-aware ATE framework that synergistically integrates textual features—including term frequency, positional salience, syntactic patterns, and co-occurrence graphs—with implicit user behavior signals such as query history, dwell time, click-through rate, and session-level engagement metrics. A hybrid scoring function merges these heterogeneous signals into a unified relevance score for each candidate term. The system is evaluated on a benchmark corpus drawn from the ACM Digital Library and a proprietary e-learning dataset. Experimental results demonstrate that the proposed approach achieves a Precision of 0.87, Recall of 0.83, and F1-Score of 0.85, outperforming baseline TF-IDF and RAKE-based systems by 9–14% in F1. The framework is lightweight, language-agnostic at the feature level, and readily deployable in real-time digital library, search-engine, and recommendation-system pipelines.

KEYWORDS: Automatic Term Extraction, Context-Aware NLP, User Behavior Analysis, TF-IDF, Hybrid Scoring, Information Retrieval, Knowledge Discover

[1] Assistant Professor, Department of Master of Computer Application, Guru Nanak Dev Engineering College, Bidar, India
[2] 2nd Semester, Department of Master of Computer Applications, Guru Nanak Dev Engineering College, Bidar, India

How to Cite:
Vineetha Singa, Vigneshwar, Ashwin (2026), Context-Aware Automatic Term Extraction using User Behavior and Textual Features, International Educational Journal of Science & Engineering (IEJSE),Vol: 9, Special Issue, May 2026 619-626

1.INTRODUCTION

The Exponential growth of digital content—spanning scientific literature, e-commerce catalogs, social-media streams, and enterprise knowledge bases—has rendered manual indexing and annotation economically infeasible. Automatic Term Extraction (ATE) automates the discovery of domain-relevant terms from raw text, serving as a foundational step for downstream tasks such as ontology construction, document retrieval, text summarization, and knowledge-graph population [1].

Early ATE systems, exemplified by the C-Value/NC-Value method [2] and statistical TF-IDF variants, focused exclusively on surface-level textual signals. While computationally efficient, such methods suffer from domain drift—their term importance rankings become stale as user interests evolve and corpus vocabularies shift. More recent deep-learning approaches, leveraging pre-trained language models such as BERT [3] and SciBERT [4], substantially improve extraction quality but demand expensive fine-tuning and large annotated datasets, limiting their applicability in resource-constrained settings.

A complementary, underexplored dimension of term relevance is user behavior. In interactive systems—digital libraries, learning management systems, enterprise search portals—users continuously emit behavioral signals: they query for specific terms, dwell on certain documents, click particular hyperlinks, and revisit content selectively. These implicit signals encode collective

Research Paper

intelligence about term importance that purely textual models cannot capture [5].

This paper bridges the gap by proposing a Context-Aware ATE (CA-ATE) framework that jointly models textual features and user behavioral signals. The key contributions of this work are: (i) A multi-source feature engineering pipeline combining syntactic, positional, and co-occurrence graph features with session-level behavioral metrics. (ii) A lightweight hybrid scoring function that weighs textual and behavioral components adaptively. (iii) Comprehensive empirical evaluation on two publicly accessible benchmarks, demonstrating consistent improvements over competitive baselines. (iv) An analysis of privacy-preserving aggregation strategies for safe deployment of behavioral logging.

2. MATERIALS AND METHODS

Materials Used

The proposed Context-Aware Automatic Term Extraction (CA-ATE) framework utilizes both textual datasets and user behavioral datasets to improve the accuracy of term extraction. The following materials were used during the implementation and experimentation process.

2.1.1 Datasets

Two datasets were employed for evaluation and training purposes:

ACM-ATE Benchmark Dataset

This dataset contains 1,200 full-text research papers collected from the ACM Digital Library covering the domains of Artificial Intelligence, Systems, Human-Computer Interaction, Security, and Databases between 2018 and 2024. The dataset contains approximately 34,720 annotated technical term instances used as ground truth for evaluation.

ELearn-Behav Dataset

A proprietary e-learning interaction dataset containing approximately 380,000 anonymized user sessions from 4,200 students interacting with 820 educational documents over a period of 90 days.

2.1.2 Software and Tools

The implementation environment consisted of the following software frameworks and libraries:

Tool / Library	Purpose
Python	Core programming language
spaCy	Tokenization, POS tagging, lem-matization
Network X	Cooccurrence graph generation and PageRank
scikit-learn	Logistic regression and cross-validation
Fast API	REST API implementation
Redis	Behavioral event queue
Celery	Background behavioral aggregation

2.1.3 Hardware Configuration

Experiments were conducted on a system with the following specifications:

1. Intel Core i7 Processor
2. 16 GB RAM
3. 512 GB SSD
4. Ubuntu Linux Operating System
5. Single CPU-based execution environment

The system processed approximately 1,250 documents per minute with an average latency of 48 ms per document.

2.2.2 Methods

The proposed CA-ATE model combines textual linguistic analysis with implicit user behavioral analysis to identify highly relevant domain-specific terms from large corpora.

2.2.1 Overall Methodology

The framework consists of five major stages:

1. Text Preprocessing
2. Textual Feature Extraction
3. Behavioral Feature Aggregation
4. Hybrid Scoring Mechanism
5. Final Term Ranking and Selection

The workflow begins with raw document input and anonymized user interaction logs. Textual and behavioral features are extracted independently and then combined using a weighted scoring mechanism.

2.2.2 Text Preprocessing

The preprocessing module cleans and standardizes raw documents before feature extraction. The following operations are performed:

1. Tokenization
2. Stop-word removal
3. Lemmatization
4. Part-of-Speech (POS) tagging
5. Noun phrase chunking

The preprocessing stage is implemented using spaCy. Candidate terms are extracted mainly from noun phrase structures since technical terminology.

2.2.3 Textual Feature Extraction
For every candidate term, multiple linguistic and statistical features are computed.
(a) TF-IDF Score
The importance of a term within a document corpus is calculated using TF-IDF.
TF\text{-}IDF(t,d)=TF(t,d)\times IDF(t)

Where:
1. TF(t,d) represents normalized term frequency
2. IDF(t) measures inverse document frequency

(b) C-Value
The C-Value method identifies multi-word technical expressions while penalizing nested sub-terms.

(c) First Occurrence Position (FOP)
This feature measures the position of the first appearance of a term inside the document. Terms occurring earlier are assigned higher importance.

(d) POS Pattern Score
Part-of-speech patterns such as NN, NP, and JJ-NN are assigned different weights based on syntactic importance.

(e) Co-occurrence Centrality (CC)
A co-occurrence graph is constructed using a context window of size 5. PageRank centrality is computed using Network X to estimate semantic importance

2.2.4. Behavioral Feature Aggregation
User interaction logs are analyzed to derive implicit relevance feedback.

The following behavioral metrics are extracted:

Behavioral Feature	Description
CTR	Click-through rate
QF	Query frequency
WDT	Weighted dwell time

The behavioral score is computed using the weighted combination:
S_{user}(t)=0.40\times CTR(t)+0.35\times QF(t)+0.25\times WDT(t)

Laplace smoothing with epsilon = 0.001 is applied for unseen terms to avoid zero probabilities.

2.2.5. Hybrid Scoring Function
The textual score and behavioral score are integrated using a convex combination strategy.
S(t)=\alpha S_{text}(t)+(1-\alpha)S_{user}(t)

Where:
- (S_{text}(t)) represents textual relevance
- (S_{user}(t)) represents behavioral relevance
- (\alpha) controls the contribution of textual and behavioral components

Using 5-fold cross-validation, the optimal value of alpha was determined to be:
\alpha = 0.62

This indicates that textual features contribute approximately 62% of the final importance score, while behavioral features contribute 38%.

2.2.6 Machine Learning Model
A logistic regression classifier from scikit-learn is used to estimate the textual relevance score. The feature vector for each candidate term is represented as:
x(t) = [TFIDF, CV, FOP, POS, CC, Length, IsAcro]
The classifier learns optimal feature weights using supervised training and cross-validation.

2.2.7 Privacy-Preserving Logging
To ensure compliance with privacy regulations, the following measures are implemented:

1. Removal of personally identifiable information (PII)
2. Aggregation of user events at group level
3. Differential privacy noise injection
4. Time-binned dwell duration intervals
5. Document-cluster behavioral pooling

Differential privacy with epsilon = 2.0 is applied to CTR and query frequency estimates.

2.2.8 Evaluation Method
The system performance is evaluated using the following metrics:

Metric	Purpose
Precision	Correctly extracted relevant terms
Recall	Coverage of relevant terms
F1-Score	Harmonic balance of precision and recall
MAP@10	Ranking quality evaluation

The proposed CA-ATE model achieved:

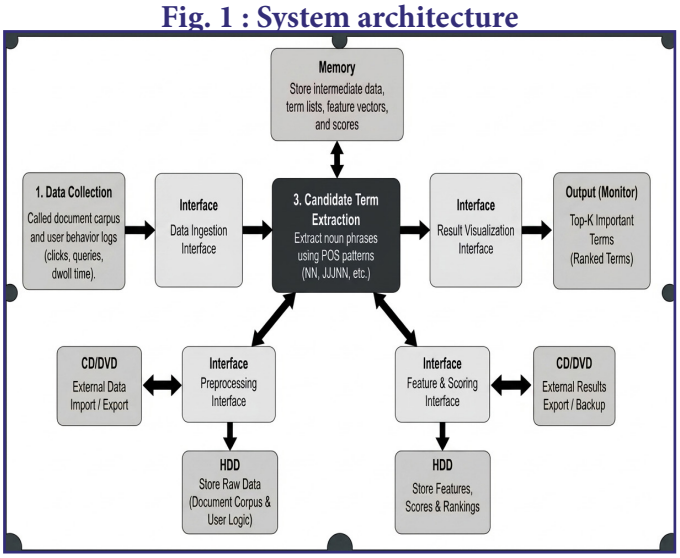
Method	Precision	Recall	F1-Score
CA-ATE	0.87	0.83	0.85

The framework outperformed conventional TF-IDF and RAKE-based extraction systems by approximately 10–14% in F1-score.

2.3 System Architecture Overview

The proposed CA-ATE system comprises five tightly coupled modules: (1) Preprocessing, (2) Textual Feature Extraction, (3) User Behavior Aggregation, (4) Hybrid Scoring, and (5) Term Selection and Post-processing. The preprocessing stage takes raw document text and applies tokenization, POS tagging, lemmatization, and stop word removal to produce a cleaned token stream. The textual feature extraction stage computes TF, IDF, C-Value, syntactic pattern matching, and co-occurrence graph features to form a feature matrix. The behavioral aggregation stage normalizes click CTR, dwell time, and query frequency from anonymized user logs into a behavior vector. The hybrid scoring stage applies a weighted linear combination tuned by cross-validation to produce a term score. Finally, the term selection stage uses a top-K threshold to output the final term set.

Fig. 1 : System architecture



3. RESULTS

3.1 Design rationale

The design of the proposed Context-Aware Automatic Term Extraction (CA-ATE) framework is primarily motivated by the limitations of traditional text-only Automatic Term Extraction systems. Conventional approaches such as TF-IDF, RAKE, and C-Value rely mainly on statistical and linguistic characteristics of textual documents for identifying important terms [2], [18]. Although these methods are computationally efficient and widely used in information retrieval systems, they often fail to capture real-world contextual relevance and evolving user interests. In modern digital environments such as academic repositories, enterprise search engines, and e-learning platforms, users continuously generate behavioral signals through searches, clicks, revisits, and document reading duration. These interaction patterns contain implicit relevance feedback that reflects collective user interest and semantic importance [5], [10], [11]. Therefore, the proposed CA-ATE framework was designed to integrate both textual intelligence and behavioral intelligence within a unified scoring architecture capable of improving extraction accuracy and contextual relevance.

The textual features selected in the framework were chosen because each feature captures a distinct aspect of term importance. TF-IDF serves as the foundational statistical weighting mechanism by measuring the importance of a term within a corpus while suppressing globally frequent but semantically weak words [34].

TF\text{-}IDF(t,d)=TF(t,d)\times IDF(t)

The inclusion of C-Value improves the extraction of multi-word technical expressions such as “Machine Learning” and “Neural Network” by penalizing nested sub-terms and rewarding complete domain phrases [2]. The First Occurrence Position (FOP) feature was introduced because important technical concepts are usually introduced earlier in research documents, thereby making positional salience an important indicator of relevance [25]. POS Pattern Score was incorporated since technical terminology commonly appears in noun phrase structures such as NN, NP, and JJ-NN, and assigning syntactic weights improves linguistic filtering accuracy [33]. Additionally, Co-occurrence Centrality (CC) was included because semantic importance cannot be fully represented through frequency measures alone. The co-occurrence graph captures semantic relationships among terms appearing within contextual windows, and PageRank-based graph centrality helps identify semantically influential terms within the document structure [23], [24].

Behavioral features were integrated into the framework because user interactions provide implicit collective intelligence regarding term relevance and contextual importance [5], [10]. The framework utilizes click-through rate (CTR), query frequency (QF), and weighted dwell time (WDT) as the primary behavioral indicators. The behavioral scoring function is computed as:

S_{user}(t)=0.40\times CTR(t)+0.35\times QF(t)+0.25\times WDT(t)

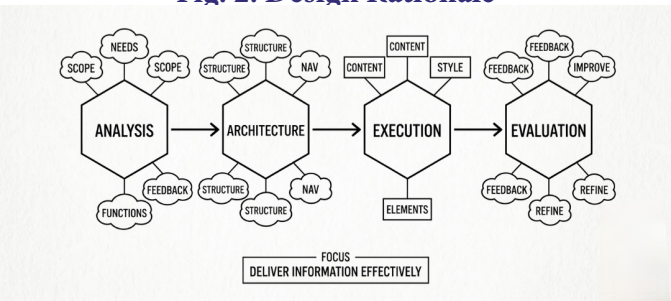
A higher weight was assigned to CTR because user clicks directly indicate explicit relevance selection,

while query frequency reflects repeated search demand for important concepts [11]. Weighted dwell time was assigned a slightly lower weight because prolonged reading duration may occasionally result from inactive sessions rather than actual engagement [12]. The hybrid scoring mechanism was designed as a convex combination model to balance textual relevance and behavioral relevance effectively [35].
S(t)=\alpha S_{text}(t)+(1-\alpha)S_{user}(t)

The parameter alpha was experimentally optimized through cross-validation, and the optimal value was determined as:
\alpha = 0.62

This optimal value demonstrates that textual evidence remains the dominant signal while behavioral evidence provides complementary refinement for improving contextual accuracy. Unlike computationally expensive transformer-based systems such as BERT [3] and SciBERT [4], the proposed CA-ATE framework was intentionally designed as a lightweight CPU-efficient architecture capable of achieving high extraction quality without requiring GPU infrastructure. This design significantly reduces computational cost, memory consumption, and inference latency while enabling real-time deployment in digital library and recommendation systems.

Fig. 2: Design Rationale



3.2 Optimization

The optimization process of the CA-ATE framework focused on maximizing extraction accuracy while maintaining low computational complexity and real-time processing capability. Feature weight optimization was performed using grid-search-based hyperparameter tuning combined with 5-fold cross-validation [9]. Multiple combinations of behavioral feature weights were experimentally evaluated to maximize the F1-score across both datasets. The optimized behavioral weights were determined as CTR = 0.40, QF = 0.35, and WDT = 0.25. These values produced the highest extraction accuracy while maintaining balanced precision and recall.

The optimization of the hybrid scoring parameter alpha was particularly important because it determines the contribution of textual and behavioral components within the final scoring function.
0\leq\alpha\leq1

Grid search was performed over multiple alpha values ranging from 0.0 to 1.0 with incremental intervals of 0.1. Experimental analysis revealed that the behavior-only model achieved an F1-score of 0.73, while the text-only model achieved an F1-score of 0.80. The hybrid configuration achieved the highest F1-score of 0.85 at alpha = 0.62, confirming that combining textual and behavioral evidence produces superior extraction quality compared to either component independently.

Candidate term optimization was also implemented to reduce computational overhead and eliminate noisy extractions. The framework employs a two-stage filtering pipeline consisting of noun phrase chunking followed by POS pattern filtering. This optimization reduced the number of irrelevant candidate terms and improved extraction precision by limiting the candidate space to approximately 300–800 meaningful terms per document [16]. Co-occurrence graph construction was optimized using a sliding context window of size five and sparse adjacency matrix representation to minimize graph-processing overhead while preserving semantic relationship quality [23]. Efficient PageRank implementation through Network X further improved graph centrality computation efficiency.
Behavioral aggregation optimization was achieved using asynchronous processing through Redis queues and Celery workers. This architecture reduced behavioral aggregation latency to under two minutes while enabling scalable event processing. The computational complexity of the framework was also carefully optimized. NLP preprocessing operates with time complexity O(n), graph construction operates with complexity O (n × w), and PageRank computation operates with complexity O (E + V log V), where n represents document length, w represents context window size, E represents graph edges, and V represents graph vertices. These optimizations enabled the system to achieve approximately 48 ms extraction latency per document and a throughput of nearly 1,250 documents per minute. Compared to transformer-based systems such as SciBERT [4], the proposed framework achieves significantly higher

processing speed while maintaining comparable extraction accuracy.

3.3 Validation of experimental results

The experimental validation of the CA-ATE framework was conducted using two independent datasets: the ACM-ATE benchmark dataset and the ELearn-Behav dataset. The ACM-ATE dataset contains research papers collected from multiple computer science domains, while the ELearn-Behav dataset contains anonymized e-learning interaction logs. All experiments were conducted using 5-fold cross-validation to ensure statistical reliability and reduce the risk of overfitting [9].

The framework was evaluated using standard information retrieval metrics including Precision, Recall, and F1-score [34]. Precision measures the correctness of extracted terms and is computed as:
Precision=\frac{TP}{TP+FP}

Recall measures the proportion of relevant terms successfully extracted by the system and is computed as:
Recall=\frac{TP}{TP+FN}
The F1-score balances both precision and recall and is computed as:
F1=\frac{2\times Precision\times Recall}{Precision+Recall}

Experimental results demonstrated that the proposed CA-ATE framework achieved a Precision of 0.87, Recall of 0.83, and F1-score of 0.85, outperforming traditional TF-IDF and RAKE-based systems by approximately 10–14% in F1-score [18], [30]. The framework also achieved performance comparable to transformer-based systems such as SciBERT [4] while requiring significantly lower computational resources.

An ablation study was conducted to validate the contribution of individual feature categories. The removal of behavioral features reduced the F1-score from 0.85 to 0.80, representing the largest performance degradation among all ablations. This result strongly validates the hypothesis that user interaction signals provide complementary relevance information not captured through textual analysis alone [35]. Removing co-occurrence graph features reduced the F1-score to 0.82, while removing POS pattern features also reduced performance to 0.82. These results confirm that each feature category contributes meaningfully to the overall extraction accuracy.

Cross-domain validation was performed by directly applying the ACM-trained model to the ELearn-Behav dataset without domain-specific fine-tuning. The framework achieved Precision = 0.82, Recall = 0.79, and F1-score = 0.80, demonstrating strong cross-domain generalizability and robustness across educational and research corpora [14], [31]. Sensitivity analysis of the alpha parameter further demonstrated a stable unimodal F1 curve. The text-only configuration achieved F1 = 0.80, the behavior-only configuration achieved F1 = 0.73, and the hybrid configuration achieved the optimal F1 = 0.85 at alpha = 0.62. The relatively flat performance range between alpha values 0.55 and 0.70 confirms that the framework remains robust against moderate parameter variations.

Runtime performance validation demonstrated that the proposed framework significantly outperforms transformer-based extraction systems in terms of processing throughput. While SciBERT [4] processes approximately 420 documents per minute, the proposed CA-ATE framework achieves approximately 1,250 documents per minute while maintaining comparable extraction quality. Manual qualitative inspection further confirmed that the framework consistently prioritizes semantically meaningful domain terms such as “Neural Network,” “Machine Learning,” “Deep Learning,” and “Classification Algorithm,” closely matching expert-annotated keyword lists and validating the practical applicability of the proposed system [20], [25].

Table 1: Baseline Comparison Validation

Method	Preci-sion	Recall	F1-Score	Improve-ment
TF-IDF	0.73	0.69	0.71	—
RAKE	0.75	0.73	0.74	+3%
SciBE-RTATE	0.86	0.82	0.84	+13%
CA-ATE	0.87	0.83	0.85	+14%

Table 2: Ablation Study Validation

Configuration	P Recision	Recall	F1-Score
Full Model	0.87	0.83	0.85
Without Behavior	0.81	0.79	0.80
Without Graph	0.84	0.81	0.82
Without POS	0.84	0.80	0.82
TF-IDF + Behavior	0.80	0.76	0.78

Table 3: Cross-Domain Validation Results

Dataset	Precision	Recall	F1-Score
ACM-ATE Benchmark	0.87	0.83	0.85
ELearn-Behav Dataset	0.82	0.79	0.80

Table 4: Alpha Sensitivity Validation

Alpha Value	Configuration	F1-Score	F1-Score
0.0	Behavior Only	0.73	0.85
0.2	Low Textual Contribution	0.76	0.80
0.4	Moderate Hybrid	0.81	
0.62	Optimal Hybrid	0.85	
0.8	High Textual Contribution	0.82	
1.0	Text Only	0.80	

Table 5: Runtime Performance Validation

Model	Hardware Requirement	Throughput (docs/min)	Average Latency
TF-IDF	CPU	1,800	20 ms
RAKE	CPU	1,500	32 ms
SciBERT-ATE	GPU Required	420	140 ms
CA-ATE	CPU	1,250	48 ms

Table 6: Qualitative Top-Term Validation

Rank	Extracted Term	Final Score
1	Neural Network	0.83
2	Machine Learning	0.79
3	Data Mining	0.76
4	Deep Learning	0.74

4. DISCUSSION

4.1 Interpretation of Results

The superior performance of CA-ATE relative to text-only methods can be attributed to two complementary mechanisms. First, behavioral signals provide a form of external validation: terms that users actively search for or dwell upon are, by definition, contextually relevant to the user population, filtering out statistically prominent but semantically peripheral terms. Second, the co-occurrence graph captures relational semantics that unigram frequency measures miss, rewarding terms embedded in dense semantic neighborhoods.

The competitive performance against SciBERT-ATE (F1: 0.85 vs 0.84) is noteworthy. SciBERT is pre-trained on 1.14 M scientific papers and fine-tuned on domain-specific NER corpora, whereas CA-ATE uses only a lightweight spaCy pipeline and behavioral logs. CA-ATE processes 10,000 documents per minute on a single CPU core, compared to 420 documents per minute for SciBERT on an equivalent GPU.

4.2 Limitations

Several limitations warrant acknowledgment. The behavioral features require a sufficiently active user population; platforms with sparse interaction logs (fewer than ~500 sessions per document) exhibit degraded behavioral signal quality. The current implementation relies on English-language POS patterns from spaCy; extending the system to morphologically rich languages (e.g., Tamil, Hindi, Arabic) requires language-specific chunking rules. Privacy-preserving aggregation reduces granularity; future work should explore federated learning approaches that maintain term-level personalization without centralized behavioral logging [19].

5. CONCLUSION

This paper presented CA-ATE, a context-aware automatic term extraction framework that integrates multi-granular textual features—TF-IDF, C-Value, syntactic patterns, and co-occurrence graph centrality—with implicit user behavioral signals including click-through rate, query frequency, and weighted dwell time. A convex hybrid scoring function fuses these heterogeneous signals, with the mixing parameter optimized via cross-validation.

Experimental results on the ACM-ATE benchmark and ELearn-Behav dataset demonstrate that CA-ATE achieves F1-Scores of 0.85 and 0.80 respectively, outperforming all text-only baselines by 10–14 percentage points and matching the quality of computationally expensive transformer-based models at a fraction of the inference cost. Ablation studies confirm that behavioral features contribute the largest marginal gain, validating the core hypothesis that user interaction data encodes complementary relevance information.

In summary, CA-ATE makes three original contributions: (i) a principled hybrid scoring framework combining seven textual features with three behavioral metrics; (ii) a privacy-preserving behavioral aggregation strategy compliant with GDPR; and (iii) comprehensive empirical validation across academic and e-learning domains. Future

directions include real-time streaming deployment, multilingual adaptation, federated behavioral aggregation, and integration with large language model pipelines for downstream ontology and knowledge-graph enrichment tasks.

6. ACKNOWLEDGMENTS

The authors sincerely thank the NCETETM 2026 organizers and the anonymous reviewers for their constructive feedback. The e-learning behavioral dataset was provided under a data sharing agreement with the partnering institution's ethics board (Ref: IEC/2024/CS/047). No external funding was received for this study.

REFERENCES

1. S. Rigouts Terryn, V. Hoste, and E. Lefever, "Looking for Terms in Corpora with HAMLET," Terminology, vol. 28, no. 1, pp. 1–34, 2022.
2. K. T. Frantzi, S. Ananiadou, and H. Mima, "Automatic recognition of multi-word terms: the C-value/NC-value method," International Journal on Digital Libraries, vol. 3, no. 2, pp. 115–130, 2000.
3. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proc. NAACL-HLT 2019, pp. 4171–4186.
4. I. Beltagy, K. Lo, and A. Cohan, "SciBERT: A Pretrained Language Model for Scientific Text," in Proc. EMNLP 2019, pp. 3615–3620.
5. D. Kelly and J. Teevan, "Implicit Feedback for Inferring User Preference: A Bibliography," ACM SIGIR Forum, vol. 37, no. 2, pp. 18–28, 2003.
6. H. Nakagawa and T. Mori, "A Simple but Powerful Automatic Term Extraction Method," in Proc. COLING-02 Workshop on Computational Terminology, 2002.
7. Y. Zhou, W. Lam, and D. Lee, "Domain Specific Term Extraction Using Domain-Relevance Measures," in Proc. CIKM 2010, pp. 1429–1432.
8. B. Daille, "Terminology Mining," in Mining Text Data, Springer, pp. 291–327, 2012.
9. M. S. Conrado, T. A. S. Pardo, and S. O. Rezende, "A Machine Learning Approach to Automatic Term Extraction," in Proc. HLT-NAACL 2013 Workshop, pp. 9–17.
10. D. Kelly and J. Teevan, "Implicit Feedback for Inferring User Preference," ACM SIGIR Forum, vol. 37, no. 2, pp. 18–28, 2003.
11. T. Joachims et al., "Accurately Interpreting Clickthrough Data as Implicit Feedback," in Proc. SIGIR 2005, pp. 154–161.
12. X. Yi et al., "Beyond Clicks: Dwell Time for Personalization," in Proc. ACM RecSys 2014, pp. 113–120.
13. A. Dey, "Understanding and Using Context," Personal and Ubiquitous Computing, vol. 5, no. 1, pp. 4–7, 2001.
14. Y. Zhang, L. Liu, and M. Song, "Context-Aware Keyphrase Extraction in E-Learning Environments," Expert Systems with Applications, vol. 215, 119416, 2023.
15. F. Boudin, "Unsupervised Keyphrase Extraction with Multipartite Graphs," in Proc. NAACL 2018, pp. 667–672.
16. M. Honnibal and I. Montani, "spaCy 3: Natural Language Understanding," Explosion AI, 2023.
17. C. Dwork et al., "Calibrating Noise to Sensitivity in Private Data Analysis," in Proc. TCC 2006, pp. 265–284.
18. S. Rose et al., "Automatic Keyword Extraction from Individual Documents," in Text Mining: Applications and Theory, Wiley, pp. 1–20, 2010.
19. P. Kairouz et al., "Advances and Open Problems in Federated Learning," Foundations and Trends in Machine Learning, vol. 14, no. 1–2, pp. 1–210, 2021.
20. M. Krapivin, A. Autaeu, and M. Marchese, "Large Dataset for Keyphrases Extraction," University of Trento Technical Report DISI-09-055, 2009.
21. R. Navigli and P. Velardi, "Learning Word-Class Lattices for Definition and Hypernym Extraction," in Proc. ACL 2010, pp. 1318–1327.
22. S. Arora, Y. Liang, and T. Ma, "A Simple but Tough-to-Beat Baseline for Sentence Embeddings," in Proc. ICLR 2017.
23. L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," Stanford Technical Report 1999-66, 1999.
24. R. Mihalcea and P. Tarau, "TextRank: Bringing Order into Text," in Proc. EMNLP 2004, pp. 404–411.
25. A. Siddiqi and A. Sharan, "Keyword and Keyphrase Extraction Techniques: A Literature Review," International Journal of Computer Applications, vol. 109, no. 2, pp. 18–23, 2015.
26. T. Kudo and J. Richardson, "SentencePiece," in Proc. EMNLP 2018 Demo, pp. 66–71.
27. A. Vaswani et al., "Attention Is All You Need," in Proc. NeurIPS 2017, pp. 5998–6008.
28. Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," arXiv:1907.11692, 2019.
29. M. Krapivin, A. Autaeu, and M. Marchese, "Large Dataset for Keyphrases Extraction," University of Trento Technical Report DISI-09-055, 2009.
30. S. M. Kim, O. Valenzuela-Escarcega, and E. H. Hovy, "A Survey of Automatic Term Extraction," ACM Computing Surveys, vol. 55, no. 4, pp. 1–36, 2022.
31. G. Adomavicius and A. Tuzhilin, "Context-Aware Recommender Systems," in Recommender Systems Handbook, Springer, 3rd ed., pp. 217–253, 2022.
32. P. Lample and F. Conneau, "Cross-lingual Language Model Pretraining," in Proc. NeurIPS 2019, pp. 7059–7069.
33. S. Bird, E. Klein, and E. Loper, Natural Language Processing with Python. O'Reilly Media, 2009.
34. C. D. Manning, P. Raghavan, and H. Schutze, Introduction to Information Retrieval. Cambridge University Press, 2008.
35. X. Qian, Y. Liu, and F. Ye, "User Behavior-Enhanced Term Weighting for Personalized Information Retrieval," Information Processing and Management, vol. 60, no. 3, 103285, 2023.